USE [REPORT\_FR2052A]

GO

/\*\*\*\*\*\* Object: Schema [app] Script Date: 07/11/2017 22:35:59 \*\*\*\*\*\*/

CREATE SCHEMA [app]

GO

/\*\*\*\*\*\* Object: Schema [SRC] Script Date: 07/11/2017 22:35:59 \*\*\*\*\*\*/

CREATE SCHEMA [SRC]

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_T24\_MULTI\_LD\_BALANCES\_FULL\_DETAIL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_T24\_MULTI\_LD\_BALANCES\_FULL\_DETAIL](

[NID] [int] IDENTITY(1,1) NOT NULL,

[CONTRACT\_ID] [varchar](12) NOT NULL,

[DATE] [datetime] NULL,

[TOTAL\_PAYMENT] [decimal](19, 2) NULL,

[PRINCIPAL] [decimal](19, 2) NULL,

[INTEREST] [decimal](19, 2) NULL,

[COMMISSION] [decimal](19, 2) NULL,

[FEES] [decimal](19, 2) NULL,

[CHARGES] [decimal](19, 2) NULL,

[OUTSTANDING] [decimal](19, 2) NULL,

[COMPANY\_CODE] [varchar](9) NULL,

[COMPANY] [varchar](3) NOT NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMMENTS] [varchar](50) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_T24\_MULTI\_LD\_BALANCES\_FULL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_T24\_MULTI\_LD\_BALANCES\_FULL](

[nid] [int] IDENTITY(1,1) NOT NULL,

[CONTRACT\_ID] [varchar](12) NOT NULL,

[CUSTOMER] [varchar](7) NULL,

[CATEGORY] [varchar](5) NULL,

[CURRENCY] [varchar](3) NULL,

[VALUE\_DATE] [datetime] NULL,

[MATURITY] [datetime] NULL,

[CURRENT\_RATE] [decimal](9, 6) NULL,

[STATUS] [varchar](3) NULL,

[OD\_STATUS] [varchar](50) NULL,

[OPEN\_BALANCE] [decimal](19, 2) NULL,

[CLOSING\_BALANCE] [decimal](19, 2) NULL,

[OPEN\_DATE] [datetime] NULL,

[CLOSE\_DATE] [datetime] NULL,

[COMPANY\_CODE] [varchar](9) NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_T24\_MULTI\_LD\_BALANCE] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_T24\_MULTI\_LD\_BALANCE](

[ID] [varchar](50) NULL,

[DATE] [varchar](50) NULL,

[TOTAL\_PAYMENT] [decimal](19, 2) NULL,

[PRINCIPAL] [decimal](19, 2) NULL,

[INTEREST] [decimal](19, 2) NULL,

[COMMISSION] [decimal](19, 2) NULL,

[FEES] [decimal](19, 2) NULL,

[CHARGES] [decimal](19, 2) NULL,

[OUTSTANDING\_BAL] [decimal](19, 2) NULL,

[CO\_CODE] [varchar](10) NULL,

[COMPANY] [varchar](3) NULL,

[ASOFDATE] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_TPOS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_TPOS](

[BR] [char](2) NOT NULL,

[COST] [char](10) NOT NULL,

[CCY] [char](3) NOT NULL,

[INVTYPE] [char](1) NOT NULL,

[PORT] [char](4) NOT NULL,

[SECID] [char](20) NOT NULL,

[ACCROUTST] [numeric](19, 4) NULL,

[ACCROUTSTYST] [numeric](19, 4) NULL,

[AMORTNEXTMNTH] [numeric](19, 4) NULL,

[PMEDSCINCEXP] [numeric](19, 4) NULL,

[PYEDSCINCEXP] [numeric](19, 4) NULL,

[TDYDSCINCEXP] [numeric](19, 4) NULL,

[YSTDSCINCEXP] [numeric](19, 4) NULL,

[AVGCOST] [numeric](19, 4) NULL,

[AVGCOSTYST] [numeric](19, 4) NULL,

[INTNEXTMONTH] [numeric](19, 4) NULL,

[PMEINTINCEXP] [numeric](19, 4) NULL,

[PYEINTINCEXP] [numeric](19, 4) NULL,

[TDYINTINCEXP] [numeric](19, 4) NULL,

[YSTINTINCEXP] [numeric](19, 4) NULL,

[INVADYAMT] [numeric](19, 4) NULL,

[MKTVAL] [numeric](19, 4) NULL,

[MKTVALYST] [numeric](19, 4) NULL,

[PRINAMT] [numeric](19, 4) NULL,

[PRINAMTYST] [numeric](19, 4) NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[PURCHAVGCOST] [numeric](19, 4) NULL,

[PURCHQTY] [numeric](19, 4) NULL,

[QTY] [numeric](19, 4) NULL,

[QTYYST] [numeric](19, 4) NULL,

[SALEAVGCOST] [numeric](19, 4) NULL,

[SALEQTY] [numeric](19, 4) NULL,

[SETTAVGCOST] [numeric](19, 4) NULL,

[SETTAVGCOSTYST] [numeric](19, 4) NULL,

[SETTQTY] [numeric](19, 4) NULL,

[SETTQTYYST] [numeric](19, 4) NULL,

[UNAMORTAMT] [numeric](19, 4) NULL,

[UNAMORTAMTYST] [numeric](19, 4) NULL,

[TDYSALEGAIN] [numeric](19, 4) NULL,

[YSTSALEGAIN] [numeric](19, 4) NULL,

[PMESALEGAIN] [numeric](19, 4) NULL,

[PYESALEGAIN] [numeric](19, 4) NULL,

[TDYMTM] [numeric](19, 4) NULL,

[YSTMTM] [numeric](19, 4) NULL,

[PMEMTM] [numeric](19, 4) NULL,

[PYEMTM] [numeric](19, 4) NULL,

[CLSGPRICE\_8] [numeric](18, 8) NULL,

[INTPAYAMT] [numeric](19, 4) NULL,

[TDYUGLAMT] [numeric](19, 4) NULL,

[YSTUGLAMT] [numeric](19, 4) NULL,

[PMEUGLAMT] [numeric](19, 4) NULL,

[PYEUGLAMT] [numeric](19, 4) NULL,

[TDACCROUTST] [numeric](19, 4) NULL,

[TDACCROUTSTYST] [numeric](19, 4) NULL,

[SETTPURCHINTAMT] [numeric](19, 4) NULL,

[SETTPURCHINTAMTYST] [numeric](19, 4) NULL,

[TDSETTPURCHINTAMT] [numeric](19, 4) NULL,

[TDSETTPURCHINTAMTYST] [numeric](19, 4) NULL,

[TDYINFADJAMT] [numeric](19, 4) NULL,

[YSTINFADJAMT] [numeric](19, 4) NULL,

[PMEINFADJAMT] [numeric](19, 4) NULL,

[PYEINFADJAMT] [numeric](19, 4) NULL,

[ADJNOMAMT] [numeric](19, 4) NULL,

[ADJNOMAMTYST] [numeric](19, 4) NULL,

[TDPRINAMT] [numeric](19, 4) NULL,

[TDPRINAMTYST] [numeric](19, 4) NULL,

[TDUNAMORTAMT] [numeric](19, 4) NULL,

[TDUNAMORTAMTYST] [numeric](19, 4) NULL,

[TDBOOKVALUE] [numeric](19, 4) NULL,

[TDBOOKVALUEYST] [numeric](19, 4) NULL,

[TDMARKETVALUE] [numeric](19, 4) NULL,

[TDMARKETVALUEYST] [numeric](19, 4) NULL,

[ADJTDBOOKVAL] [numeric](19, 4) NULL,

[ADJTDBOOKVALYST] [numeric](19, 4) NULL,

[ADJTDMTMPYE] [numeric](19, 4) NULL,

[ADJTDMTMPME] [numeric](19, 4) NULL,

[ADJTDMTM] [numeric](19, 4) NULL,

[ADJTDMTMYST] [numeric](19, 4) NULL,

[TDRGLADJPYE] [numeric](19, 4) NULL,

[TDRGLADJPME] [numeric](19, 4) NULL,

[TDRGLADJ] [numeric](19, 4) NULL,

[TDRGLADJYST] [numeric](19, 4) NULL,

[ADJTDMKTVAL] [numeric](19, 4) NULL,

[ADJTDMKTVALYST] [numeric](19, 4) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SSAH] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SSAH](

[BR] [char](2) NOT NULL,

[ACCOUNTNO] [char](15) NOT NULL,

[SECID] [char](20) NOT NULL,

[TDYPOOLQTY] [numeric](19, 4) NULL,

[YSTPOOLQTY] [numeric](19, 4) NULL,

[PMEPOOLQTY] [numeric](19, 4) NULL,

[PYEPOOLQTY] [numeric](19, 4) NULL,

[TDYSEGQTY] [numeric](19, 4) NULL,

[YSTSEGQTY] [numeric](19, 4) NULL,

[PMESEGQTY] [numeric](19, 4) NULL,

[PYESEGQTY] [numeric](19, 4) NULL,

[ASOFDATE] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SPSH] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SPSH](

[BR] [char](2) NOT NULL,

[DEALNO] [char](8) NOT NULL,

[SEQ] [char](4) NOT NULL,

[FIXINCIND] [char](1) NOT NULL,

[AETRAD] [char](4) NULL,

[ASSIGNIND] [char](1) NULL,

[BVAMORTAMT] [numeric](19, 4) NULL,

[BVINTAMT] [numeric](19, 4) NULL,

[BRPRCINDTE] [datetime] NULL,

[BROK] [char](11) NULL,

[BROKFEEAMT] [numeric](19, 4) NULL,

[BROKFEECCY] [char](3) NULL,

[CHARGEAMT] [numeric](19, 4) NULL,

[CLEARDATE] [datetime] NULL,

[CLEARINPUTDATE] [datetime] NULL,

[CLEAROPER] [char](4) NULL,

[COMMAMT] [numeric](19, 4) NULL,

[COMMRATE\_8] [numeric](18, 8) NULL,

[COMREFNO] [char](16) NULL,

[CONFDATE] [datetime] NULL,

[CONFOPER] [char](4) NULL,

[COST] [char](10) NULL,

[COUPONNO] [char](7) NULL,

[CCYAUTHDATE] [datetime] NULL,

[CCYAUTHIND] [char](1) NULL,

[CCYAUTHOPER] [char](4) NULL,

[CCY] [char](3) NULL,

[CNO] [char](10) NULL,

[CUSTCONFDATE] [datetime] NULL,

[CUSTREFNO] [char](16) NULL,

[DEALDATE] [datetime] NULL,

[DEALSRCE] [char](10) NULL,

[DEALTEXT] [char](70) NULL,

[DEALTIME] [char](8) NULL,

[DISCAMT] [numeric](19, 4) NULL,

[DISCRATE\_8] [numeric](18, 8) NULL,

[DISPRICE\_8] [numeric](18, 8) NULL,

[FACEAMT] [numeric](19, 4) NULL,

[FAILDATE] [datetime] NULL,

[FAILINPUTDATE] [datetime] NULL,

[FAILOPER] [char](4) NULL,

[FAILINTAMT] [numeric](19, 4) NULL,

[FEEAMT] [numeric](19, 4) NULL,

[GAINAMT] [numeric](19, 4) NULL,

[INPUTDATE] [datetime] NULL,

[IOPER] [char](4) NULL,

[INPUTTIME] [char](8) NULL,

[INVTYPE] [char](1) NULL,

[LINKORDERNO] [char](8) NULL,

[LINKSEQ] [char](4) NULL,

[LOSSAMT] [numeric](19, 4) NULL,

[MUTFUNDIND] [char](1) NULL,

[ORIGAMT] [numeric](19, 4) NULL,

[ORIGQTY] [numeric](19, 4) NULL,

[PORT] [char](4) NULL,

[PREMAMT] [numeric](19, 4) NULL,

[PRICE\_8] [numeric](18, 8) NULL,

[PRINAMT] [numeric](19, 4) NULL,

[PRINAGIND] [char](1) NULL,

[PROCEEDAMT] [numeric](19, 4) NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[PS] [char](1) NULL,

[PURCHINTAMT] [numeric](19, 4) NULL,

[QTY] [numeric](19, 4) NULL,

[REVDATE] [datetime] NULL,

[REVOPER] [char](4) NULL,

[REVREASON] [char](2) NULL,

[REVTEXT] [char](70) NULL,

[REVTIME] [char](8) NULL,

[SECSACCT] [char](15) NULL,

[SECAUTHDATE] [datetime] NULL,

[SECAUTHIND] [char](1) NULL,

[SECAUTHOPER] [char](4) NULL,

[SECID] [char](20) NULL,

[SERIES] [char](5) NULL,

[CCYSACCT] [char](15) NULL,

[SETTDATE] [datetime] NULL,

[CCYSMEANS] [char](7) NULL,

[SPREADRATE\_8] [numeric](18, 8) NULL,

[SUPPCONFIND] [char](1) NULL,

[SYMBOLID] [char](8) NULL,

[VATAMT] [numeric](19, 4) NULL,

[TRAD] [char](4) NULL,

[VERIND] [char](1) NULL,

[VERDATE] [datetime] NULL,

[VOPER] [char](4) NULL,

[WHENISSIND] [char](1) NULL,

[YIELD\_8] [numeric](18, 8) NULL,

[EXCHCNO] [char](11) NULL,

[EXCHCOMMAMT] [numeric](19, 4) NULL,

[EXCHCCY] [char](3) NULL,

[CNARR] [char](70) NULL,

[FROMDAYS] [char](4) NULL,

[TODAYS] [char](4) NULL,

[DELAYDELIVIND] [char](1) NULL,

[ISSUER] [char](11) NULL,

[RATCNO] [char](11) NULL,

[CRATING] [char](10) NULL,

[RATECODE] [char](7) NULL,

[AGENCY] [char](11) NULL,

[WHTAMT] [numeric](19, 4) NULL,

[EXCHIND] [char](1) NULL,

[OFFASSAMT] [numeric](19, 4) NULL,

[ESTAMT] [numeric](19, 4) NULL,

[ASSIGNEDQTY] [numeric](19, 4) NULL,

[CONVINTAMT] [numeric](19, 4) NULL,

[CONVINTBAMT] [numeric](19, 4) NULL,

[SETTAMT] [numeric](19, 4) NULL,

[SETTBASEAMT] [numeric](19, 4) NULL,

[COSTAMT] [numeric](19, 4) NULL,

[COSTBAMT] [numeric](19, 4) NULL,

[INTCCY] [char](3) NULL,

[SETTCCY] [char](3) NULL,

[SETTPROCEXCHRATE\_8] [numeric](18, 8) NULL,

[SETTPROCTERMS] [char](1) NULL,

[SETTPROCPREMDISC\_8] [numeric](18, 8) NULL,

[INTSETTEXCHRATE\_8] [numeric](18, 8) NULL,

[INTSETTTERMS] [char](1) NULL,

[INTSETTPREMDISC\_8] [numeric](18, 8) NULL,

[SETTBASEEXCHRATE\_8] [numeric](18, 8) NULL,

[SETTBASETERMS] [char](1) NULL,

[SETTBASEPREMDISC\_8] [numeric](18, 8) NULL,

[INTBASEEXCHRATE\_8] [numeric](18, 8) NULL,

[INTBASETERMS] [char](1) NULL,

[INTBASEPREMDISC\_8] [numeric](18, 8) NULL,

[AMENDDATE] [datetime] NULL,

[AMENDTIME] [char](8) NULL,

[AMENDOPER] [char](4) NULL,

[FEDEALNO] [char](15) NULL,

[INTERNALYIELD\_8] [numeric](18, 8) NULL,

[INTERNALDISCRATE\_8] [numeric](18, 8) NULL,

[INTERNALPRICE\_8] [numeric](18, 8) NULL,

[INTERNALPROCEEDAMT] [numeric](19, 4) NULL,

[INTERNALSPREADRATE\_8] [numeric](18, 8) NULL,

[MARKUPCOMMAMT] [numeric](19, 4) NULL,

[AMORTYIELD\_8] [numeric](18, 8) NULL,

[AMORTIND] [char](1) NULL,

[PLEDGEID] [char](30) NULL,

[PAIREDFACEAMT] [numeric](19, 4) NULL,

[PAIREDAMT] [numeric](19, 4) NULL,

[PAIREDIND] [char](1) NULL,

[CURRSPLITSEQ] [char](4) NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[BTBDEALNO] [char](8) NULL,

[BTBPORT] [char](4) NULL,

[USUALID] [char](15) NULL,

[LINKDEALNO] [char](8) NULL,

[LINKIND] [char](1) NULL,

[BLOCKIND] [char](1) NULL,

[SPLITIND] [char](1) NULL,

[INDEXRATIO\_8] [numeric](18, 8) NULL,

[SWIFTBOTHIND] [char](1) NULL,

[NEWISSUEIND] [char](1) NULL,

[CCPIND] [char](1) NULL,

[EXTCOMPARE] [char](1) NULL,

[NOVFACEAMT] [numeric](19, 4) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SECS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SECS](

[SECID] [char](20) NOT NULL,

[SEQ] [char](4) NOT NULL,

[IPAYDATE] [datetime] NOT NULL,

[CALLPRICE\_8] [numeric](18, 8) NULL,

[CASHFLOW\_8] [numeric](18, 8) NULL,

[INTRATE\_8] [numeric](18, 8) NULL,

[EXDIVDATE] [datetime] NULL,

[INTPAYAMT\_8] [numeric](18, 8) NULL,

[INTENDDTE] [datetime] NULL,

[INTSTRTDTE] [datetime] NULL,

[CAPRATE\_8] [numeric](18, 8) NULL,

[RATECODE] [char](7) NULL,

[RATEFIXDATE] [datetime] NULL,

[FLOORRATE\_8] [numeric](18, 8) NULL,

[SPREADRATE\_8] [numeric](18, 8) NULL,

[LSTMNTDATE] [datetime] NULL,

[PRINAMT\_8] [numeric](18, 8) NULL,

[PRINPAYAMT\_8] [numeric](18, 8) NULL,

[PUTPRICE\_8] [numeric](18, 8) NULL,

[STEPRATE\_8] [numeric](18, 8) NULL,

[EXERDATE] [datetime] NULL,

[PUBLISHDATE] [datetime] NULL,

[BASIS] [char](6) NULL,

[NOTPRINAMT\_8] [numeric](18, 8) NULL,

[NOTPRINPAYAMT\_8] [numeric](18, 8) NULL,

[NOTINTPAYAMT\_8] [numeric](18, 8) NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[GROUPID] [int] NULL,

[FORMULAID] [int] NULL,

[FIXINGOVERRIDE] [int] NULL,

[RATESETDATE] [datetime] NULL,

[INTSETTCCY] [char](3) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SECM] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SECM](

[SECID] [char](20) NOT NULL,

[ACCTNGTYPE] [char](10) NULL,

[CALLDATE] [datetime] NULL,

[CALLPRICE\_8] [numeric](18, 8) NULL,

[CLSGINDXRATE\_8] [numeric](18, 8) NULL,

[CONVCOUPRATE\_8] [numeric](18, 8) NULL,

[CONVDATE] [datetime] NULL,

[COUPONNO] [char](8) NULL,

[COUPRATE\_8] [numeric](18, 8) NULL,

[COUPENDDATE] [datetime] NULL,

[CCY] [char](3) NULL,

[DENOM] [char](15) NULL,

[DESCR] [char](70) NULL,

[ENDTIME] [char](8) NULL,

[FIRSTIPAYDATE] [datetime] NULL,

[FIXCCY] [char](3) NULL,

[GUARANTOR] [char](10) NULL,

[INFRATECODE] [char](7) NULL,

[INITINDEXRATE\_8] [numeric](18, 8) NULL,

[BASIS] [char](6) NULL,

[INTCALCRULE] [char](3) NULL,

[INTPAYCYCLE] [char](4) NULL,

[INTPAYRULE] [char](1) NULL,

[RATECODE] [char](7) NULL,

[ISSDATE] [datetime] NULL,

[ISSUER] [char](10) NULL,

[LASTIPAYDATE] [datetime] NULL,

[LSTMNTDATE] [datetime] NULL,

[MARKMAKIND] [char](1) NULL,

[MDATE] [datetime] NULL,

[MUTFUNDIND] [char](1) NULL,

[NORMCOUPPERIOD] [char](4) NULL,

[PARAMT] [numeric](19, 4) NULL,

[PRICEDECS] [char](8) NULL,

[PRICERNDRULE] [char](1) NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[PUTDATE] [datetime] NULL,

[PUTPRICE\_8] [numeric](18, 8) NULL,

[RATEDECS] [char](8) NULL,

[RATERNDRULE] [char](1) NULL,

[RATETOL] [char](4) NULL,

[REDEMPAMT] [numeric](19, 4) NULL,

[REPOCOLLPCT] [char](5) NULL,

[SERIES] [char](5) NULL,

[SETTDAYS] [char](2) NULL,

[SPREADRATE\_8] [numeric](18, 8) NULL,

[STARTTIME] [char](8) NULL,

[SYMBOLID] [char](8) NULL,

[TAXIND] [char](1) NULL,

[TENOR] [char](2) NULL,

[TRADSTAT] [char](1) NULL,

[SECUNIT] [char](3) NULL,

[VDATE] [datetime] NULL,

[EXCHCNO] [char](11) NULL,

[WHACCNO] [char](15) NULL,

[SETTCCY] [char](3) NULL,

[PRICETOL] [char](4) NULL,

[CALLTOPAR] [datetime] NULL,

[PUTTOPAR] [datetime] NULL,

[MINDENOM] [char](15) NULL,

[MININCREMENT] [char](15) NULL,

[LINKSECID] [char](20) NULL,

[INTCCY] [char](3) NULL,

[INTEXCHRATE\_8] [numeric](18, 8) NULL,

[INTEXCHTERMS] [char](1) NULL,

[SECMSIC] [char](10) NULL,

[REPOFLAG] [char](1) NULL,

[EXDIVDAYS] [char](3) NULL,

[DCPRICEIND] [char](1) NULL,

[FINCENT] [char](15) NULL,

[EXERCISEDAYS] [char](3) NULL,

[INTENDDATERULE] [char](1) NULL,

[SHORTCOUPIND] [char](1) NULL,

[EMTAIND] [char](1) NULL,

[PUBLISHDAYS] [char](3) NULL,

[ANNPERS] [char](3) NULL,

[FIXPERS] [char](3) NULL,

[FIXRATE\_8] [numeric](18, 8) NULL,

[CASHDECS] [char](2) NULL,

[AIDECS] [char](2) NULL,

[PMTFREQ] [char](4) NULL,

[PRINPMTFREQ] [char](4) NULL,

[FIXFREQ] [char](4) NULL,

[MAXMFIX\_8] [numeric](18, 8) NULL,

[MAXMLIFE\_8] [numeric](18, 8) NULL,

[INTRNDRULE] [char](1) NULL,

[MININCRMT] [char](8) NULL,

[NOAMORTS] [char](3) NULL,

[LASTAMORT\_8] [numeric](18, 8) NULL,

[PITYPE] [char](1) NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[PHYSDELIND] [char](1) NULL,

[DVPIND] [char](1) NULL,

[RECORDDATEIND] [char](1) NULL,

[PERRATEFLOOR\_12] [numeric](18, 12) NULL,

[LIFERATEFLOOR\_12] [numeric](18, 12) NULL,

[DELAYDAYS] [char](3) NULL,

[PMTDAY] [char](2) NULL,

[SETTLECLASS] [char](15) NULL,

[PMTCAP\_12] [numeric](18, 12) NULL,

[POOLNO] [char](15) NULL,

[CASHFLOWIND] [char](1) NULL,

[PREPMTMODEL] [char](15) NULL,

[WITHTAXRATE\_12] [numeric](18, 12) NULL,

[FACTSECIND] [char](1) NULL,

[AMT1] [numeric](19, 4) NULL,

[AMT2] [numeric](19, 4) NULL,

[RATE1\_12] [numeric](18, 12) NULL,

[RATE2\_12] [numeric](18, 12) NULL,

[DATE1] [datetime] NULL,

[DATE2] [datetime] NULL,

[INDEXRATECODE] [char](7) NULL,

[ILIND] [char](1) NULL,

[DEFLATIONIND] [char](1) NULL,

[ILDECS] [char](2) NULL,

[NGPRICEIND] [char](1) NULL,

[CPIDAYCYCLE] [char](2) NULL,

[SCHEDID] [int] NULL,

[PRODID] [int] NULL,

[STRUCTID] [char](1) NULL,

[MKTVALSOURCE] [char](15) NULL,

[FORMULAID] [int] NULL,

[INTSETTCCY] [char](3) NULL,

[RATREVFIRSTLAST] [char](1) NULL,

[RATEFIXDAYS] [char](3) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SECL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SECL](

[BR] [char](2) NOT NULL,

[SECID] [char](20) NOT NULL,

[ASKPRICE\_8] [numeric](18, 8) NULL,

[ASKPRICEYST\_8] [numeric](18, 8) NULL,

[ASKRATE\_8] [numeric](18, 8) NULL,

[ASKRATEYST\_8] [numeric](18, 8) NULL,

[BIDPRICE\_8] [numeric](18, 8) NULL,

[BIDPRICEYST\_8] [numeric](18, 8) NULL,

[BIDRATE\_8] [numeric](18, 8) NULL,

[BIDRATEYST\_8] [numeric](18, 8) NULL,

[CLSGPRICE\_8] [numeric](18, 8) NULL,

[CLSGPRICEYST\_8] [numeric](18, 8) NULL,

[CLSGRATE\_8] [numeric](18, 8) NULL,

[CLSGRATEYST\_8] [numeric](18, 8) NULL,

[COUPREMAINAMT] [numeric](19, 4) NULL,

[CURRFIXRATE\_8] [numeric](18, 8) NULL,

[EFFLAGDAY] [char](4) NULL,

[FIXBASERATE\_8] [numeric](18, 8) NULL,

[FIXBASETERMS] [char](1) NULL,

[FIXCYCLE] [char](1) NULL,

[FIXFREQ] [char](3) NULL,

[FIXDAY] [char](2) NULL,

[FIXLAGDAY] [char](4) NULL,

[FIXSPOTRATE\_8] [numeric](18, 8) NULL,

[FIXSPOTDAYS] [char](2) NULL,

[FIXSPOTTERMS] [char](1) NULL,

[LSTPRCEMNTDATE] [datetime] NULL,

[MARKMAKIND] [char](1) NULL,

[NEXTEFFDATE] [datetime] NULL,

[NEXTFIXDATE] [datetime] NULL,

[NEXTFIXRATE\_8] [numeric](18, 8) NULL,

[NEXTIPAYDATE] [datetime] NULL,

[OFFCLSGPRICE\_8] [numeric](18, 8) NULL,

[OFFCLSGPRICEYST\_8] [numeric](18, 8) NULL,

[OFFCLSGRATE\_8] [numeric](18, 8) NULL,

[OFFCLSGRATEYST\_8] [numeric](18, 8) NULL,

[PREVCOUPDATE] [datetime] NULL,

[PREVCOUPRATE\_8] [numeric](18, 8) NULL,

[PAYCNO] [char](1) NULL,

[SECSACCT] [char](15) NULL,

[SETTMEANS] [char](7) NULL,

[SETTACCT] [char](15) NULL,

[TAXIND] [char](1) NULL,

[LSTMNTDATE] [datetime] NULL,

[EFFDATE] [datetime] NULL,

[FIXDATE] [datetime] NULL,

[EXCLDATE] [datetime] NULL,

[TDYACCINT\_8] [numeric](18, 8) NULL,

[YSTACCINT\_8] [numeric](18, 8) NULL,

[PRICESOURCE\_DDE] [char](50) NULL,

[YIELDSOURCE\_DDE] [char](50) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_SCHD] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_SCHD](

[BR] [char](2) NOT NULL,

[DEALNO] [char](8) NOT NULL,

[SEQ] [char](4) NOT NULL,

[PRODUCT] [char](6) NOT NULL,

[PRODTYPE] [char](2) NOT NULL,

[INTENDDTE] [datetime] NOT NULL,

[SCHDSEQ] [char](4) NOT NULL,

[SCHDTYPE] [char](1) NOT NULL,

[CCY] [char](3) NULL,

[COMPOUND] [char](1) NULL,

[TOTPAYAMT] [numeric](19, 4) NULL,

[PRINAMT] [numeric](19, 4) NULL,

[INTAMT] [numeric](19, 4) NULL,

[POSNINTAMT] [numeric](19, 4) NULL,

[INTSTRTDTE] [datetime] NULL,

[IPAYDATE] [datetime] NULL,

[AUTOMANPAY] [char](1) NULL,

[RATEREVDTE] [datetime] NULL,

[RATEFIXDTE] [datetime] NULL,

[SMEANS] [char](7) NULL,

[SACCT] [char](15) NULL,

[RATECODE] [char](7) NULL,

[INTRATE\_8] [numeric](18, 8) NULL,

[BASIS] [char](6) NULL,

[SPREAD\_8] [numeric](18, 8) NULL,

[REVDATE] [datetime] NULL,

[VERDATE] [datetime] NULL,

[MANPAYDTE] [datetime] NULL,

[PPAYAMT] [numeric](19, 4) NULL,

[AL] [char](1) NULL,

[LSTMNTDATE] [datetime] NULL,

[BRPRCINDTE] [datetime] NULL,

[NOTPRIN] [numeric](19, 4) NULL,

[WHTRATE\_8] [numeric](18, 8) NULL,

[WHTBASIS] [char](6) NULL,

[WHTAMT] [numeric](19, 4) NULL,

[WHTIND] [char](1) NULL,

[VATAMT] [numeric](19, 4) NULL,

[VATRATE\_8] [numeric](18, 8) NULL,

[FUNDRATE\_8] [numeric](18, 8) NULL,

[FUNDBASIS] [char](6) NULL,

[FUNDAMT] [numeric](19, 4) NULL,

[PPAYDATE] [datetime] NULL,

[PAYAUTHDTE] [datetime] NULL,

[PAYAUTHIND] [char](1) NULL,

[INTCCY] [char](3) NULL,

[INTEXCHARATE\_8] [numeric](18, 8) NULL,

[TERMS] [char](1) NULL,

[POSNFUNDAMT] [numeric](19, 4) NULL,

[INTSMEANS] [char](7) NULL,

[INTSACCT] [char](15) NULL,

[INTAUTHIND] [char](1) NULL,

[INTAUTHOPER] [char](4) NULL,

[INTAUTHDATE] [datetime] NULL,

[CAPINTIND] [char](1) NULL,

[CAPINTAMT] [numeric](19, 4) NULL,

[CAPTAXAMT] [numeric](19, 4) NULL,

[APPLIEDMONTHALL] [numeric](19, 4) NULL,

[POSNPRINAMT] [numeric](19, 4) NULL,

[RATEAMTINT] [char](1) NULL,

[RATEAMTFUND] [char](1) NULL,

[RATEAMTTAX1] [char](1) NULL,

[RATEAMTTAX2] [char](1) NULL,

[RATEAMTTAX3] [char](1) NULL,

[TEXT1] [char](35) NULL,

[TEXT2] [char](35) NULL,

[DATE1] [datetime] NULL,

[DATE2] [datetime] NULL,

[NEXTAPPMONTHALL] [numeric](19, 4) NULL,

[AMOUNT2] [numeric](19, 4) NULL,

[INTOPER] [char](4) NULL,

[PRINOPER] [char](4) NULL,

[PRINAUTHOPER] [char](4) NULL,

[NETSIIND] [char](1) NULL,

[NETDEALNO] [char](8) NULL,

[NETSEQ] [char](4) NULL,

[CORPNETIND] [char](1) NULL,

[PRINUSUALID] [char](4) NULL,

[INTUSUALID] [char](4) NULL,

[AMOUNT1] [numeric](19, 4) NULL,

[RATE1] [numeric](18, 8) NULL,

[RATE2] [numeric](18, 8) NULL,

[TAXSTATUSIND] [char](1) NULL,

[STEPUPCORRRATE\_8] [numeric](18, 8) NULL,

[STEPUPCORRINTAMT] [numeric](19, 4) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_RPRH] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_RPRH](

[BR] [char](2) NOT NULL,

[DEALNO] [char](7) NOT NULL,

[SEQ] [char](3) NOT NULL,

[AETRAD] [char](4) NULL,

[ASSIGNIND] [char](1) NULL,

[ASSGNQTY] [numeric](19, 4) NULL,

[BRPRCINDTE] [datetime] NULL,

[BROK] [char](11) NULL,

[BROKFEEAMT] [numeric](19, 4) NULL,

[BROKFEECCY] [char](3) NULL,

[CHARGEAMT] [numeric](19, 4) NULL,

[COMAUTHDATE] [datetime] NULL,

[COMAUTHIND] [char](1) NULL,

[COMAUTHOPER] [char](4) NULL,

[COMCCYSACCT] [char](15) NULL,

[COMCCYSMEANS] [char](7) NULL,

[COMREFNO] [char](16) NULL,

[CONFDATE] [datetime] NULL,

[CONFOPER] [char](4) NULL,

[CNARR] [char](70) NULL,

[COST] [char](10) NULL,

[CCY] [char](3) NULL,

[CNO] [char](11) NULL,

[CUSTCONFDATE] [datetime] NULL,

[CUSTREFNO] [char](16) NULL,

[DEALDATE] [datetime] NULL,

[DEALSCRE] [char](10) NULL,

[DEALTEXT] [char](70) NULL,

[DEALTIME] [char](8) NULL,

[FEEAMT] [numeric](19, 4) NULL,

[FINFXRATE\_8] [numeric](18, 8) NULL,

[FINTERMS] [char](1) NULL,

[INITFXRATE\_8] [numeric](18, 8) NULL,

[INITTERMS] [char](1) NULL,

[INPUTDATE] [datetime] NULL,

[IOPER] [char](4) NULL,

[INPUTTIME] [char](8) NULL,

[BASIS] [char](6) NULL,

[INTCALCTYPE] [char](3) NULL,

[MARGCALLPCT\_8] [numeric](18, 8) NULL,

[MARGINPCT] [char](5) NULL,

[MATAUTHDATE] [datetime] NULL,

[MATAUTHIND] [char](1) NULL,

[MATAUTHOPER] [char](4) NULL,

[MATDATE] [datetime] NULL,

[MATCCYSACCT] [char](15) NULL,

[MATCCYSMEANS] [char](7) NULL,

[NOTAMT] [numeric](19, 4) NULL,

[NOTCCY] [char](3) NULL,

[NOTTOTLINT] [numeric](19, 4) NULL,

[OPENREPOIND] [char](1) NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[PS] [char](1) NULL,

[MQTY] [numeric](19, 4) NULL,

[QUOTQTY] [numeric](19, 4) NULL,

[REPOAMT] [numeric](19, 4) NULL,

[REPORATE\_8] [numeric](18, 8) NULL,

[REVDATE] [datetime] NULL,

[REVOPER] [char](4) NULL,

[REVREASON] [char](2) NULL,

[REVTEXT] [char](70) NULL,

[REVTIME] [char](8) NULL,

[SPREADRATE\_8] [numeric](18, 8) NULL,

[INDEXRATECODE] [char](7) NULL,

[SUBSTITUTE] [char](1) NULL,

[SUPPCONFIND] [char](1) NULL,

[TENOR] [char](2) NULL,

[TOTALINT] [numeric](19, 4) NULL,

[TRAD] [char](4) NULL,

[VDATE] [datetime] NULL,

[VATAMT] [numeric](19, 4) NULL,

[VERIND] [char](1) NULL,

[VERDATE] [datetime] NULL,

[VOPER] [char](4) NULL,

[WHTAMT] [numeric](19, 4) NULL,

[YIELD\_8] [numeric](18, 8) NULL,

[SAFEKEEPACCT] [char](15) NULL,

[RPIND] [char](1) NULL,

[TOTPROCDAMT] [numeric](19, 4) NULL,

[COMPROCDAMT] [numeric](19, 4) NULL,

[MATPROCDAMT] [numeric](19, 4) NULL,

[ORIGPROCDAMT] [numeric](19, 4) NULL,

[CLOSEREPOIND] [char](1) NULL,

[EMPENAMT] [numeric](19, 4) NULL,

[EMIOPER] [char](4) NULL,

[EMINPUTDATE] [datetime] NULL,

[ORIGMDATE] [datetime] NULL,

[INTCALCDIND] [char](1) NULL,

[PORT] [char](4) NULL,

[RATECODE] [char](7) NULL,

[COLLATERALCODE] [char](20) NULL,

[COLLUNIT] [char](3) NULL,

[DELIVTYPE] [char](3) NULL,

[COUPREINVEST] [char](1) NULL,

[COUPREINVRATE\_8] [numeric](18, 8) NULL,

[CONFCODE] [char](4) NULL,

[INITBASERATE\_8] [numeric](18, 8) NULL,

[INITBASETERMS] [char](1) NULL,

[FINBASERATE\_8] [numeric](18, 8) NULL,

[FINBASETERMS] [char](1) NULL,

[OPENREPOMATDAYS] [char](3) NULL,

[INITINDEXRATE\_8] [numeric](18, 8) NULL,

[FININDEXRATE\_8] [numeric](18, 8) NULL,

[FINEXCHRATEFIXDATE] [datetime] NULL,

[FININDXRATEFIXDATE] [datetime] NULL,

[INTRATEFIXDATE] [datetime] NULL,

[FINEXCHRATEENTDATE] [datetime] NULL,

[FININDXRATEENTDATE] [datetime] NULL,

[INTRATEENDDATE] [datetime] NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[COMPROCBASEAMT] [numeric](19, 4) NULL,

[MATPROCBASEAMT] [numeric](19, 4) NULL,

[CONVINTAMT] [numeric](19, 4) NULL,

[CONVINTBAMT] [numeric](19, 4) NULL,

[ASSIGNDATE] [datetime] NULL,

[PRINBASERATEFIXDATE] [datetime] NULL,

[PRINBASERATEENTDATE] [datetime] NULL,

[OVERRIDEWXTAXIND] [char](1) NULL,

[BORRLENDIND] [char](1) NULL,

[TEXT1] [char](35) NULL,

[TEXT2] [char](35) NULL,

[DATE1] [datetime] NULL,

[DATE2] [datetime] NULL,

[AMOUNT1] [numeric](19, 4) NULL,

[AMOUNT2] [numeric](19, 4) NULL,

[RATE1\_8] [numeric](18, 8) NULL,

[RATE2\_8] [numeric](18, 8) NULL,

[COMMAMT] [numeric](19, 4) NULL,

[EXCHCOMMAMT] [numeric](19, 4) NULL,

[STAMPDUTYAMT] [numeric](19, 4) NULL,

[EXCHCNO] [char](11) NULL,

[EXCHIND] [char](1) NULL,

[BTBDEALNO] [char](7) NULL,

[COMMUSUALID] [char](15) NULL,

[MATUSUALID] [char](15) NULL,

[SWIFTBOTHIND] [char](1) NULL,

[LINKDEALNO] [char](8) NULL,

[LINKIND] [char](3) NULL,

[BLOCKIND] [char](1) NULL,

[SPLITIND] [char](1) NULL,

[CCPIND] [char](1) NULL,

[EXTCOMPARE] [char](1) NULL,

[VDATENET] [char](1) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_RPDT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_RPDT](

[BR] [char](2) NOT NULL,

[DEALNO] [char](7) NOT NULL,

[SEQ] [char](3) NOT NULL,

[ASSIGNSEQ] [char](3) NOT NULL,

[BRPRCINDTE] [datetime] NULL,

[CHARGEAMT] [numeric](19, 4) NULL,

[CLPRICE\_8] [numeric](18, 8) NULL,

[CCY] [char](3) NULL,

[DEALTIME] [char](8) NULL,

[DRPRICE\_8] [numeric](18, 8) NULL,

[DISCAMT] [numeric](19, 4) NULL,

[DISCRATE\_8] [numeric](18, 8) NULL,

[DISPRICE\_8] [numeric](18, 8) NULL,

[EXCHCNO] [char](10) NULL,

[EXCHCOMMAMT] [numeric](19, 4) NULL,

[EXCHCCY] [char](3) NULL,

[FACEAMT] [numeric](19, 4) NULL,

[INPUTDATE] [datetime] NULL,

[IOPER] [char](4) NULL,

[INPUTTIME] [char](8) NULL,

[INVTYPE] [char](1) NULL,

[MDATE] [datetime] NULL,

[PORT] [char](4) NULL,

[PREMAMT] [numeric](19, 4) NULL,

[PRINAMT] [numeric](19, 4) NULL,

[PROCEEDAMT] [numeric](19, 4) NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[PS] [char](1) NULL,

[PURCHINTAMT] [numeric](19, 4) NULL,

[REPOINTAMT] [numeric](19, 4) NULL,

[RPOSMDATE] [datetime] NULL,

[QTY] [numeric](19, 4) NULL,

[SECID] [char](20) NULL,

[VATAMT] [numeric](19, 4) NULL,

[VDATE] [datetime] NULL,

[YIELD\_8] [numeric](18, 8) NULL,

[WHTAMT] [numeric](19, 4) NULL,

[CPNPAIDAMT] [numeric](19, 4) NULL,

[NXTCPNAMT] [numeric](19, 4) NULL,

[TDYMTMAMT] [numeric](19, 4) NULL,

[YSTMTMAMT] [numeric](19, 4) NULL,

[PMEMTMAMT] [numeric](19, 4) NULL,

[PYEMTMAMT] [numeric](19, 4) NULL,

[TDYINCEXPAMT] [numeric](19, 4) NULL,

[YSTINCEXPAMT] [numeric](19, 4) NULL,

[PMEINCEXPAMT] [numeric](19, 4) NULL,

[PYEINCEXPAMT] [numeric](19, 4) NULL,

[TDYDSCAMT] [numeric](19, 4) NULL,

[YSTDSCAMT] [numeric](19, 4) NULL,

[PMEDSCAMT] [numeric](19, 4) NULL,

[PYEDSCAMT] [numeric](19, 4) NULL,

[TDYPREMAMT] [numeric](19, 4) NULL,

[YSTPREMAMT] [numeric](19, 4) NULL,

[COUPACCROUTSTAMT] [numeric](19, 4) NULL,

[COUPACCRNEXTMONTH] [numeric](19, 4) NULL,

[SETTAVGCOST] [numeric](19, 4) NULL,

[SETTAVGCOSTYST] [numeric](19, 4) NULL,

[BOOKVALAMT] [numeric](19, 4) NULL,

[BOOKVALAMTYST] [numeric](19, 4) NULL,

[DELIVTYPE] [char](3) NULL,

[CNO] [char](10) NULL,

[REPOCOST] [char](10) NULL,

[BTBPORT] [char](4) NULL,

[VERDATE] [datetime] NULL,

[REVDATE] [datetime] NULL,

[PRINEXCHRATE\_8] [numeric](18, 8) NULL,

[INTEXCHRATE\_8] [numeric](18, 8) NULL,

[PRINBASERATE\_8] [numeric](18, 8) NULL,

[INTBASERATE\_8] [numeric](18, 8) NULL,

[PRINEXCHTERMS] [char](1) NULL,

[INTEXCHTERMS] [char](1) NULL,

[PRINBASETERMS] [char](1) NULL,

[INTBASETERMS] [char](1) NULL,

[HAIRCUT\_8] [numeric](18, 8) NULL,

[VDFAILDATE] [datetime] NULL,

[MDFAILDATE] [datetime] NULL,

[VDCLEARDATE] [datetime] NULL,

[MDCLEARDATE] [datetime] NULL,

[VDFAILOPER] [char](4) NULL,

[MDFAILOPER] [char](4) NULL,

[VDCLEAROPER] [char](4) NULL,

[MDCLEAROPER] [char](4) NULL,

[VDFAILREFNO] [char](7) NULL,

[MDFAILREFNO] [char](7) NULL,

[COLLUNIT] [char](3) NULL,

[VDCLEANPRICE\_8] [numeric](18, 8) NULL,

[VDDIRTYPRICE\_8] [numeric](18, 8) NULL,

[MDCLEANPRICE\_8] [numeric](18, 8) NULL,

[MDDIRTYPRICE\_8] [numeric](18, 8) NULL,

[NOTCCY] [char](3) NULL,

[COMMPROCAMT] [numeric](19, 4) NULL,

[MATPROCAMT] [numeric](19, 4) NULL,

[MATPURCHINTAMT] [numeric](19, 4) NULL,

[REINVESTIND] [char](1) NULL,

[REINVESTLINKSEQNO] [char](3) NULL,

[MKTVALUE] [numeric](19, 4) NULL,

[UNAMORTAMT] [numeric](19, 4) NULL,

[TDYREPOINTAMT] [numeric](19, 4) NULL,

[YSTREPOINTAMT] [numeric](19, 4) NULL,

[PMEREPOINTAMT] [numeric](19, 4) NULL,

[PYEREPOINTAMT] [numeric](19, 4) NULL,

[TDYREPOMTMAMT] [numeric](19, 4) NULL,

[YSTREPOMTMAMT] [numeric](19, 4) NULL,

[PMEREPOMTMAMT] [numeric](19, 4) NULL,

[PYEREPOMTMAMT] [numeric](19, 4) NULL,

[PVMATPROCAMT] [numeric](19, 4) NULL,

[REPORATE\_8] [numeric](18, 8) NULL,

[SPREADRATE\_8] [numeric](18, 8) NULL,

[HCPURCHINTAMT] [numeric](19, 4) NULL,

[HCMATPURCHINTAMT] [numeric](19, 4) NULL,

[REPOPRINAMT] [numeric](19, 4) NULL,

[COST] [char](10) NULL,

[CONVINTAMT] [numeric](19, 4) NULL,

[CONVINTBAMT] [numeric](19, 4) NULL,

[SETTCCY] [char](3) NULL,

[BORRLENDIND] [char](1) NULL,

[TDYPURCHINTAMT] [numeric](19, 4) NULL,

[YSTPURCHINTAMT] [numeric](19, 4) NULL,

[TEXT1] [char](35) NULL,

[TEXT2] [char](35) NULL,

[DATE1] [datetime] NULL,

[DATE2] [datetime] NULL,

[AMOUNT1] [numeric](19, 4) NULL,

[AMOUNT2] [numeric](19, 4) NULL,

[RATE1\_8] [numeric](18, 8) NULL,

[RATE2\_8] [numeric](18, 8) NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[VDPAIREDFACEAMT] [numeric](19, 4) NULL,

[VDPAIREDAMT] [numeric](19, 4) NULL,

[VDPAIREDIND] [char](1) NULL,

[VDCURRSPLITSEQ] [char](4) NULL,

[MDPAIREDFACEAMT] [numeric](19, 4) NULL,

[MDPAIREDAMT] [numeric](19, 4) NULL,

[MDPAIREDIND] [char](1) NULL,

[MDCURRSPLITSEQ] [char](4) NULL,

[INDEXRATIO\_8] [numeric](18, 8) NULL,

[NOVFACEAMT] [numeric](19, 4) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_FXDH\_REVAL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_FXDH\_REVAL](

[BR] [char](2) NOT NULL,

[DEALNO] [char](8) NOT NULL,

[SEQ] [char](4) NOT NULL,

[PS] [char](1) NULL,

[PRODCODE] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[CCY] [char](3) NULL,

[CTRCCY] [char](3) NULL,

[VDATE] [datetime] NULL,

[CCYSETTDATE] [datetime] NULL,

[CTRSETTDATE] [datetime] NULL,

[CCYAMT] [numeric](19, 4) NOT NULL,

[CTRAMT] [numeric](19, 4) NOT NULL,

[CCYREVALAMT] [numeric](19, 4) NOT NULL,

[CTRREVALAMT] [numeric](19, 4) NOT NULL,

[CCYNPVAMT] [numeric](19, 4) NOT NULL,

[CTRNPVAMT] [numeric](19, 4) NOT NULL,

[NPVRCCY1] [char](3) NULL,

[NPVRCCY2] [char](3) NULL,

[NPVRVDATE] [datetime] NULL,

[DEALCCYNPVAMT] [numeric](19, 4) NULL,

[DEALCTRNPVAMT] [numeric](19, 4) NULL,

[CCYSPOTRATE] [float] NULL,

[CTRSPOTRATE] [float] NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_FXDH] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_FXDH](

[BR] [char](2) NOT NULL,

[DEALNO] [char](8) NOT NULL,

[SEQ] [char](4) NOT NULL,

[TRAD] [char](4) NULL,

[VDATE] [datetime] NULL,

[CUST] [char](11) NULL,

[BROK] [char](11) NULL,

[BROKCCY] [char](3) NULL,

[BROKAMT] [numeric](19, 4) NULL,

[PHONCI] [char](1) NULL,

[PORT] [char](4) NULL,

[COST] [char](11) NULL,

[ODATE] [datetime] NULL,

[DEALDATE] [datetime] NULL,

[DEALTIME] [char](8) NULL,

[IOPER] [char](4) NULL,

[VOPER] [char](4) NULL,

[BROKCDATE] [datetime] NULL,

[CUSTCDATE] [datetime] NULL,

[PHONECDATE] [datetime] NULL,

[DEALTEXT] [char](70) NULL,

[PHONETEXT] [char](70) NULL,

[PS] [char](1) NULL,

[PRODCODE] [char](6) NULL,

[VERIND] [char](1) NULL,

[REVTEXT] [char](70) NULL,

[SWAPDEAL] [char](8) NULL,

[FARNEARIND] [char](1) NULL,

[PAYAUTHIND] [char](1) NULL,

[SWAPVDATE] [datetime] NULL,

[REVREASON] [char](2) NULL,

[INPUTDATE] [datetime] NULL,

[INPUTTIME] [char](8) NULL,

[DEALSRCE] [char](10) NULL,

[ORIGOCAMT] [numeric](19, 4) NULL,

[TRCEDATE] [datetime] NULL,

[TRACECNT] [char](3) NULL,

[PAYOPER] [char](4) NULL,

[LSTMNTDATE] [datetime] NULL,

[CCY] [char](3) NULL,

[CCYAMT] [numeric](19, 4) NULL,

[CCYTERMS] [char](1) NULL,

[CCYRATE\_8] [numeric](18, 8) NULL,

[CCYPD\_8] [numeric](18, 8) NULL,

[CCYBAMT] [numeric](19, 4) NULL,

[CCYBRATE\_8] [numeric](18, 8) NULL,

[CCYBPD\_8] [numeric](18, 8) NULL,

[BASETERMS] [char](1) NULL,

[NETSI] [char](1) NULL,

[SPOTFWDIND] [char](1) NULL,

[NETDEAL] [char](11) NULL,

[CCYSMEANS] [char](7) NULL,

[CCYSACCT] [char](15) NULL,

[CTRSMEANS] [char](7) NULL,

[CTRSACCT] [char](15) NULL,

[CTRCCY] [char](3) NULL,

[CTRAMT] [numeric](19, 4) NULL,

[CTRRATE\_8] [numeric](18, 8) NULL,

[CTRPD\_8] [numeric](18, 8) NULL,

[CTRBAMT] [numeric](19, 4) NULL,

[CTRBRATE\_8] [numeric](18, 8) NULL,

[NSRNOCCY] [char](8) NULL,

[NSRCCYSEQ] [char](4) NULL,

[NSRNOCTR] [char](8) NULL,

[NSRCTRSEQ] [char](4) NULL,

[BRPRCINDTE] [datetime] NULL,

[RECOPER] [char](4) NULL,

[PAYAUTHDTE] [datetime] NULL,

[RECAUTHDTE] [datetime] NULL,

[VERDATE] [datetime] NULL,

[REVDATE] [datetime] NULL,

[CCYREVALAMT] [numeric](19, 4) NULL,

[CTRREVALAMT] [numeric](19, 4) NULL,

[CCYNPVAMT] [numeric](19, 4) NULL,

[CTRNPVAMT] [numeric](19, 4) NULL,

[RECAUTHOPER] [char](4) NULL,

[PAYAUTHOPER] [char](4) NULL,

[SIIND] [char](1) NULL,

[PRODTYPE] [char](2) NULL,

[SPOTTRAD] [char](4) NULL,

[SPOTPORT] [char](4) NULL,

[SPOTCOST] [char](10) NULL,

[INTERNALRATE\_8] [numeric](18, 8) NULL,

[CORPSPREAD\_8] [numeric](18, 8) NULL,

[CORPCOST] [char](10) NULL,

[BRSPREAD\_8] [numeric](18, 8) NULL,

[BRCOST] [char](10) NULL,

[TENOR] [char](2) NULL,

[CCYSETTDATE] [datetime] NULL,

[CTRSETTDATE] [datetime] NULL,

[CUSTENTEREDIND] [char](1) NULL,

[LINKDEALNO] [char](8) NULL,

[LINKPRODUCT] [char](6) NULL,

[LINKPRODTYPE] [char](2) NULL,

[FINCNTR1] [char](10) NULL,

[FINCNTR2] [char](10) NULL,

[COMMAMT] [numeric](19, 4) NULL,

[EXCOVERALLLIM] [char](1) NULL,

[EXCCUSTLIM] [char](1) NULL,

[CORPCCY] [char](3) NULL,

[CORPPORT] [char](4) NULL,

[CORPTRAD] [char](4) NULL,

[REVDEALNO] [char](8) NULL,

[OPTIONIND] [char](1) NULL,

[MARGINAMT] [numeric](19, 4) NULL,

[MARGINCCY] [char](3) NULL,

[CORPSPREADAMT] [numeric](19, 4) NULL,

[SPOTCCY] [char](3) NULL,

[SPOTIND] [char](1) NULL,

[TENOREXCEEDIND] [char](1) NULL,

[CCYSETTSTATUS] [char](1) NULL,

[CTRSETTSTATUS] [char](1) NULL,

[CUSTREFNO] [char](35) NULL,

[CCYFAILDATE] [datetime] NULL,

[CCYCLEARDATE] [datetime] NULL,

[CCYFAILOPER] [char](4) NULL,

[CCYCLEAROPER] [char](4) NULL,

[CTRFAILDATE] [datetime] NULL,

[CTRCLEARDATE] [datetime] NULL,

[CTRFAILOPER] [char](4) NULL,

[CTRCLEAROPER] [char](4) NULL,

[MARGINFEEAMT] [numeric](19, 4) NULL,

[MARGINFEECCY] [char](3) NULL,

[MARGINFEESETTMEANS] [char](7) NULL,

[MARGINFEESETTACCT] [char](15) NULL,

[MARGINFEEREBAMT] [numeric](19, 4) NULL,

[MARGINFEEREBCCY] [char](3) NULL,

[MARGINFEEREBSETTMEANS] [char](7) NULL,

[MARGINFEEREBSETTACCT] [char](15) NULL,

[PHONECONFOPER] [char](4) NULL,

[PHONECONFTIME] [char](8) NULL,

[SWIFTMATCHIND] [char](1) NULL,

[SIINDPAY] [char](1) NULL,

[COMMCCY] [char](3) NULL,

[REXRATETOLIND] [char](1) NULL,

[DEALNOEXCESS] [char](8) NULL,

[FIXRATEIND] [char](1) NULL,

[FIXRATECODE] [char](7) NULL,

[REVOPER] [char](4) NULL,

[CTRPARTYDEALNO] [char](8) NULL,

[CTRPARTYPHONENO] [char](20) NULL,

[CTRPARTYPHONEOPER] [char](4) NULL,

[BLOCKIND] [char](1) NULL,

[CONFCREATEDIND] [char](1) NULL,

[SPOTDATE] [datetime] NULL,

[DISAGGCCY] [char](3) NULL,

[DISAGGSPOTTRAD] [char](4) NULL,

[DISAGGSPOTPORT] [char](4) NULL,

[DISAGGSPOTCOST] [char](10) NULL,

[DISAGGFWDTRAD] [char](4) NULL,

[DISAGGFWDPORT] [char](4) NULL,

[DISAGGFWDCOST] [char](10) NULL,

[FWDPREMAMT] [numeric](19, 4) NULL,

[AMENDDATE] [datetime] NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_EXPT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_EXPT](

[BR] [char](2) NOT NULL,

[PRODCODE] [char](6) NOT NULL,

[TYPE] [char](2) NOT NULL,

[ACCTNGTYPE] [char](10) NOT NULL,

[TENOR] [char](2) NOT NULL,

[SEQ] [char](4) NOT NULL,

[DESCR] [char](35) NULL,

[GLNO] [char](15) NULL,

[LSTMNTDTE] [datetime] NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_DLDT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_DLDT](

[BR] [char](2) NOT NULL,

[DEALNO] [char](8) NOT NULL,

[SEQ] [char](4) NOT NULL,

[PRODUCT] [char](6) NULL,

[PRODTYPE] [char](2) NULL,

[TRAD] [char](4) NULL,

[VDATE] [datetime] NULL,

[MDATE] [datetime] NULL,

[CNO] [char](10) NULL,

[CCY] [char](3) NULL,

[CCYAMT] [numeric](19, 4) NULL,

[ORIGOCAMT] [numeric](19, 4) NULL,

[AL] [char](1) NULL,

[RATECODE] [char](7) NULL,

[BASIS] [char](6) NULL,

[INTRATE] [char](16) NULL,

[SPREAD\_8] [numeric](18, 8) NULL,

[ACRFRSTLST] [char](1) NULL,

[BROK] [char](11) NULL,

[BROKCCY] [char](3) NULL,

[BROKAMT] [numeric](19, 4) NULL,

[BROKRATE\_8] [numeric](18, 8) NULL,

[PORT] [char](4) NULL,

[COST] [char](11) NULL,

[DEALDATE] [datetime] NULL,

[DEALTIME] [char](8) NULL,

[IOPER] [char](4) NULL,

[VOPER] [char](4) NULL,

[GUARANTOR] [char](10) NULL,

[CUSTCDATE] [datetime] NULL,

[VERIND] [char](1) NULL,

[VERDATE] [datetime] NULL,

[REVREASON] [char](2) NULL,

[REVDATE] [datetime] NULL,

[REVTEXT] [char](70) NULL,

[PAYAUTHIND] [char](1) NULL,

[PAYOPER] [char](4) NULL,

[PAYAUTHDTE] [datetime] NULL,

[RECOPER] [char](4) NULL,

[RECAUTHDTE] [datetime] NULL,

[INPUTDATE] [datetime] NULL,

[INPUTTIME] [char](8) NULL,

[DEALSRCE] [char](10) NULL,

[ARBDEALNO] [char](10) NULL,

[TRCEDATE] [datetime] NULL,

[TRACECNT] [char](3) NULL,

[COMMMEANS] [char](7) NULL,

[COMMACCT] [char](15) NULL,

[MATMEANS] [char](7) NULL,

[MATACCT] [char](15) NULL,

[BRPRCINDTE] [datetime] NULL,

[DEALTEXT] [char](70) NULL,

[LSTMNTDATE] [datetime] NULL,

[RATEREVDTE] [datetime] NULL,

[NXTRATEREV] [datetime] NULL,

[SCHDFLAG] [char](1) NULL,

[NEWDEALNO] [char](8) NULL,

[ORIGDEALNO] [char](8) NULL,

[INTCAPFLAG] [char](1) NULL,

[INTCAPAMT] [numeric](19, 4) NULL,

[LIENAMT] [numeric](19, 4) NULL,

[LIENOFFCR] [char](8) NULL,

[LIENDATE] [datetime] NULL,

[PRMDSCAMRTD] [numeric](19, 4) NULL,

[EMPENAMT] [numeric](19, 4) NULL,

[EMIOPER] [char](4) NULL,

[EMINPUTDATE] [datetime] NULL,

[ORIGMDATE] [datetime] NULL,

[TENOR] [char](2) NULL,

[FUNDRATE\_8] [numeric](18, 8) NULL,

[FUNDBASIS] [char](6) NULL,

[FUNDCOST] [char](11) NULL,

[DISCRATE\_8] [numeric](18, 8) NULL,

[FIRSTDEALNO] [char](8) NULL,

[FIXRULE] [char](1) NULL,

[INTCALCRULE] [char](1) NULL,

[FIRSTIPAYDATE] [datetime] NULL,

[LASTIPAYDATE] [datetime] NULL,

[INTPAYCYCLE] [char](2) NULL,

[PRINPAYCYCLE] [char](2) NULL,

[INTPAYDAY] [char](3) NULL,

[PRINPAYDAY] [char](3) NULL,

[YIELD\_8] [numeric](18, 8) NULL,

[INTSMEANS] [char](7) NULL,

[FINCAL] [char](15) NULL,

[SIIND] [char](1) NULL,

[INTSETTAUTHIND] [char](1) NULL,

[INTAUTHOPER] [char](4) NULL,

[INTAUTHDATE] [datetime] NULL,

[SECID] [char](20) NULL,

[CORPTRAD] [char](4) NULL,

[CORPPORT] [char](4) NULL,

[AGENTIND] [char](1) NULL,

[UPDATECOUNTER] [numeric](15, 0) NULL,

[TEXT1] [char](35) NULL,

[TEXT2] [char](35) NULL,

[DATE1] [datetime] NULL,

[DATE2] [datetime] NULL,

[AMOUNT1] [numeric](19, 4) NULL,

[AMOUNT2] [numeric](19, 4) NULL,

[MATSTATUS] [char](15) NULL,

[REVDEALNO] [char](8) NULL,

[BTBDEALNO] [char](8) NULL,

[AMENDDATE] [datetime] NULL,

[CUSTREFNO] [char](35) NULL,

[INTRATE\_8] [numeric](18, 8) NULL,

[REVOPER] [char](4) NULL,

[PAYAUTHOPER] [char](4) NULL,

[RECAUTHOPER] [char](4) NULL,

[BTBTRAD] [char](4) NULL,

[BTBPORT] [char](4) NULL,

[BTBCOST] [char](10) NULL,

[LINKDEALNO] [char](8) NULL,

[LINKPRODUCT] [char](6) NULL,

[LINKPRODTYPE] [char](2) NULL,

[LINKSEQ] [char](4) NULL,

[SIPAYIND] [char](1) NULL,

[SWIFTMATCHIND] [char](1) NULL,

[INTSACCT] [char](15) NULL,

[COMMUSUALID] [char](15) NULL,

[MATUSUALID] [char](15) NULL,

[COMMTYPECODE] [char](1) NULL,

[MATTYPECODE] [char](1) NULL,

[SCHEDULETYPE] [char](2) NULL,

[PRINDATERULE] [char](3) NULL,

[PRINDATEDIR] [char](2) NULL,

[PRINTOTPAYMENTS] [numeric](19, 4) NULL,

[INTDATERULE] [char](3) NULL,

[INTDATEDIR] [char](2) NULL,

[LASTAMOUNT] [numeric](19, 4) NULL,

[BTBIND] [char](1) NULL,

[BTBBRANCH] [char](2) NULL,

[BTBSEQ] [char](4) NULL,

[BTBPRODUCT] [char](6) NULL,

[BTBPRODTYPE] [char](2) NULL,

[BTBFUNDCENTER] [char](10) NULL,

[DELAYEDINTIND] [char](1) NULL,

[DELAYEDINTDATE] [datetime] NULL,

[EARLYMATIND] [char](1) NULL,

[BROKEXCHRATE\_8] [numeric](18, 8) NULL,

[PRINSIPAYIND] [char](1) NULL,

[PRINSIRECIND] [char](1) NULL,

[INTSIPAYIND] [char](1) NULL,

[INTSIRECIND] [char](1) NULL,

[NETSI] [char](1) NULL,

[NETDEALNO] [char](8) NULL,

[NETSEQ] [char](4) NULL,

[CORPNETIND] [char](1) NULL,

[TAXCHGDESC] [char](25) NULL,

[TAXAMT] [numeric](19, 4) NULL,

[TAXAUTHORITY] [char](10) NULL,

[TAXPRODUCT] [char](6) NULL,

[TAXPRODTYPE] [char](2) NULL,

[TAXTYPE] [char](1) NULL,

[ACCTGMETHOD] [char](1) NULL,

[FUNDPORT] [char](4) NULL,

[FUNDTRAD] [char](4) NULL,

[BTBCUSTOMER] [char](10) NULL,

[BTBCOMMUSUALID] [char](15) NULL,

[BTBMATUSUALID] [char](15) NULL,

[CORPDEALNO] [char](8) NULL,

[CORPSEQ] [char](4) NULL,

[STEPUPRATE\_8] [numeric](18, 8) NULL,

[FEDEALNO] [char](15) NULL,

[SERVER] [char](15) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_OPICS\_CUST] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_OPICS\_CUST](

[CNO] [char](11) NOT NULL,

[CMNE] [char](10) NOT NULL,

[BIC] [char](11) NULL,

[CCODE] [char](2) NULL,

[SIC] [char](10) NULL,

[SN] [char](40) NULL,

[CFN1] [char](35) NULL,

[CFN2] [char](35) NULL,

[CA1] [char](35) NULL,

[CA2] [char](35) NULL,

[CA3] [char](35) NULL,

[CA4] [char](35) NULL,

[CA5] [char](35) NULL,

[CTYPE] [char](1) NULL,

[CPOST] [char](10) NULL,

[LSTMNTDTE] [datetime] NULL,

[UCCODE] [char](2) NULL,

[BIRTHDATE] [datetime] NULL,

[TAXID] [char](15) NULL,

[ACCTNGTYPE] [char](10) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_TableFilters\_NB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_TableFilters\_NB](

[FilterID] [int] IDENTITY(1,1) NOT NULL,

[Portfolio] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[FilterName] [varchar](255) NULL,

[Module] [varchar](255) NULL,

[Filter] [varchar](max) NULL,

[Default\_Filter] [varchar](50) NULL,

[Table\_Source] [varchar](max) NULL,

[Comments] [varchar](max) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_TableFilters] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_TableFilters](

[FilterID] [int] IDENTITY(1,1) NOT NULL,

[Portfolio] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[FilterName] [varchar](255) NULL,

[Module] [varchar](255) NULL,

[Filter] [varchar](max) NULL,

[Default\_Filter] [varchar](50) NULL,

[Table\_Source] [varchar](max) NULL,

[Comments] [varchar](max) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL,

CONSTRAINT [PK\_US\_FR2052A\_REF\_TableFilters] PRIMARY KEY CLUSTERED

(

[FilterID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_TableDetails\_NB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_TableDetails\_NB](

[Ref\_table\_Details\_ID] [int] IDENTITY(1,1) NOT NULL,

[PID] [varchar](20) NULL,

[Portfolio] [varchar](255) NULL,

[SourceTableName] [varchar](255) NULL,

[TablePrimaryKeyName] [varchar](255) NULL,

[DatabaseName] [varchar](255) NULL,

[TargetTableName] [varchar](255) NULL,

[TargetTablePrimaryKeyName] [varchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_TableDetails] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_TableDetails](

[Ref\_table\_Details\_ID] [int] IDENTITY(1,1) NOT NULL,

[PID] [varchar](20) NULL,

[Portfolio] [varchar](255) NULL,

[SourceTableName] [varchar](255) NULL,

[TablePrimaryKeyName] [varchar](255) NULL,

[DatabaseName] [varchar](255) NULL,

[TargetTableName] [varchar](255) NULL,

[TargetTablePrimaryKeyName] [varchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_RequiredTables] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_RequiredTables](

[ID] [int] IDENTITY(1,1) NOT NULL,

[Store Sprocedure] [nvarchar](255) NULL,

[PID ] [nvarchar](255) NULL,

[PID\_Type] [nvarchar](255) NULL,

[Portfolio] [nvarchar](255) NULL,

[Database Source] [nvarchar](255) NULL,

[Table Name ] [nvarchar](255) NULL,

[Schema name] [nvarchar](255) NULL,

[Comments] [nvarchar](255) NULL,

[Date Completed] [datetime] NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_ReportType] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_ReportType](

[ReportID] [int] IDENTITY(1,1) NOT NULL,

[ReportType] [nvarchar](255) NULL,

[PID] [nvarchar](255) NULL,

[Product] [nvarchar](500) NULL,

[Product\_Old] [nvarchar](500) NULL,

[SID] [varchar](2) NULL,

[SubProduct] [varchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_ReportType] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_ReportType](

[ReportType] [nvarchar](255) NULL,

[PID] [nvarchar](255) NULL,

[Product] [nvarchar](500) NULL,

[Product\_Old] [nvarchar](500) NULL,

[SID] [varchar](2) NULL,

[SubProduct] [varchar](255) NULL,

[INPUTTER] [varchar](50) NULL,

[AUTHORISER] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_OperationalDepositTag] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_OperationalDepositTag](

[TB\_LINE] [varchar](4) NOT NULL,

[CONTACT\_ID] [varchar](8) NOT NULL,

[CURRENCY] [varchar](3) NOT NULL,

[DEAL\_BALANCE] [decimal](19, 2) NOT NULL,

[OPERATIONAL\_DEPOSIT\_RATIO] [decimal](19, 6) NOT NULL,

[OPERATIONAL\_DEPOSIT\_BAL] [decimal](19, 2) NOT NULL,

[NON\_OPERATIONAL\_DEPOSIT\_BAL] [decimal](19, 2) NULL,

[ASOFDATE] [datetime] NOT NULL,

[INPUTTER] [varchar](255) NULL,

[AUTHORIZER] [varchar](255) NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_OperationalDepositTag] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_OperationalDepositTag](

[TB\_LINE] [varchar](4) NOT NULL,

[CONTACT\_ID] [varchar](8) NOT NULL,

[CURRENCY] [varchar](3) NOT NULL,

[DEAL\_BALANCE] [decimal](19, 2) NOT NULL,

[OPERATIONAL\_DEPOSIT\_RATIO] [decimal](19, 6) NOT NULL,

[OPERATIONAL\_DEPOSIT\_BAL] [decimal](19, 2) NOT NULL,

[NON\_OPERATIONAL\_DEPOSIT\_BAL] [decimal](19, 2) NULL,

[INPUTTER] [varchar](255) NULL,

[AUTHORIZER] [varchar](255) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_InternalCounterPartySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_InternalCounterPartySTB](

[CNO] [varchar](10) NOT NULL,

[SN] [varchar](50) NULL,

[CCODE] [varchar](2) NULL,

[ACCTINGTYPE] [varchar](35) NULL,

[Internal] [varchar](1) NOT NULL,

[InternalCounterParty] [varchar](100) NOT NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[comments] [varchar](500) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_InternalCounterPartySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_InternalCounterPartySTB](

[cno] [varchar](10) NOT NULL,

[sn] [varchar](50) NULL,

[ccode] [varchar](2) NULL,

[ACCTINGTYPE] [varchar](35) NULL,

[Internal] [varchar](1) NOT NULL,

[InternalCounterparty] [varchar](100) NOT NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[Comments] [varchar](500) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_InternalCounterParty] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_InternalCounterParty](

[CUSTOMER\_ID] [varchar](50) NOT NULL,

[SHORT\_NAME] [varchar](100) NULL,

[Sector] [varchar](4) NOT NULL,

[Description] [varchar](100) NULL,

[INTERNAL] [varchar](1) NULL,

[InternalCounterparty] [varchar](100) NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[Comments] [varchar](1000) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_InternalCounterParty] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_InternalCounterParty](

[Customer\_ID] [varchar](50) NOT NULL,

[Short\_name] [varchar](100) NULL,

[Sector] [varchar](4) NOT NULL,

[Description] [varchar](100) NULL,

[Internal] [varchar](1) NULL,

[InternalCounterparty] [varchar](100) NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[Comments] [varchar](1000) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_9] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_9](

[Validation\_ID] [int] IDENTITY(1,1) NOT NULL,

[Table] [nvarchar](255) NULL,

[Product] [nvarchar](255) NULL,

[ProductDescription] [nvarchar](255) NULL,

[XML\_ProductDescription] [nvarchar](255) NULL,

[Type\_Bank] [nvarchar](255) NULL,

[BankType] [nvarchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsofDate] [date] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_6] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_6](

[ID ] [int] NULL,

[TYPE] [nvarchar](255) NULL,

[Table] [nvarchar](255) NULL,

[Product] [nvarchar](255) NULL,

[Product Description] [nvarchar](255) NULL,

[Inter-affiliate Aggregation] [nvarchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsofDate] [date] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_13] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_13](

[ReportType] [nvarchar](255) NULL,

[PID] [nvarchar](255) NULL,

[Product] [nvarchar](255) NULL,

[Required] [float] NULL,

[Dependent] [float] NULL,

[Not Applica] [float] NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsofDate] [date] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_12] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_12](

[Valid12\_ID] [int] IDENTITY(1,1) NOT NULL,

[GroupID] [float] NULL,

[report\_type] [nvarchar](255) NULL,

[Product] [nvarchar](255) NULL,

[PID\_No] [float] NULL,

[PID] [nvarchar](255) NULL,

[CounterParty\_Values] [nvarchar](255) NULL,

[Applicable] [varchar](25) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsofDate] [date] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_EntitySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_EntitySTB](

[BranchID] [varchar](2) NOT NULL,

[BranchName] [varchar](50) NULL,

[ReportingEntityDescription] [varchar](250) NOT NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_EntitySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_EntitySTB](

[BranchID] [varchar](2) NOT NULL,

[BranchName] [varchar](50) NULL,

[ReportingEntityDescription] [varchar](250) NOT NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_Entity] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_Entity](

[Company\_code] [varchar](9) NOT NULL,

[Company\_name] [varchar](100) NULL,

[ReportingEntityDescription] [varchar](100) NOT NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_Entity] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_Entity](

[Company\_code] [varchar](9) NOT NULL,

[Company\_name] [varchar](100) NULL,

[ReportingEntityDescription] [varchar](100) NOT NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_DepositScope] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_DepositScope](

[ReportScopeID] [int] IDENTITY(1,1) NOT NULL,

[DESC1] [nvarchar](255) NULL,

[DepositDescription] [nvarchar](255) NULL,

[Step2] [nvarchar](255) NULL,

[OD1\_Check] [nvarchar](255) NULL,

[OD1\_2\_Check] [nvarchar](255) NULL,

[WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16] [nvarchar](255) NULL,

[OD4\_5] [nvarchar](255) NULL,

[Interestpayable] [nvarchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NULL,

[ReportingEntity] [varchar](100) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_DepositScope] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_DepositScope](

[ReportScopeID] [int] NOT NULL,

[DESC1] [nvarchar](255) NULL,

[DepositDescription] [nvarchar](255) NULL,

[Step2] [nvarchar](255) NULL,

[OD1\_Check] [nvarchar](255) NULL,

[OD1\_2\_Check] [nvarchar](255) NULL,

[WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16] [nvarchar](255) NULL,

[OD4\_5] [nvarchar](255) NULL,

[Interestpayable] [nvarchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](100) NULL,

[Asofdate] [date] NULL,

[Orig\_AsOfDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_CounterPartySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_CounterPartySTB](

[Ctype] [varchar](1) NOT NULL,

[CustomerType] [varchar](35) NOT NULL,

[CID] [varchar](3) NOT NULL,

[CounterParty] [varchar](100) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](100) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_CounterPartySTB] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_CounterPartySTB](

[Ctype] [varchar](1) NOT NULL,

[CustomerType] [varchar](35) NOT NULL,

[CID] [varchar](3) NOT NULL,

[CounterParty] [varchar](100) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](100) NOT NULL,

[Asofdate] [date] NULL,

[Orig\_AsOfDate] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_CounterParty] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_CounterParty](

[Sector] [varchar](4) NOT NULL,

[Description] [varchar](100) NULL,

[CID] [varchar](3) NULL,

[CounterParty] [varchar](50) NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[Comments] [varchar](1000) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[Asofdate] [datetime] NOT NULL,

[ReportingEntity] [varchar](50) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_CounterParty] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_CounterParty](

[Sector] [varchar](4) NOT NULL,

[Description] [varchar](100) NULL,

[CID] [varchar](3) NULL,

[CounterParty] [varchar](50) NULL,

[ExceptionLogicNeeded] [varchar](500) NULL,

[Comments] [varchar](1000) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](50) NOT NULL,

[Asofdate] [date] NULL,

[Orig\_AsOfDate] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_Comments] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_Comments](

[Comment\_id] [int] IDENTITY(1,1) NOT NULL,

[ReportScope] [varchar](50) NULL,

[Type] [varchar](50) NULL,

[Product] [varchar](255) NULL,

[Subproduct] [varchar](255) NULL,

[Text] [varchar](max) NULL,

[PID] [varchar](50) NULL,

[SID] [varchar](50) NULL,

[Portfolio] [varchar](50) NULL,

[CreationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_Collateral\_MISDW] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_Collateral\_MISDW](

[ID] [int] IDENTITY(1,1) NOT NULL,

[PID] [varchar](100) NULL,

[Portfolio] [varchar](100) NULL,

[DESCR] [varchar](100) NULL,

[CollateralClass] [varchar](100) NULL,

[ASOFDATE] [date] NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_Collateral\_MISDW] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_Collateral\_MISDW](

[PID] [varchar](100) NULL,

[Portfolio] [varchar](100) NULL,

[DESCR] [varchar](100) NULL,

[CollateralClass] [varchar](100) NULL,

[INPUTTER] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [date] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_COLLATERAL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_COLLATERAL](

[CollateralType] [varchar](10) NOT NULL,

[Description] [varchar](255) NULL,

[CollateralClassHQLA] [varchar](255) NULL,

[CollateralClassFR2052A] [varchar](255) NULL,

[Comments] [varchar](255) NULL,

[Inputter] [varchar](255) NULL,

[Authoriser] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[Reportingentity] [varchar](10) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_COLLATERAL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_COLLATERAL](

[CollateralType] [varchar](10) NOT NULL,

[Description] [varchar](255) NULL,

[CollateralClassHQLA] [varchar](255) NULL,

[CollateralClassFR2052A] [varchar](255) NULL,

[Comments] [varchar](255) NULL,

[Inputter] [varchar](255) NULL,

[Authoriser] [varchar](255) NULL,

[Reportingentity] [varchar](10) NOT NULL,

[Asofdate] [date] NULL,

[Orig\_AsOfDate] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_CID\_Exception] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_CID\_Exception](

[Ref\_CID\_Exception\_ID] [int] IDENTITY(1,1) NOT NULL,

[Sector] [varchar](20) NULL,

[Industry] [varchar](255) NULL,

[CustomerID] [varchar](255) NULL,

[CID] [varchar](255) NULL,

[Counterparty] [varchar](255) NULL,

[Comments] [varchar](100) NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[AsOfdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_CID\_Exception] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_CID\_Exception](

[Sector] [varchar](20) NULL,

[Industry] [varchar](255) NULL,

[CustomerID] [varchar](255) NULL,

[CID] [varchar](255) NULL,

[Counterparty] [varchar](255) NULL,

[INPUTTER] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[asofdate] [date] NULL,

[orig\_asofdate] [datetime] NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_REF\_CDARS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_REF\_CDARS](

[ReportingEntity] [varchar](9) NULL,

[Company\_Code] [varchar](9) NULL,

[AccountId] [varchar](10) NULL,

[AccountTitle] [varchar](100) NULL,

[CustomerTaxId] [varchar](50) NULL,

[OriginalPrincipalAmount] [varchar](50) NULL,

[Currency] [varchar](3) NULL,

[Sector] [varchar](4) NULL,

[CounterParty] [varchar](50) NULL,

[InternalTransactionIdr] [varchar](1) NULL,

[CustomerId] [varchar](7) NULL,

[BookBalance] [varchar](50) NULL,

[AnticipatedPayoutAtMaturity] [varchar](50) NULL,

[EffectiveDate] [datetime] NULL,

[MaturityDate] [datetime] NULL,

[TermType] [varchar](50) NULL,

[AnnaulInterestRate] [varchar](50) NULL,

[RelationshipAccountId] [varchar](10) NULL,

[RelationshipCustomerId] [varchar](50) NULL,

[CompanyID] [varchar](9) NULL,

[CompanyName] [varchar](100) NULL,

[SalesContactName] [varchar](50) NULL,

[InterestPaymentFrequency] [varchar](50) NULL,

[InterestDisbursementType] [varchar](50) NULL,

[MaturityDisbursementType] [varchar](50) NULL,

[PlacementType] [varchar](50) NULL,

[SourceOfFunds] [varchar](100) NULL,

[OneWayIdr] [varchar](1) NULL,

[AccruedInterestToDate] [varchar](50) NULL,

[AccruedInterestMTD] [varchar](50) NULL,

[InterestPaidYTD] [varchar](50) NULL,

[InterestPaidLastYear] [varchar](50) NULL,

[AccountOpeningFee] [varchar](50) NULL,

[OneWayRevenue] [varchar](50) NULL,

[Asofdate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[US\_FR2052A\_REF\_CDARS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[US\_FR2052A\_REF\_CDARS](

[ReportingEntity] [varchar](9) NULL,

[Company\_Code] [varchar](9) NULL,

[AccountId] [varchar](10) NULL,

[AccountTitle] [varchar](100) NULL,

[CustomerTaxId] [varchar](50) NULL,

[OriginalPrincipalAmount] [varchar](50) NULL,

[Currency] [varchar](3) NULL,

[Sector] [varchar](4) NULL,

[CounterParty] [varchar](50) NULL,

[InternalTransactionIdr] [varchar](1) NULL,

[CustomerId] [varchar](7) NULL,

[BookBalance] [varchar](50) NULL,

[AnticipatedPayoutAtMaturity] [varchar](50) NULL,

[EffectiveDate] [datetime] NULL,

[MaturityDate] [datetime] NULL,

[TermType] [varchar](50) NULL,

[AnnaulInterestRate] [varchar](50) NULL,

[RelationshipAccountId] [varchar](10) NULL,

[RelationshipCustomerId] [varchar](50) NULL,

[CompanyID] [varchar](9) NULL,

[CompanyName] [varchar](100) NULL,

[SalesContactName] [varchar](50) NULL,

[InterestPaymentFrequency] [varchar](50) NULL,

[InterestDisbursementType] [varchar](50) NULL,

[MaturityDisbursementType] [varchar](50) NULL,

[PlacementType] [varchar](50) NULL,

[SourceOfFunds] [varchar](100) NULL,

[OneWayIdr] [varchar](1) NULL,

[AccruedInterestToDate] [varchar](50) NULL,

[AccruedInterestMTD] [varchar](50) NULL,

[InterestPaidYTD] [varchar](50) NULL,

[InterestPaidLastYear] [varchar](50) NULL,

[AccountOpeningFee] [varchar](50) NULL,

[OneWayRevenue] [varchar](50) NULL,

[Asofdate] [date] NULL,

[Orig\_AsOfDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_SI] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_SI](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](3) NOT NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[SID] [varchar](255) NULL,

[Sub\_Product] [varchar](255) NULL,

[SID2] [varchar](255) NULL,

[Sub\_Product2] [varchar](255) NULL,

[MarketValue\_USD] [decimal](30, 15) NULL,

[CollateralClass] [varchar](255) NULL,

[Internal] [varchar](3) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MarketValueOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMarketValue] [varchar](255) NOT NULL,

[CollateralCode] [varchar](255) NOT NULL,

[CollateralDescription] [varchar](255) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_OW] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_OW](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](255) NOT NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[CID\_CounterpartyID] [int] NULL,

[Counterparty] [varchar](255) NULL,

[MaturityAmountUSD] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValueUSD] [decimal](30, 15) NULL,

[CollateralCurrency] [varchar](3) NULL,

[ForwardStartAmountUSD] [decimal](30, 15) NULL,

[ForwardStartDate] [datetime] NULL,

[Internal] [varchar](3) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyTypeCode] [varchar](255) NULL,

[CounterpartyTypeName] [varchar](255) NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MaturityDate] [datetime] NOT NULL,

[MaturityAmountOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMaturityAmount] [varchar](3) NOT NULL,

[ForwardStartAmountOriginalCurrency] [decimal](30, 15) NULL,

[OriginalCurrencyOfForwardStartAmount] [varchar](3) NULL,

[CollateralValueOriginalCurrency] [decimal](30, 15) NULL,

[OriginalCurrencyOfCollateralValue] [varchar](3) NULL,

[CollateralCode] [varchar](255) NULL,

[CollateralDescription] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_OS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_OS](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](3) NOT NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[SID] [varchar](255) NULL,

[Sub\_Product] [varchar](255) NULL,

[MaturityAmountUSD] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[ForwardStartAmountUSD] [varchar](255) NULL,

[ForwardStartDate] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValueUSD] [decimal](30, 15) NULL,

[CollateralCurrency] [varchar](3) NULL,

[TreasuryControl] [varchar](255) NULL,

[Internal] [varchar](255) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[Settlement] [varchar](255) NULL,

[Rehypothecated] [varchar](255) NULL,

[CID\_CounterpartyID] [varchar](255) NULL,

[Counterparty] [varchar](255) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyTypeCode] [varchar](255) NOT NULL,

[CounterpartyTypeName] [varchar](255) NOT NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MaturityDate] [datetime] NOT NULL,

[MaturityAmountOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMaturityAmount] [varchar](255) NOT NULL,

[ForwardStartAmountOriginalCurrency] [varchar](255) NULL,

[OriginalCurrencyOfForwardStartAmount] [varchar](255) NULL,

[OriginalCurrencyOfCollateralValue] [varchar](255) NOT NULL,

[CollateralValueOriginalCurrency] [decimal](30, 15) NOT NULL,

[CollateralCode] [varchar](255) NOT NULL,

[CollateralDescription] [varchar](255) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_OO] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_OO](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](3) NOT NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[CID\_CounterpartyID] [varchar](255) NULL,

[Counterparty] [varchar](255) NULL,

[MaturityAmountUSD] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[ForwardStartAmountUSD] [varchar](255) NULL,

[ForwardStartDate] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValueUSD] [decimal](30, 15) NULL,

[CollateralCurrency] [varchar](3) NULL,

[Internal] [varchar](3) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyTypeCode] [varchar](255) NULL,

[CounterpartyTypeName] [varchar](255) NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MaturityDate] [datetime] NOT NULL,

[MaturityAmountOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMaturityAmount] [varchar](255) NOT NULL,

[ForwardStartAmountinLocalCurrency] [varchar](255) NULL,

[LocalCurrencyforForwardStartAmount] [varchar](255) NULL,

[CollateralValueOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfCollateralValue] [varchar](255) NOT NULL,

[CollateralCode] [varchar](255) NOT NULL,

[CollateralDescription] [varchar](255) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_IU] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_IU](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](3) NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[CID] [varchar](255) NULL,

[Counterparty] [varchar](255) NULL,

[MaturityAmountUSD] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[ForwardStartAmountUSD] [varchar](255) NULL,

[ForwardStartDate] [varchar](255) NULL,

[Internal] [varchar](255) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyTypeCode] [varchar](255) NOT NULL,

[CounterpartyTypeName] [varchar](255) NOT NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MaturityDate] [datetime] NOT NULL,

[MaturityAmountOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMaturityAmount] [varchar](255) NOT NULL,

[ForwardStartAmountinLocalCurrency] [varchar](255) NULL,

[LocalCurrencyforForwardStartAmount] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[US\_FR2052A\_NB\_IO] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[US\_FR2052A\_NB\_IO](

[ID] [int] NOT NULL,

[Reportingentity] [varchar](255) NOT NULL,

[Currency] [varchar](3) NULL,

[Converted] [varchar](3) NOT NULL,

[PID\_ProductID] [int] NOT NULL,

[Product] [varchar](255) NOT NULL,

[MaturityAmountUSD] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[ForwardStartAmountUSD] [varchar](255) NULL,

[ForwardStartDate] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValue] [varchar](255) NULL,

[CollateralCurrency] [varchar](255) NULL,

[TreasuryControl] [varchar](255) NULL,

[Internal] [varchar](3) NULL,

[InternalCounterparty] [varchar](255) NULL,

[PrimeBrokerage] [varchar](3) NULL,

[PortfolioName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[Asofdate] [datetime] NOT NULL,

[EntityCode] [varchar](255) NOT NULL,

[EntityName] [varchar](255) NOT NULL,

[CounterpartyCode] [varchar](255) NOT NULL,

[CounterpartyName] [varchar](255) NOT NULL,

[MaturityDate] [datetime] NOT NULL,

[MaturityAmountOriginalCurrency] [decimal](30, 15) NOT NULL,

[OriginalCurrencyOfMaturityAmount] [varchar](255) NOT NULL,

[ForwardStartAmountOriginalCurrency] [varchar](255) NULL,

[OriginalCurrencyOfForwardStartAmount] [varchar](255) NULL,

[CollateralValueOriginalCurrency] [varchar](255) NULL,

[OriginalCurrencyofCollateralValue] [varchar](255) NULL,

[CollateralCode] [varchar](255) NULL,

[CollateralDescription] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[TMP\_MG\_Mortgage\_Schedule] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[TMP\_MG\_Mortgage\_Schedule](

[COMPANY] [varchar](3) NOT NULL,

[CONTRACT\_NO] [varchar](17) NOT NULL,

[CUSTOMER] [varchar](10) NULL,

[CATEGORY] [varchar](5) NULL,

[DUE\_DATE] [varchar](8) NOT NULL,

[CURRENCY] [varchar](3) NULL,

[INTEREST\_AMOUNT] [decimal](19, 2) NULL,

[PRINCIPAL\_AMOUNT] [decimal](19, 2) NULL,

[OS\_BALANCE] [decimal](19, 2) NULL,

[SHORT\_NAME] [varchar](35) NULL,

[PERIOD] [decimal](15, 0) NULL,

[REMARK] [varchar](1) NULL,

[INTEREST\_RATE] [decimal](9, 6) NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_wholesaleOutflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_wholesaleOutflow](

[Tgt\_wholesaleOutflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [varchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [varchar](255) NULL,

[maturityStart] [varchar](255) NULL,

[maturityEnd] [varchar](255) NULL,

[maturityBucket] [varchar](255) NULL,

[reportScope] [varchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[counterPartySector] [varchar](255) NULL,

[internal] [varchar](255) NULL,

[internalCounterParty] [varchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[primeBrokerage] [varchar](255) NULL,

[product] [varchar](255) NULL,

[reportType] [varchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[CollateralCurrency] [varchar](255) NULL,

[Converted] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[AsofDate] [date] NULL,

[creationDate] [datetime] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_unsecuredInflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_unsecuredInflow](

[TGT\_unsecuredInflowID] [int] IDENTITY(1,1) NOT NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [varchar](255) NULL,

[maturityStart] [varchar](255) NULL,

[maturityEnd] [varchar](255) NULL,

[maturityBucket] [varchar](255) NULL,

[reportScope] [varchar](255) NULL,

[counterPartySector] [varchar](255) NULL,

[internal] [varchar](255) NULL,

[internalCounterParty] [varchar](255) NULL,

[lendablevalue] [decimal](30, 15) NULL,

[maturityValue] [decimal](30, 15) NULL,

[primeBrokerage] [varchar](255) NULL,

[product] [varchar](255) NULL,

[reportType] [varchar](255) NULL,

[CID] [decimal](20, 15) NULL,

[Converted] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[SID] [varchar](255) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](100) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[AsofDate] [date] NULL,

[creationDate] [datetime] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_securedOutflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_securedOutflow](

[Tgt\_securedOutflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [nvarchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [nvarchar](255) NULL,

[maturityStart] [nvarchar](255) NULL,

[maturityEnd] [nvarchar](255) NULL,

[maturityBucket] [nvarchar](255) NULL,

[reportScope] [nvarchar](255) NULL,

[treasuryControl] [nvarchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[counterPartySector] [nvarchar](255) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterParty] [nvarchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[rehypothecated] [nvarchar](255) NULL,

[primeBrokerage] [nvarchar](255) NULL,

[settlementMechanism] [nvarchar](255) NULL,

[product] [nvarchar](255) NULL,

[subProduct] [nvarchar](255) NULL,

[reportType] [nvarchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[CollateralCurrency] [nvarchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[Currency] [nvarchar](255) NULL,

[PID] [varchar](255) NULL,

[SID] [decimal](18, 0) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[creationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL,

[publicCounterParty] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_securedInflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_securedInflow](

[Tgt\_securedInflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [varchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [varchar](255) NULL,

[maturityStart] [varchar](255) NULL,

[maturityEnd] [varchar](255) NULL,

[maturityBucket] [varchar](255) NULL,

[reportScope] [varchar](255) NULL,

[treasuryControl] [varchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[counterPartySector] [varchar](255) NULL,

[internal] [varchar](255) NULL,

[internalCounterParty] [varchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[primeBrokerage] [varchar](255) NULL,

[effectiveMaturityBucket] [varchar](255) NULL,

[product] [varchar](255) NULL,

[settlementMechanism] [varchar](255) NULL,

[subProduct] [varchar](255) NULL,

[unencumbered] [varchar](255) NULL,

[reportType] [varchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[CollateralCurrency] [varchar](255) NULL,

[Converted] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[SID] [varchar](255) NULL,

[effectiveMaturityStart] [varchar](255) NULL,

[effectiveMaturityEnd] [varchar](255) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NOT NULL,

[SrcID] [varchar](100) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[creationDate] [date] NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_otherOutflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_otherOutflow](

[Tgt\_otherOutflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [varchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [varchar](255) NULL,

[maturityStart] [varchar](255) NULL,

[maturityEnd] [varchar](255) NULL,

[maturityBucket] [varchar](255) NULL,

[reportScope] [varchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[counterPartySector] [varchar](255) NULL,

[internal] [varchar](255) NULL,

[internalCounterParty] [varchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[primeBrokerage] [varchar](255) NULL,

[product] [varchar](255) NULL,

[reportType] [varchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[Converted] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[CollateralCurrency] [varchar](255) NULL,

[maturityDate] [date] NULL,

[creationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_otherInflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_otherInflow](

[Tgt\_otherInflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [nvarchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [nvarchar](255) NULL,

[maturityStart] [nvarchar](255) NULL,

[maturityEnd] [nvarchar](255) NULL,

[maturityBucket] [nvarchar](255) NULL,

[reportScope] [nvarchar](255) NULL,

[treasuryControl] [nvarchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterParty] [nvarchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[primeBrokerage] [nvarchar](255) NULL,

[product] [nvarchar](255) NULL,

[reportType] [nvarchar](255) NULL,

[CollateralCurrency] [nvarchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[Currency] [nvarchar](255) NULL,

[PID] [varchar](255) NULL,

[Original\_PID] [varchar](50) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[original\_maturity] [decimal](30, 15) NULL,

[maturityDate] [date] NULL,

[creationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[QTY] [decimal](30, 15) NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_informationalSupplemental] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_informationalSupplemental](

[Tgt\_informationalSupplementalID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [nvarchar](255) NULL,

[marketValue] [decimal](30, 15) NULL,

[reportScope] [nvarchar](255) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterParty] [nvarchar](255) NULL,

[primeBrokerage] [nvarchar](255) NULL,

[subProduct] [nvarchar](255) NULL,

[subProduct2] [nvarchar](255) NULL,

[Product] [nvarchar](255) NULL,

[reportType] [nvarchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[Currency] [nvarchar](255) NULL,

[PID] [varchar](255) NULL,

[SID] [decimal](18, 0) NULL,

[SID2] [decimal](18, 0) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](255) NULL,

[DESC1] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[CompanyName] [varchar](255) NULL,

[creationDate] [date] NULL,

[AsofDate] [date] NULL,

[Event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_ForeignExchange] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_ForeignExchange](

[Tgt\_ForeignExchange\_ID] [int] IDENTITY(1,1) NOT NULL,

[SrcID] [varchar](255) NULL,

[ReportScope] [varchar](255) NULL,

[Currency\_1] [varchar](255) NULL,

[Currency\_1\_Converted] [varchar](255) NULL,

[Currency\_2] [varchar](255) NULL,

[Currency\_2\_Converted] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[Product] [varchar](255) NULL,

[MaturityAmountCurrency1] [decimal](30, 15) NULL,

[MaturityAmountCurrency2] [decimal](30, 15) NULL,

[MaturityBucket] [varchar](255) NULL,

[ForeignExchangeOptionDirection] [varchar](255) NULL,

[ForwardStartAmountCurrenc1] [decimal](30, 15) NULL,

[ForwardStartAmountCurrenc2] [decimal](30, 15) NULL,

[ForwardStartBucket] [varchar](255) NULL,

[Settlement] [varchar](255) NULL,

[PrimeBrokeage] [varchar](255) NULL,

[Internal] [varchar](255) NULL,

[InternalCounterParty] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[maturityDate] [date] NULL,

[portfolio] [varchar](255) NULL,

[AsofDate] [date] NULL,

[CreationDate] [datetime] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_depositsOutflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_depositsOutflow](

[Tgt\_depositsOutflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [nvarchar](255) NULL,

[comment] [nvarchar](255) NULL,

[maturityStart] [nvarchar](255) NULL,

[maturityEnd] [nvarchar](255) NULL,

[maturityBucket] [nvarchar](255) NULL,

[reportScope] [nvarchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[counterPartySector] [nvarchar](255) NULL,

[insured] [nvarchar](255) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterParty] [nvarchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[product] [nvarchar](255) NULL,

[trigger] [nvarchar](255) NULL,

[reportType] [nvarchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[CollateralCurrency] [nvarchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[Currency] [nvarchar](255) NULL,

[PID] [varchar](255) NULL,

[Rehypothecated] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[Portfolio] [varchar](255) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[creationDate] [date] NULL,

[Orig\_PID] [varchar](50) NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_Comments] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_Comments](

[Comment\_id] [int] IDENTITY(1,1) NOT NULL,

[ReportScope] [varchar](50) NULL,

[Type] [varchar](50) NULL,

[Product] [varchar](255) NULL,

[Subproduct] [varchar](255) NULL,

[Text] [varchar](max) NULL,

[PID] [varchar](50) NULL,

[SID] [varchar](50) NULL,

[Portfolio] [varchar](50) NULL,

[CreationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[Inputter] [varchar](50) NULL,

[Authoriser] [varchar](50) NULL,

[ReportingEntity] [varchar](1000) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Tgt\_assetInflow] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Tgt\_assetInflow](

[Tgt\_assetInflowID] [int] IDENTITY(1,1) NOT NULL,

[collateralClass] [varchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [varchar](255) NULL,

[maturityStart] [varchar](255) NULL,

[maturityEnd] [varchar](255) NULL,

[lendableValue] [decimal](30, 15) NULL,

[QTY] [decimal](30, 15) NULL,

[marketValue] [decimal](30, 15) NULL,

[maturityBucket] [varchar](255) NULL,

[product] [varchar](255) NULL,

[reportScope] [varchar](255) NULL,

[SubProduct] [varchar](255) NULL,

[treasuryControl] [varchar](255) NULL,

[reportType] [varchar](255) NULL,

[converted] [varchar](255) NULL,

[currency] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[SID] [decimal](18, 0) NULL,

[forwardStart] [varchar](255) NULL,

[forwardEnd] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[Sector] [varchar](255) NULL,

[maturityDate] [date] NULL,

[creationDate] [datetime] NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL,

[customerNumber] [varchar](255) NULL,

[orig\_marketValue] [decimal](30, 15) NULL,

[orig\_maturityValue] [decimal](30, 15) NULL,

[orig\_currency] [varchar](255) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[T\_FMDDATA] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[T\_FMDDATA](

[N\_ID] [int] NOT NULL,

[SEQ\_NO] [varchar](20) NULL,

[REPORT\_PERIOD\_END] [datetime] NULL,

[CONTACT\_ID] [varchar](255) NULL,

[CONSOL\_KEY] [varchar](255) NULL,

[APPLICATION\_ID] [varchar](2) NULL,

[CATEGORY] [int] NULL,

[ASSET\_TYPE] [varchar](12) NULL,

[CURRENCY] [varchar](3) NULL,

[DESC1] [varchar](255) NULL,

[DEAL\_LOCAL\_BAL] [decimal](19, 2) NULL,

[DEAL\_MAT\_DATE] [datetime] NULL,

[LD\_ROLLOVER\_DATE] [datetime] NULL,

[MG\_ROLLOVER] [datetime] NULL,

[LOCAL\_BALANCE] [decimal](19, 2) NULL,

[FOREIGN\_BALANCE] [decimal](19, 2) NULL,

[SCHED\_LOCAL\_BAL] [decimal](19, 2) NULL,

[SCHED\_FCY\_BAL] [decimal](19, 2) NULL,

[CUSTOMER\_NO] [varchar](10) NULL,

[CUSTOMER\_NAME] [varchar](255) NULL,

[CUST\_RESIDENCE] [varchar](10) NULL,

[CUST\_COUNTRY] [varchar](10) NULL,

[DEAL\_BALANCE] [decimal](19, 2) NULL,

[DEAL\_RATE] [decimal](11, 5) NULL,

[DEAL\_VALUE\_DATE] [datetime] NULL,

[DEAL\_MAT\_PERIOD] [int] NULL,

[DEAL\_REMAIN\_DAYS] [int] NULL,

[COMPOUND\_ID] [varchar](255) NULL,

[COMPANY\_CODE] [varchar](11) NULL,

[SECTOR] [varchar](4) NULL,

[OFFICER] [varchar](10) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NULL,

[REPORT\_LINE] [varchar](10) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[RPT\_FR2052a] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [app].[RPT\_FR2052a](

[RPT\_ID] [int] IDENTITY(1,1) NOT NULL,

[AGG\_ID] [int] NOT NULL,

[collateralClass] [nvarchar](255) NULL,

[comment] [nvarchar](255) NULL,

[forwardStartValue] [decimal](20, 2) NULL,

[forwardStartBucket] [nvarchar](255) NULL,

[maturityStart] [nvarchar](255) NULL,

[maturityEnd] [nvarchar](255) NULL,

[lendableValue] [decimal](20, 2) NULL,

[marketValue] [decimal](20, 2) NULL,

[maturityBucket] [nvarchar](255) NULL,

[assetProduct] [nvarchar](255) NULL,

[reportScope] [nvarchar](255) NOT NULL,

[assetSource] [nvarchar](255) NULL,

[treasuryControl] [nvarchar](255) NULL,

[reportID] [nvarchar](255) NULL,

[collateralValue] [decimal](20, 2) NULL,

[counterPartySector] [nvarchar](255) NULL,

[insured] [nvarchar](255) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterparty] [nvarchar](255) NULL,

[publicCounterparty] [nvarchar](255) NULL,

[maturityValue] [decimal](20, 2) NULL,

[depositsProduct] [nvarchar](255) NULL,

[rehypothecated] [nvarchar](255) NULL,

[trigger] [nvarchar](255) NULL,

[primeBrokerage] [nvarchar](255) NULL,

[otherInflowProduct] [nvarchar](255) NULL,

[otherOutflowProduct] [nvarchar](255) NULL,

[effectiveMaturityBucket] [nvarchar](255) NULL,

[securedInflowProduct] [nvarchar](255) NULL,

[settlementMechanism] [nvarchar](255) NULL,

[pledgeLevel] [nvarchar](255) NULL,

[unencumbered] [nvarchar](255) NULL,

[wholesaleOutflowProduct] [nvarchar](255) NULL,

[securedOutflowProduct] [nvarchar](255) NULL,

[securedOutflowCounterPartyOrCoverage] [nvarchar](255) NULL,

[unsecuredProduct] [nvarchar](255) NULL,

[segregationCategory] [nvarchar](255) NULL,

[supplementalInformationProduct] [nvarchar](255) NULL,

[assetInflowReport] [nvarchar](255) NULL,

[depositsOutflowReport] [nvarchar](255) NULL,

[otherInflowReport] [nvarchar](255) NULL,

[otherOutflowReport] [nvarchar](255) NULL,

[securedInflowReport] [nvarchar](255) NULL,

[securedOutflowReport] [nvarchar](255) NULL,

[unsecuredInflowReport] [nvarchar](255) NULL,

[wholesaleOutflowReport] [nvarchar](255) NULL,

[informationalSupplementalReport] [nvarchar](255) NULL,

[Converted] [nvarchar](5) NULL,

[Currency] [nvarchar](3) NULL,

[FmaturityStart] [nvarchar](255) NULL,

[FmaturityEnd] [nvarchar](255) NULL,

[EmaturityStart] [nvarchar](255) NULL,

[EmaturityEnd] [nvarchar](255) NULL,

[asofdate] [date] NULL,

[creationDate] [datetime] NULL

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [SRC].[FBNK\_LIMIT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[FBNK\_LIMIT](

[ID] [varchar](32) NOT NULL,

[CREDIT\_LINE\_NO] [varchar](32) NULL,

[LIMIT\_CURRENCY] [varchar](3) NULL,

[COUNTRY\_OF\_RISK] [varchar](2) NULL,

[COUNTRY\_PERCENT] [varchar](3) NULL,

[APPROVAL\_DATE] [varchar](8) NULL,

[OFFERED\_UNTIL] [varchar](8) NULL,

[REVIEW\_FREQUENCY] [varchar](17) NULL,

[EXPIRY\_DATE] [varchar](8) NULL,

[ADM\_EXTENSION\_DATE] [varchar](8) NULL,

[PROPOSAL\_DATE] [varchar](8) NULL,

[ONLINE\_LIMIT\_DATE] [varchar](8) NULL,

[PRODUCT\_ALLOWED] [varchar](85) NULL,

[LIMIT\_GRADE] [varchar](4) NULL,

[GRADE\_PERCENT] [varchar](3) NULL,

[NOTES] [varchar](85) NULL,

[INTERNAL\_AMOUNT] [decimal](19, 2) NULL,

[ADVISED\_AMOUNT] [decimal](19, 2) NULL,

[FIXED\_VARIABLE] [varchar](8) NULL,

[COLLATERAL\_CODE] [varchar](85) NULL,

[COLL\_REQD\_AMT] [varchar](85) NULL,

[COLL\_REQD\_PRCNT] [varchar](85) NULL,

[UP\_TO\_PERIOD] [varchar](11) NULL,

[PERIOD\_AMT] [varchar](85) NULL,

[PERIOD\_PRCNT] [varchar](85) NULL,

[MAXIMUM\_SECURED] [varchar](85) NULL,

[MAXIMUM\_UNSECURED] [decimal](19, 2) NULL,

[MAXIMUM\_TOTAL] [decimal](19, 2) NULL,

[OTHER\_SECURED] [decimal](19, 2) NULL,

[COLLAT\_RIGHT] [varchar](65) NULL,

[SECURED\_AMT] [varchar](85) NULL,

[TIME\_CODE] [varchar](85) NULL,

[LIMIT\_BAND] [varchar](85) NULL,

[ONLINE\_LIMIT] [varchar](85) NULL,

[SUB\_ALLOC\_TAKEN] [varchar](85) NULL,

[SUB\_ALLOC\_GIVEN] [varchar](85) NULL,

[COMMT\_AMT\_AVAIL] [varchar](85) NULL,

[TOTAL\_OS] [varchar](85) NULL,

[AVAIL\_AMT] [varchar](85) NULL,

[S\_AL\_TKN\_CCY] [varchar](3) NULL,

[S\_AL\_TKN\_AMT] [varchar](85) NULL,

[S\_AL\_GVN\_CCY] [varchar](3) NULL,

[S\_AL\_GVN\_AMT] [varchar](85) NULL,

[COMMITM\_CCY] [varchar](3) NULL,

[COMMITM\_AMT] [varchar](85) NULL,

[OS\_CCY] [varchar](65) NULL,

[OS\_AMT] [varchar](255) NULL,

[DATE\_OF\_EXCESS] [varchar](8) NULL,

[AMT\_LAST\_EXCESS] [varchar](85) NULL,

[DATE\_LAST\_MOVED] [varchar](85) NULL,

[AMT\_LAST\_MOVED] [varchar](85) NULL,

[DATE\_1ST\_EXCESS] [varchar](8) NULL,

[DATE\_LAST\_REVAL] [varchar](8) NULL,

[ORIGINAL\_LIMIT] [decimal](19, 2) NULL,

[CLEAN\_RISK] [decimal](19, 2) NULL,

[DATE\_CLRISK\_EXC] [varchar](8) NULL,

[AVAILABLE\_MARKER] [varchar](1) NULL,

[RESTRICTED\_CUST] [varchar](10) NULL,

[ALLOWED\_CCY] [varchar](21) NULL,

[ALLOWED\_AMT] [varchar](85) NULL,

[RESTRICTED\_CCY] [varchar](21) NULL,

[ALLOWED\_COMP] [varchar](21) NULL,

[RESTRICTED\_COMP] [varchar](21) NULL,

[COMMODITY] [varchar](4) NULL,

[COMMODITY\_PCENT] [varchar](3) NULL,

[COLLAT\_CODE] [varchar](4) NULL,

[COLLAT\_CUSTOMER] [varchar](10) NULL,

[COLLAT\_CCY] [varchar](3) NULL,

[COLLAT\_AMOUNT] [varchar](85) NULL,

[COLLAT\_PERCENT] [decimal](9, 6) NULL,

[COLLAT\_LOCATION] [varchar](2) NULL,

[COLLAT\_GRADE] [varchar](4) NULL,

[COLLAT\_NOTES] [varchar](35) NULL,

[REDUCING\_SCHED] [varchar](6) NULL,

[REPAY\_FREQ] [varchar](17) NULL,

[REPAY\_NUMBER] [varchar](3) NULL,

[REPAY\_CCY] [varchar](3) NULL,

[REPAY\_AMOUNT] [varchar](85) NULL,

[REPAY\_END\_DATE] [varchar](8) NULL,

[REPAY\_NOTES] [varchar](35) NULL,

[ACC\_COMPANY] [varchar](255) NULL,

[ACCOUNT] [varchar](85) NULL,

[REDUCING\_LIMIT] [varchar](1) NULL,

[NO\_OF\_SUB\_ALLOCS] [varchar](4) NULL,

[FX\_OR\_TIME\_BAND] [varchar](2) NULL,

[LIABILITY\_NUMBER] [varchar](10) NULL,

[RECORD\_PARENT] [varchar](35) NULL,

[CREDIT\_LINE] [varchar](24) NULL,

[CHECK\_LIMIT] [varchar](1) NULL,

[CONSOL\_KEY] [varchar](65) NULL,

[LIMIT\_PRODUCT] [varchar](7) NULL,

[ALLOW\_NETTING] [varchar](3) NULL,

[DISPO\_OFFICER] [varchar](10) NULL,

[ONLINE\_UPDATE] [varchar](1) NULL,

[INCREASING\_TYPE] [varchar](10) NULL,

[INCREASE\_FREQ] [varchar](17) NULL,

[INCREASE\_CCY] [varchar](3) NULL,

[INCREASE\_AMOUNT] [varchar](85) NULL,

[INCR\_END\_DATE] [varchar](8) NULL,

[UNUTIL\_ACCT] [varchar](16) NULL,

[CURR\_UNUTIL\_AMT] [decimal](19, 2) NULL,

[ALLOW\_UNUTIL\_CR] [varchar](3) NULL,

[PREV\_UNUTIL\_ACCT] [varchar](16) NULL,

[PREV\_UNUTIL\_AMT] [decimal](19, 2) NULL,

[RESERVED\_2] [varchar](55) NULL,

[RESERVED\_1] [varchar](55) NULL,

[OVERRIDE] [varchar](125) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](35) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[LAST\_DRAW\_DATE] [varchar](8) NULL,

[RESERVED\_3] [varchar](55) NULL,

[ALLOWED\_CUST] [varchar](10) NULL,

[CREATED\_FROM] [varchar](32) NULL,

[LIMITS\_MODIFIED] [varchar](32) NULL,

[RESERVED\_7] [varchar](55) NULL,

[RESERVED\_6] [varchar](55) NULL,

[RESERVED\_5] [varchar](55) NULL,

[RESERVED\_4] [varchar](55) NULL,

[SYNDICATE\_LDR] [varchar](85) NULL,

[ARRANGER] [varchar](85) NULL,

[BUYER\_SELLER] [varchar](85) NULL,

[CUSIP\_NO] [varchar](9) NULL,

[COMMITTED\_YN] [varchar](1) NULL,

[PRICING] [varchar](35) NULL,

[FAC\_RISK\_RATING] [varchar](4) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[UTIL\_ACCOUNT] [varchar](16) NULL,

[CURR\_UTIL\_AMT] [decimal](16, 2) NULL,

[PREV\_UTIL\_ACCT] [decimal](16, 2) NULL,

[PREV\_UTIL\_AMT] [decimal](16, 2) NULL,

[ALLOC\_WORK\_ID] [varchar](5) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[FBNK\_CUSTOMER] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[FBNK\_CUSTOMER](

[ID] [varchar](10) NOT NULL,

[CUSTOMER\_CODE] [varchar](10) NULL,

[MNEMONIC] [varchar](10) NULL,

[SHORT\_NAME] [varchar](35) NULL,

[NAME\_1] [varchar](35) NULL,

[NAME\_2] [varchar](35) NULL,

[STREET] [varchar](35) NULL,

[TOWN\_COUNTRY] [varchar](35) NULL,

[RELATION\_CODE] [varchar](255) NULL,

[REL\_CUSTOMER] [varchar](255) NULL,

[REVERS\_REL\_CODE] [varchar](2) NULL,

[SECTOR] [varchar](4) NULL,

[ACCOUNT\_OFFICER] [varchar](10) NULL,

[OTHER\_OFFICER] [varchar](10) NULL,

[INDUSTRY] [varchar](4) NULL,

[TARGET] [varchar](4) NULL,

[NATIONALITY] [varchar](2) NULL,

[CUSTOMER\_STATUS] [varchar](4) NULL,

[RESIDENCE] [varchar](2) NULL,

[CONTACT\_DATE] [varchar](8) NULL,

[INTRODUCER] [varchar](35) NULL,

[TEXT1] [varchar](255) NULL,

[LEGAL\_ID] [varchar](20) NULL,

[REVIEW\_FREQUENCY] [varchar](17) NULL,

[BIRTH\_INCORP\_DATE] [varchar](8) NULL,

[GLOBAL\_CUSTOMER] [varchar](10) NULL,

[CUSTOMER\_LIABILITY] [varchar](10) NULL,

[LANGUAGE] [varchar](2) NULL,

[OVERRIDES] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[RESERVED02] [varchar](35) NULL,

[RESERVED01] [varchar](35) NULL,

[POSTING\_RESTRICT] [varchar](2) NULL,

[DISPO\_OFFICER] [varchar](10) NULL,

[POST\_CODE] [varchar](35) NULL,

[COUNTRY] [varchar](35) NULL,

[BOOK] [varchar](3) NULL,

[CONFID\_TXT] [varchar](3) NULL,

[DISPO\_EXEMPT] [varchar](3) NULL,

[ISSUE\_CHEQUES] [varchar](3) NULL,

[CLS\_CPARTY] [varchar](3) NULL,

[FX\_COMM\_GROUP\_ID] [varchar](3) NULL,

[RESIDENCE\_REGION] [varchar](9) NULL,

[CUST\_COND\_PRIOR] [varchar](35) NULL,

[US\_TOWN\_CITY] [varchar](20) NULL,

[US\_STATE] [varchar](2) NULL,

[US\_ZIP\_CODE] [varchar](10) NULL,

[US\_PH\_TYPE] [varchar](255) NULL,

[US\_PH\_NUM] [varchar](255) NULL,

[US\_TAX\_ID\_TYP] [varchar](15) NULL,

[US\_TAX\_ID] [varchar](11) NULL,

[US\_OTH\_ID\_TYP] [varchar](255) NULL,

[US\_OTH\_ID] [varchar](255) NULL,

[FOREIGN\_TAX\_ID] [varchar](200) NULL,

[W8\_W9\_TAX] [varchar](5) NULL,

[PERCENTAGE] [varchar](10) NULL,

[PRIMARY\_ID] [varchar](35) NULL,

[CARD\_NUMBER] [varchar](35) NULL,

[EXPIRED\_ON] [varchar](8) NULL,

[ID\_ISSUED\_BY] [varchar](35) NULL,

[ISSUE\_DATE] [varchar](8) NULL,

[SECONDARY\_ID] [varchar](35) NULL,

[OCCUPATION] [varchar](35) NULL,

[ANNUAL\_INCOME] [varchar](20) NULL,

[GENDER] [varchar](5) NULL,

[MOTHERS\_MAIDEN] [varchar](35) NULL,

[BUSINESS\_NAME] [varchar](35) NULL,

[PRODUCT\_TYPES] [varchar](35) NULL,

[BNSS\_OPERATION] [varchar](15) NULL,

[OTHER\_BUSINESS] [varchar](35) NULL,

[ANTICIPATE\_VOL] [varchar](35) NULL,

[GEOGRAPHIC\_AREA] [varchar](35) NULL,

[INTL\_ACCT\_TRANS] [varchar](1) NULL,

[TRAN\_DESCRIPT] [varchar](35) NULL,

[MAJOR\_SUPPLIER] [varchar](80) NULL,

[SUPPLIER\_CUST\_R] [varchar](35) NULL,

[SUPPLIER\_ADDR] [varchar](35) NULL,

[BANK\_REFERENCE] [varchar](35) NULL,

[BANK\_ACCT\_NO] [varchar](35) NULL,

[EMAIL] [varchar](35) NULL,

[WEBSITE] [varchar](35) NULL,

[CONTACT\_NAME] [varchar](35) NULL,

[LC\_HANDLE\_INFO] [varchar](35) NULL,

[LC\_COURIER\_NAME] [varchar](35) NULL,

[LC\_COURIER\_ACCT] [varchar](35) NULL,

[PRIOR\_CUST\_NO] [varchar](15) NULL,

[PRIMARY\_COMPANY] [varchar](11) NULL,

[CURRENT\_STATUS] [varchar](24) NULL,

[COUNTRY\_RISK] [varchar](9) NULL,

[BK\_RISK\_TYPE] [varchar](11) NULL,

[BK\_RISK\_CLASS] [varchar](2) NULL,

[CBD\_EXT\_RSK\_TYP] [varchar](10) NULL,

[CBD\_EXT\_RSK\_RAT] [varchar](10) NULL,

[LC\_RISK\_TYPE] [varchar](20) NULL,

[LC\_RISK\_RATE] [varchar](2) NULL,

[LD\_LOAN\_RESERVE] [varchar](50) NULL,

[LD\_OWNER\_CODE] [varchar](2) NULL,

[LD\_OWNER\_TYPE] [varchar](2) NULL,

[DATE\_OPENED] [varchar](8) NULL,

[INTRODUCED\_BY] [varchar](35) NULL,

[LAST\_ACTIV\_DATE] [varchar](8) NULL,

[US\_REL\_INST] [varchar](10) NULL,

[US\_REL\_OWN] [varchar](10) NULL,

[MAJOR\_CUSTOMER] [varchar](80) NULL,

[SYNDICATED\_BORR] [varchar](8) NULL,

[US\_TAX\_STATUS] [varchar](8) NULL,

[US\_TAX\_DOC] [varchar](12) NULL,

[US\_EMAIL] [varchar](255) NULL,

[US\_PRIMARY\_COMP] [varchar](11) NULL,

[US\_POL\_POSN] [varchar](35) NULL,

[US\_PEP] [varchar](1) NULL,

[US\_CUST\_STATUS] [varchar](24) NULL,

[US\_COUNTRY\_RISK] [varchar](9) NULL,

[US\_OCCUPATION] [varchar](35) NULL,

[US\_PRIMARY\_ID] [varchar](35) NULL,

[US\_BSA\_EXEMPT] [varchar](1) NULL,

[US\_BSA\_RISK] [varchar](2) NULL,

[CONTINGENT\_RISK] [varchar](3) NULL,

[LOSS\_SPECIFIC] [varchar](50) NULL,

[NOT\_USED] [varchar](7) NULL,

[HO\_OWNERSHIP] [varchar](2) NULL,

[HO\_INDUSTRY] [varchar](2) NULL,

[LC\_STATE\_INCORP] [varchar](2) NULL,

[LC\_REMARK] [varchar](65) NULL,

[US\_INACT\_SINCE] [varchar](8) NULL,

[US\_ASP\_SINCE] [varchar](8) NULL,

[US\_LAST\_CONTACT] [varchar](8) NULL,

[SECOND\_TAX\_ID] [varchar](35) NULL,

[CA\_NAICS\_CODE] [varchar](20) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[COMPANY\_BOOK] [varchar](11) NULL,

[ASSET\_CLASS] [varchar](3) NULL,

[CUSTOMER\_RATING] [varchar](35) NULL,

[ADDRESS] [varchar](125) NULL,

[CR\_PROFILE\_TYPE] [varchar](125) NULL,

[CR\_PROFILE] [varchar](125) NULL,

[NO\_UPDATE\_CRM] [varchar](3) NULL,

[TITLE] [varchar](6) NULL,

[GIVEN\_NAMES] [varchar](50) NULL,

[FAMILY\_NAME] [varchar](50) NULL,

[DATE\_OF\_BIRTH] [varchar](11) NULL,

[MARITAL\_STATUS] [varchar](11) NULL,

[NO\_OF\_DEPENDENTS] [varchar](2) NULL,

[PHONE\_1] [varchar](55) NULL,

[SMS\_1] [varchar](55) NULL,

[EMAIL\_1] [varchar](255) NULL,

[EMPLOYMENT\_STATUS] [varchar](125) NULL,

[EMPLOYERS\_NAME] [varchar](125) NULL,

[EMPLOYERS\_ADD] [varchar](125) NULL,

[EMPLOYERS\_BUSS] [varchar](125) NULL,

[EMPLOYMENT\_START] [varchar](36) NULL,

[CUSTOMER\_CURRENCY] [varchar](12) NULL,

[SALARY] [varchar](60) NULL,

[ANNUAL\_BONUS] [varchar](60) NULL,

[SALARY\_DATE\_FREQ] [varchar](55) NULL,

[NET\_MONTHLY\_IN] [decimal](19, 2) NULL,

[NET\_MONTHLY\_OUT] [decimal](19, 2) NULL,

[RESIDENCE\_STATUS] [varchar](125) NULL,

[RESIDENCE\_TYPE] [varchar](125) NULL,

[RESIDENCE\_SINCE] [varchar](36) NULL,

[RESIDENCE\_VALUE] [varchar](60) NULL,

[MORTGAGE\_AMT] [varchar](60) NULL,

[OTHER\_FIN\_REL] [varchar](12) NULL,

[OTHER\_FIN\_INST] [varchar](125) NULL,

[COMM\_TYPE] [varchar](125) NULL,

[PREF\_CHANNEL] [varchar](60) NULL,

[INTERESTS] [varchar](125) NULL,

[RESERVED\_04] [varchar](55) NULL,

[RESERVED\_03] [varchar](55) NULL,

[RESERVED\_02] [varchar](55) NULL,

[RESERVED\_01] [varchar](55) NULL,

[FAX\_1] [varchar](55) NULL,

[PREVIOUS\_NAME] [varchar](125) NULL,

[CHANGE\_DATE] [varchar](36) NULL,

[CHANGE\_REASON] [varchar](125) NULL,

[CUSTOMER\_SINCE] [varchar](11) NULL,

[CUSTOMER\_TYPE] [varchar](35) NULL,

[RESERVED\_06] [varchar](55) NULL,

[RESERVED\_08] [varchar](55) NULL,

[RESERVED\_07] [varchar](55) NULL,

[DATE\_LAST\_VERIFIED] [varchar](15) NULL,

[REL\_DELIV\_OPT] [varchar](30) NULL,

[REL\_RESERV6] [varchar](225) NULL,

[REL\_RESERV5] [varchar](225) NULL,

[REL\_RESERV4] [varchar](225) NULL,

[REL\_RESERV3] [varchar](225) NULL,

[REL\_RESERV2] [varchar](225) NULL,

[REL\_RESERV1] [varchar](225) NULL,

[RESERVED\_51] [varchar](55) NULL,

[RESERVED\_33] [varchar](55) NULL,

[RESERVED\_32] [varchar](55) NULL,

[RESERVED\_31] [varchar](55) NULL,

[RESERVED\_30] [varchar](55) NULL,

[RESERVED\_29] [varchar](55) NULL,

[RESERVED\_28] [varchar](55) NULL,

[RESERVED\_27] [varchar](55) NULL,

[RESERVED\_26] [varchar](55) NULL,

[RESERVED\_25] [varchar](55) NULL,

[RESERVED\_24] [varchar](55) NULL,

[RESERVED\_23] [varchar](55) NULL,

[RESERVED\_22] [varchar](55) NULL,

[RESERVED\_21] [varchar](55) NULL,

[RESERVED\_20] [varchar](55) NULL,

[RESERVED\_19] [varchar](55) NULL,

[RESERVED\_18] [varchar](55) NULL,

[RESERVED\_17] [varchar](55) NULL,

[RESERVED\_16] [varchar](55) NULL,

[RESERVED\_15] [varchar](55) NULL,

[RESERVED\_14] [varchar](55) NULL,

[RESERVED\_13] [varchar](55) NULL,

[RESERVED\_12] [varchar](55) NULL,

[RESERVED\_11] [varchar](55) NULL,

[RESERVED\_10] [varchar](55) NULL,

[RESERVED\_09] [varchar](55) NULL,

[RESERVED\_05] [varchar](55) NULL,

[ROLE] [varchar](65) NULL,

[ROLE\_MORE\_INFO] [varchar](125) NULL,

[ROLE\_NOTES] [varchar](125) NULL,

[LEGAL\_DOC\_NAME] [varchar](60) NULL,

[LEGAL\_ISS\_AUTH] [varchar](60) NULL,

[LEGAL\_ISS\_DATE] [varchar](36) NULL,

[LEGAL\_EXP\_DATE] [varchar](36) NULL,

[OFF\_PHONE] [varchar](255) NULL,

[LEGAL\_HOLDER\_NAME] [varchar](310) NULL,

[JOB\_TITLE] [varchar](35) NULL,

[SPOKEN\_LANGUAGE] [varchar](35) NULL,

[PASTIMES] [varchar](125) NULL,

[FURTHER\_DETAILS] [varchar](125) NULL,

[DOMICILE] [varchar](9) NULL,

[OTHER\_NATIONALITY] [varchar](30) NULL,

[CALC\_RISK\_CLASS] [varchar](10) NULL,

[MANUAL\_RISK\_CLASS] [varchar](10) NULL,

[OVERRIDE\_REASON] [varchar](35) NULL,

[TAX\_ID] [varchar](125) NULL,

[VIS\_TYPE] [varchar](35) NULL,

[VIS\_COMMENT] [varchar](125) NULL,

[VIS\_INTERNAL\_REVIEW] [varchar](125) NULL,

[FORMER\_VIS\_TYPE] [varchar](35) NULL,

[FORMER\_VIS\_COMMENT] [varchar](125) NULL,

[RISK\_ASSET\_TYPE] [varchar](30) NULL,

[RISK\_LEVEL] [varchar](10) NULL,

[RISK\_TOLERANCE] [varchar](10) NULL,

[RISK\_FROM\_DATE] [varchar](36) NULL,

[LAST\_KYC\_REVIEW\_DATE] [varchar](11) NULL,

[AUTO\_NEXT\_KYC\_REVIEW\_DATE] [varchar](11) NULL,

[MANUAL\_NEXT\_KYC\_REVIEW\_DATE] [varchar](11) NULL,

[LAST\_SUIT\_REVIEW\_DATE] [varchar](11) NULL,

[AUTO\_NEXT\_SUIT\_REVIEW\_DATE] [varchar](11) NULL,

[MANUAL\_NEXT\_SUIT\_REVIEW\_DATE] [varchar](11) NULL,

[KYC\_RELATIONSHIP] [varchar](35) NULL,

[US\_PRIM\_ID\_TYPE] [varchar](35) NULL,

[US\_PRIM\_ID\_EXP] [varchar](11) NULL,

[US\_OTH\_ID\_EXP] [varchar](11) NULL,

[US\_GENDER] [varchar](1) NULL,

[US\_MARITAL\_STAT] [varchar](3) NULL,

[US\_PH\_EXTN] [varchar](10) NULL,

[CA\_WATCH\_LIST] [varchar](35) NULL,

[NOT\_IN\_USE] [varchar](85) NULL,

[CA\_KMV\_ID] [varchar](50) NULL,

[CA\_TICKER] [varchar](125) NULL,

[BR\_CUST\_NAME1] [varchar](35) NULL,

[BR\_CUST\_NAME2] [varchar](35) NULL,

[BR\_STREE] [varchar](35) NULL,

[BR\_TOWN\_COUNTRY] [varchar](35) NULL,

[BR\_FATHER\_NAME] [varchar](35) NULL,

[BR\_MOTHER\_NAME] [varchar](35) NULL,

[BR\_SPOUSE\_NAME] [varchar](35) NULL,

[BR\_CONTACT\_NAME] [varchar](35) NULL,

[BR\_BUSS\_NAME] [varchar](35) NULL,

[BR\_BIRTH\_PLACE] [varchar](35) NULL,

[MANDATE\_APPL] [varchar](125) NULL,

[SECURE\_MESSAGE] [varchar](3) NULL,

[MANDATE\_RECORD] [varchar](125) NULL,

[HOLDINGS\_PIVOT] [varchar](65) NULL,

[INTERNET\_BANKING\_SERVICE] [varchar](4) NULL,

[ALLOW\_BULK\_PROCESS] [varchar](6) NULL,

[KYC\_COMPLETE] [varchar](3) NULL,

[MOBILE\_BANKING\_SERVICE] [varchar](4) NULL,

[MANDATE\_REG] [varchar](125) NULL,

[AML\_RESULT] [varchar](14) NULL,

[AML\_CHECK] [varchar](4) NULL,

[LAST\_AML\_RESULT\_DATE] [varchar](11) NULL,

[REPORT\_TEMPLATE] [varchar](35) NULL,

[EXTERN\_CUS\_ID] [varchar](170) NULL,

[MERGED\_STATUS] [varchar](35) NULL,

[LEGAL\_ID\_DOC\_NAME] [varchar](125) NULL,

[ADDR\_LOCATION] [varchar](50) NULL,

[MERGED\_TO] [varchar](35) NULL,

[ALT\_CUS\_ID] [varchar](125) NULL,

[EXTERN\_SYS\_ID] [varchar](170) NULL,

[NCB\_BUS\_CLASS] [varchar](6) NULL,

[NCB\_INTERCOMPAN] [varchar](6) NULL,

[NCB\_PARENT\_CODE] [varchar](2) NULL,

[NCB\_S\_CHI\_OWN] [varchar](2) NULL,

[NCB\_OWNERSHIP] [varchar](5) NULL,

[HKMA\_INDUSTRY\_C] [varchar](4) NULL,

[NCB\_CUST\_TYPE] [varchar](5) NULL,

[NCB\_YBD\_CLS] [varchar](20) NULL,

[NCB\_GROUP\_NUM] [varchar](5) NULL,

[BR\_MAIN\_BUSS] [varchar](200) NULL,

[BR\_BUSS\_H\_ADDR] [varchar](200) NULL,

[NCB\_PUBLIC\_TRAD] [varchar](200) NULL,

[NCB\_MAJ\_INCOM\_L] [varchar](200) NULL,

[NCB\_MAJ\_ASSET\_L] [varchar](200) NULL,

[OC\_LOCATION] [varchar](300) NULL,

[HKMA\_TREASURY] [varchar](300) NULL,

[S80\_INDICATOR] [varchar](300) NULL,

[S\_HKMA\_CRG7] [varchar](300) NULL,

[S\_83] [varchar](300) NULL,

[INCOP\_CN] [varchar](300) NULL,

[INCOP\_ANY\_CN] [varchar](300) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[FBNK\_CURRENCY] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[FBNK\_CURRENCY](

[ID] [varchar](3) NOT NULL,

[CURRENCY\_CODE] [varchar](3) NULL,

[RANK] [varchar](3) NULL,

[NUMERIC\_CCY\_CODE] [varchar](3) NULL,

[CCY\_NAME] [varchar](35) NULL,

[NO\_OF\_DECIMALS] [varchar](1) NULL,

[QUOTATION\_CODE] [varchar](1) NULL,

[QUOTATION\_PIPS] [varchar](1) NULL,

[DAYS\_DELIVERY] [varchar](2) NULL,

[DAYS\_FORWARD] [varchar](2) NULL,

[INTEREST\_DAY\_BASIS] [varchar](9) NULL,

[RATE\_ALLOWANCE] [decimal](9, 6) NULL,

[FIXING\_DATE] [decimal](19, 2) NULL,

[CURRENCY\_MARKET] [varchar](10) NULL,

[QUOTATION\_SUSP] [varchar](10) NULL,

[MID\_REVAL\_RATE] [varchar](65) NULL,

[DEFAULT\_SPREAD] [varchar](65) NULL,

[BUY\_RATE] [varchar](65) NULL,

[SELL\_RATE] [varchar](65) NULL,

[UPTO\_SMALL\_AMT] [varchar](18) NULL,

[TRSY\_SMALL\_SPRD] [varchar](11) NULL,

[CUST\_SMALL\_SPRD] [varchar](11) NULL,

[NEGOTIABLE\_AMT] [varchar](65) NULL,

[TRSY\_MED\_SPREAD] [varchar](11) NULL,

[CUST\_MED\_SPREAD] [varchar](11) NULL,

[TRSY\_LIMIT\_AMT] [varchar](18) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[MIN\_ROUND\_AMOUNT] [decimal](19, 2) NULL,

[CASH\_ONLY\_ROUNDING] [varchar](3) NULL,

[MIN\_ROUND\_TYPE] [varchar](1) NULL,

[CASH\_ROUND\_TYPE] [varchar](1) NULL,

[PRECIOUS\_METAL] [varchar](3) NULL,

[RESERVED2] [varchar](55) NULL,

[RESERVED1] [varchar](55) NULL,

[EQUIVALENT\_CCYS] [varchar](3) NULL,

[COUNTRY\_CODE] [varchar](2) NULL,

[FIXED\_RATE] [decimal](9, 6) NULL,

[FIXED\_CCY] [varchar](3) NULL,

[FIXED\_START\_DATE] [varchar](8) NULL,

[BASE\_CCY\_RANK] [varchar](3) NULL,

[AVAILABLE\_DATE] [varchar](8) NULL,

[BLOCKED\_DATE] [varchar](8) NULL,

[RESERVED14] [varchar](55) NULL,

[RESERVED13] [varchar](55) NULL,

[RESERVED12] [varchar](55) NULL,

[RESERVED11] [varchar](55) NULL,

[RESERVED10] [varchar](55) NULL,

[RESERVED9] [varchar](55) NULL,

[RESERVED8] [varchar](55) NULL,

[RESERVED7] [varchar](55) NULL,

[RESERVED6] [varchar](55) NULL,

[RESERVED5] [varchar](55) NULL,

[RESERVED4] [varchar](55) NULL,

[RESERVED3] [varchar](55) NULL,

[REVAL\_RATE] [varchar](65) NULL,

[CLS\_CCY] [varchar](3) NULL,

[CUT\_OFF\_TIME] [varchar](6) NULL,

[REMIT\_CURR\_110] [varchar](15) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[FBNK\_COLLATERAL\_RIGHT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[FBNK\_COLLATERAL\_RIGHT](

[ID] [varchar](13) NOT NULL,

[COLLATERAL\_RIGHT] [varchar](13) NULL,

[COLLATERAL\_CODE] [varchar](3) NULL,

[COMPANY\_ID] [varchar](3) NULL,

[LIMIT\_REFERENCE] [varchar](32) NULL,

[PERCENTAGE\_COVER] [varchar](3) NULL,

[VALIDITY\_DATE] [varchar](8) NULL,

[REVIEW\_DATE\_FQU] [varchar](17) NULL,

[EXPIRY\_DATE] [varchar](8) NULL,

[NOTARY] [varchar](10) NULL,

[NOTES] [varchar](35) NULL,

[CUSTOMER] [varchar](10) NULL,

[STATUS] [varchar](3) NULL,

[LIMIT\_ID] [varchar](32) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[LIMIT\_REF\_CUST] [varchar](10) NULL,

[PERCENT\_ALLOC] [varchar](3) NULL,

[RESERVED\_4] [varchar](32) NULL,

[RESERVED\_3] [varchar](32) NULL,

[RESERVED\_2] [varchar](32) NULL,

[RESERVED\_1] [varchar](32) NULL,

[OS\_PERCENT\_COVER] [varchar](3) NULL,

[TOT\_COLLAT\_AMT] [varchar](19) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[ALLOC\_WORK\_ID] [varchar](5) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[FBNK\_COLLATERAL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[FBNK\_COLLATERAL](

[ID] [varchar](16) NOT NULL,

[COLLATERAL\_ID] [varchar](16) NULL,

[COLLATERAL\_TYPE] [varchar](3) NULL,

[DESCRIPTION] [varchar](35) NULL,

[COMPANY\_ID] [varchar](3) NULL,

[APPLICATION\_ID] [varchar](34) NULL,

[CURRENCY] [varchar](3) NULL,

[COUNTRY] [varchar](2) NULL,

[NOMINAL\_VALUE] [decimal](19, 2) NULL,

[EXECUTION\_VALUE] [decimal](19, 2) NULL,

[THIRD\_PARTY\_VALUE] [decimal](19, 2) NULL,

[GEN\_LEDGER\_VALUE] [decimal](19, 2) NULL,

[CENTRAL\_BANK\_VALUE] [decimal](19, 2) NULL,

[VALUE\_DATE] [varchar](8) NULL,

[REVIEW\_DATE\_FQU] [varchar](17) NULL,

[EXPIRY\_DATE] [varchar](8) NULL,

[ADDRESS] [varchar](50) NULL,

[NOTES] [varchar](35) NULL,

[APPLICATION] [varchar](2) NULL,

[STATUS] [varchar](3) NULL,

[REVALUED\_DATE] [varchar](8) NULL,

[REVALUED\_AMOUNT] [decimal](19, 2) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[MAXIMUM\_VALUE] [decimal](19, 2) NULL,

[RESERVED11] [varchar](55) NULL,

[RESERVED10] [varchar](55) NULL,

[RESERVED9] [varchar](55) NULL,

[RESERVED8] [varchar](55) NULL,

[RESERVED7] [varchar](55) NULL,

[RESERVED6] [varchar](55) NULL,

[RESERVED5] [varchar](55) NULL,

[RESERVED4] [varchar](55) NULL,

[RESERVED3] [varchar](55) NULL,

[RESERVED2] [varchar](55) NULL,

[RESERVED1] [varchar](55) NULL,

[RESERVED14] [varchar](55) NULL,

[RESERVED13] [varchar](55) NULL,

[COLLATERAL\_CODE] [varchar](3) NULL,

[COLLAT\_ISSUES] [varchar](28) NULL,

[COMPANY] [varchar](3) NOT NULL,

[ASOFDATE] [datetime] NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Ctrl\_Tgt\_Agg\_BOC] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Ctrl\_Tgt\_Agg\_BOC](

[CtrlTgtAggID] [int] IDENTITY(1,1) NOT NULL,

[ValidationCheck] [varchar](255) NULL,

[Tgt\_reportScope] [varchar](255) NULL,

[Tgt\_PID] [varchar](50) NULL,

[Tgt\_product] [varchar](255) NULL,

[Tgt\_maturityValue] [decimal](30, 15) NULL,

[Tgt\_marketValue] [decimal](30, 15) NULL,

[Tgt\_NumRec] [int] NULL,

[Agg\_reportScope] [varchar](255) NULL,

[Agg\_PID] [int] NULL,

[Agg\_product] [varchar](255) NULL,

[Agg\_maturityValue] [decimal](30, 15) NULL,

[Agg\_marketValue] [decimal](30, 15) NULL,

[Agg\_NumRec] [int] NULL,

[AsofDate] [datetime] NULL,

[ExecutionDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Ctrl\_Tgt\_Agg] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Ctrl\_Tgt\_Agg](

[CtrlTgtAggID] [int] IDENTITY(1,1) NOT NULL,

[ValidationCheck] [varchar](255) NULL,

[Tgt\_reportScope] [varchar](255) NULL,

[Tgt\_PID] [varchar](50) NULL,

[Tgt\_product] [varchar](255) NULL,

[Tgt\_maturityValue] [decimal](30, 15) NULL,

[Tgt\_marketValue] [decimal](30, 15) NULL,

[Tgt\_NumRec] [int] NULL,

[Agg\_reportScope] [varchar](255) NULL,

[Agg\_PID] [int] NULL,

[Agg\_product] [varchar](255) NULL,

[Agg\_maturityValue] [decimal](30, 15) NULL,

[Agg\_marketValue] [decimal](30, 15) NULL,

[Agg\_NumRec] [int] NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Ctrl\_Src\_Tgt] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Ctrl\_Src\_Tgt](

[CtrlSrcTgtID] [int] IDENTITY(1,1) NOT NULL,

[ValidationCheck] [varchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[Product] [varchar](255) NULL,

[SourceTableName] [nchar](200) NULL,

[SourceFieldGroupBy] [varchar](255) NULL,

[SourceFieldGroupByValue] [varchar](255) NULL,

[SourceAggregatedField] [varchar](255) NULL,

[SourceAggregatedValue] [decimal](30, 15) NULL,

[#ofRecordsFromSrc] [int] NULL,

[TargetTableName] [varchar](255) NULL,

[TargetFieldGroupBy] [varchar](255) NULL,

[TargetFieldGroupByValue] [varchar](255) NULL,

[TargetAggregatedField] [varchar](255) NULL,

[TargetAggregatedValue] [decimal](30, 15) NULL,

[#ofRecordsFromTgt] [int] NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[COMPANY] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[COMPANY](

[ID] [varchar](255) NULL,

[COMPANY\_CODE] [varchar](255) NULL,

[COMPANY\_NAME] [varchar](255) NULL,

[NAME\_ADDRESS] [varchar](255) NULL,

[MNEMONIC] [varchar](255) NULL,

[LANGUAGE\_CODE] [varchar](255) NULL,

[STAT\_REP\_AREA] [varchar](255) NULL,

[STAT\_REP\_NAME] [varchar](255) NULL,

[STAT\_REP\_DELIV] [varchar](255) NULL,

[MGMT\_REP\_AREA] [varchar](255) NULL,

[MGMT\_REP\_NAME] [varchar](255) NULL,

[MGMT\_REP\_DELIV] [varchar](255) NULL,

[CONSOLIDATION\_MARK] [varchar](255) NULL,

[DEFAULT\_NO\_OF\_AUTH] [varchar](255) NULL,

[PGM\_AUTOM\_ID] [varchar](255) NULL,

[LOCAL\_COUNTRY] [varchar](255) NULL,

[LOCAL\_REGION] [varchar](255) NULL,

[LOCAL\_CURRENCY] [varchar](255) NULL,

[RATE\_SYSTEM] [varchar](255) NULL,

[CUSTOMER\_COMPANY] [varchar](255) NULL,

[CUSTOMER\_MNEMONIC] [varchar](255) NULL,

[DEFAULT\_CUST\_COM] [varchar](255) NULL,

[DEFAULT\_CUST\_MNE] [varchar](255) NULL,

[SPCL\_CUST\_FILE] [varchar](255) NULL,

[SPCL\_CUST\_COM] [varchar](255) NULL,

[SPCL\_CUST\_MNE] [varchar](255) NULL,

[DEFAULT\_FINAN\_COM] [varchar](255) NULL,

[DEFAULT\_FINAN\_MNE] [varchar](255) NULL,

[SPCL\_FIN\_FILE] [varchar](255) NULL,

[SPCL\_FIN\_COM] [varchar](255) NULL,

[SPCL\_FIN\_MNE] [varchar](255) NULL,

[CURRENCY\_COMPANY] [varchar](255) NULL,

[CURRENCY\_MNEMONIC] [varchar](255) NULL,

[FINAN\_FINAN\_COM] [varchar](255) NULL,

[FINAN\_FINAN\_MNE] [varchar](255) NULL,

[NOSTRO\_COMPANY] [varchar](255) NULL,

[NOSTRO\_MNEMONIC] [varchar](255) NULL,

[NOSTRO\_SUB\_DIV] [varchar](255) NULL,

[INTER\_COM\_CATEGORY] [varchar](255) NULL,

[APPLICATIONS] [varchar](255) NULL,

[LOCAL\_PROCESS\_NAME] [varchar](255) NULL,

[DEFAULT\_RATE\_ALLOW] [decimal](30, 15) NULL,

[FCY\_CONV\_TOLER\_MIN] [decimal](30, 15) NULL,

[FCY\_CONV\_TOLER\_MAX] [decimal](30, 15) NULL,

[CHARGE\_CALCULATION] [varchar](255) NULL,

[FINANCIAL\_YEAR\_END] [varchar](255) NULL,

[INACTIVITY\_MONTHS] [varchar](255) NULL,

[SUB\_DIVISION\_CODE] [varchar](255) NULL,

[SWIFT\_ADD\_COM\_REF] [varchar](255) NULL,

[LAST\_YEAR\_END] [varchar](255) NULL,

[DC\_TOLERANCE\_RATE] [decimal](30, 15) NULL,

[DC\_MAX\_ROUND\_AMT] [decimal](30, 15) NULL,

[ACCT\_CHECKDIG\_TYPE] [varchar](255) NULL,

[ACC\_BKNO\_PREFIX] [varchar](255) NULL,

[ACCOUNT\_MASK] [varchar](255) NULL,

[RECORD\_STATUS] [varchar](255) NULL,

[CURR\_NO] [varchar](255) NULL,

[INPUTTER] [varchar](255) NULL,

[DATE\_TIME] [varchar](255) NULL,

[AUTHORISER] [varchar](255) NULL,

[CO\_CODE] [varchar](255) NULL,

[DEPT\_CODE] [varchar](255) NULL,

[AUDITOR\_CODE] [varchar](255) NULL,

[AUDIT\_DATE\_TIME] [varchar](255) NULL,

[LOCAL\_REF] [varchar](255) NULL,

[BOOK] [varchar](255) NULL,

[RESERVED\_2] [varchar](255) NULL,

[RESERVED\_1] [varchar](255) NULL,

[COMPANY\_GROUP\_ID] [varchar](255) NULL,

[CURR\_DATE] [varchar](255) NULL,

[RELATIVE\_TIME] [varchar](255) NULL,

[SEPARATE\_EOD] [varchar](255) NULL,

[CONS\_KEY\_CO] [varchar](255) NULL,

[COB\_DATE] [varchar](255) NULL,

[COMP\_STATUS] [varchar](255) NULL,

[COMPANY\_GROUP] [varchar](255) NULL,

[OFFICIAL\_HOLIDAY] [varchar](255) NULL,

[BRANCH\_HOLIDAY] [varchar](255) NULL,

[BATCH\_HOLIDAY] [varchar](255) NULL,

[BRANCH\_CLOSED] [varchar](255) NULL,

[ASOFDATE] [datetime] NULL,

[COMPANY] [varchar](255) NULL,

[FINANCIAL\_COM] [varchar](255) NULL,

[FINANCIAL\_MNE] [varchar](255) NULL,

[DEF\_RATING\_AGENCY] [varchar](255) NULL,

[CUST\_RETENTION] [varchar](255) NULL,

[PROSPECT\_ALLOWED] [varchar](255) NULL,

[COMPANY\_SCHEMA] [varchar](255) NULL,

[RESERVED\_7] [varchar](255) NULL,

[RESERVED\_6] [varchar](255) NULL,

[RESERVED\_5] [varchar](255) NULL,

[RESERVED\_4] [varchar](255) NULL,

[RESERVED\_3] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[AGG\_FR2052a] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[AGG\_FR2052a](

[AGG\_FR2052aID] [int] IDENTITY(1,1) NOT NULL,

[PID] [varchar](255) NULL,

[Product] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[ReportScope] [varchar](255) NULL,

[Internal] [varchar](255) NULL,

[ForwardStartBucket] [varchar](255) NULL,

[MaturityBucket] [varchar](255) NULL,

[EffectiveMaturityBucket] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[LendableValue] [decimal](30, 15) NULL,

[MarketValue] [decimal](30, 15) NULL,

[MaturityValue] [decimal](30, 15) NULL,

[ForwardStartValue] [decimal](30, 15) NULL,

[ForwardStart] [varchar](255) NULL,

[ForwardEnd] [varchar](255) NULL,

[MaturityStart] [varchar](255) NULL,

[MaturityEnd] [varchar](255) NULL,

[EffectiveMaturityStart] [varchar](255) NULL,

[EffectiveMaturityEnd] [varchar](255) NULL,

[TreasuryControl] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValue] [decimal](30, 15) NULL,

[CollateralCurrency] [varchar](255) NULL,

[CounterPartySector] [nvarchar](255) NULL,

[Insured] [varchar](255) NULL,

[InternalCounterParty] [varchar](255) NULL,

[Rehypothecated] [varchar](255) NULL,

[trigger] [nvarchar](255) NULL,

[PrimeBrokerage] [varchar](255) NULL,

[SettlementMechanism] [varchar](255) NULL,

[Unencumbered] [varchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[SID] [decimal](18, 0) NULL,

[SID2] [decimal](18, 0) NULL,

[SubProduct] [varchar](255) NULL,

[Subproduct2] [varchar](255) NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[AGG\_BOC] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[AGG\_BOC](

[Agg\_BOC\_ID] [int] IDENTITY(1,1) NOT NULL,

[Portfolio] [varchar](255) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[PID] [varchar](255) NULL,

[Product] [varchar](255) NULL,

[ReportType] [varchar](255) NULL,

[ReportScope] [varchar](255) NULL,

[Internal] [varchar](255) NULL,

[ForwardStartBucket] [varchar](255) NULL,

[MaturityBucket] [varchar](255) NULL,

[EffectiveMaturityBucket] [varchar](255) NULL,

[Currency] [varchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[LendableValue] [decimal](30, 15) NULL,

[MarketValue] [decimal](30, 15) NULL,

[MaturityValue] [decimal](30, 15) NULL,

[ForwardStartValue] [decimal](30, 15) NULL,

[ForwardStart] [varchar](255) NULL,

[ForwardEnd] [varchar](255) NULL,

[MaturityStart] [varchar](255) NULL,

[MaturityEnd] [varchar](255) NULL,

[EffectiveMaturityStart] [varchar](255) NULL,

[EffectiveMaturityEnd] [varchar](255) NULL,

[TreasuryControl] [varchar](255) NULL,

[CollateralClass] [varchar](255) NULL,

[CollateralValue] [decimal](30, 15) NULL,

[CollateralCurrency] [varchar](255) NULL,

[CounterPartySector] [nvarchar](255) NULL,

[Insured] [varchar](255) NULL,

[InternalCounterParty] [varchar](255) NULL,

[Rehypothecated] [varchar](255) NULL,

[trigger] [nvarchar](255) NULL,

[PrimeBrokerage] [varchar](255) NULL,

[SettlementMechanism] [varchar](255) NULL,

[Unencumbered] [varchar](255) NULL,

[CID] [decimal](18, 0) NULL,

[SID] [decimal](18, 0) NULL,

[SID2] [decimal](18, 0) NULL,

[SubProduct] [varchar](255) NULL,

[Subproduct2] [varchar](255) NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[Multi\_MM] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[Multi\_MM](

[ID] [varchar](14) NOT NULL,

[CONTRACT\_NO] [varchar](14) NULL,

[CUSTOMER\_ID] [varchar](10) NULL,

[CURRENCY] [varchar](3) NULL,

[PRINCIPAL] [decimal](19, 2) NULL,

[DEAL\_DATE] [varchar](8) NULL,

[VALUE\_DATE] [varchar](8) NULL,

[MATURITY\_DATE] [varchar](8) NULL,

[CATEGORY] [varchar](6) NULL,

[INT\_RATE\_TYPE] [varchar](10) NULL,

[INTEREST\_BASIS] [varchar](1) NULL,

[INTEREST\_RATE] [decimal](9, 6) NULL,

[INTEREST\_KEY] [varchar](4) NULL,

[INTEREST\_SPREAD\_1] [decimal](9, 6) NULL,

[FIRST\_DAY\_ACCRUAL] [varchar](3) NULL,

[TOT\_INTEREST\_AMT] [decimal](19, 2) NULL,

[INT\_DUE\_DATE] [varchar](19) NULL,

[LIQ\_DEFER\_INTEREST] [varchar](1) NULL,

[BROKER\_CODE] [varchar](10) NULL,

[BROKERAGE] [decimal](19, 2) NULL,

[BROKER\_CCY] [varchar](3) NULL,

[BROKER\_AMOUNT] [decimal](19, 2) NULL,

[OUR\_REMARKS] [varchar](35) NULL,

[DRAWDOWN\_ACCOUNT] [varchar](16) NULL,

[PRIN\_LIQ\_ACCT] [varchar](16) NULL,

[PRIN\_BEN\_BANK\_1] [varchar](10) NULL,

[PRIN\_BEN\_BANK\_2] [varchar](10) NULL,

[PRIN\_ADDRESS] [varchar](35) NULL,

[BENEF\_PRIN\_ACCT] [varchar](35) NULL,

[INT\_LIQ\_ACCT] [varchar](16) NULL,

[INT\_BEN\_BANK\_1] [varchar](10) NULL,

[INT\_BEN\_BANK\_2] [varchar](10) NULL,

[INT\_ADDRESS] [varchar](35) NULL,

[BEN\_INT\_ACCT] [varchar](35) NULL,

[TAX\_INTEREST\_KEY] [varchar](2) NULL,

[LIMIT\_REFERENCE] [varchar](10) NULL,

[THIRD\_PARTY\_REF] [varchar](16) NULL,

[CUSTOMER\_REF] [varchar](16) NULL,

[CUST\_REMARKS] [varchar](70) NULL,

[FEDERAL\_FUNDS] [varchar](1) NULL,

[DELIVERY\_LINK] [varchar](3) NULL,

[CHRG\_LIQ\_ACCT] [varchar](16) NULL,

[CHARGE\_CODE] [varchar](11) NULL,

[CHARGE\_AMOUNT] [decimal](19, 2) NULL,

[CURRENCY\_MARKET] [varchar](1) NULL,

[POSITION\_TYPE] [varchar](2) NULL,

[NEG\_REF\_NUMBER] [varchar](5) NULL,

[INT\_REF\_NUMBER] [varchar](5) NULL,

[MIS\_INTEREST\_KEY] [varchar](3) NULL,

[MIS\_INTEREST\_RATE] [decimal](9, 6) NULL,

[COUNTRY\_RISK] [varchar](2) NULL,

[COUNTRY\_EXPOSURE] [varchar](2) NULL,

[MIS\_ACCT\_OFFICER] [varchar](10) NULL,

[FID\_COMM\_RATE] [decimal](9, 6) NULL,

[FID\_TOT\_COMM\_AMNT] [decimal](19, 2) NULL,

[FIDUCIARY\_BANK] [varchar](10) NULL,

[FID\_ASSOC\_LD\_REF] [varchar](14) NULL,

[FID\_DRAWDOWN\_ACCT] [varchar](16) NULL,

[FID\_PRIN\_LIQ\_ACCT] [varchar](16) NULL,

[FID\_PRIN\_BEN\_BK\_1] [varchar](10) NULL,

[FID\_PRIN\_BEN\_BK\_2] [varchar](10) NULL,

[FID\_PRIN\_ADD] [varchar](35) NULL,

[FID\_BEN\_PRIN\_AC] [varchar](35) NULL,

[ACTIVITY\_CODE] [varchar](8) NULL,

[ADVICE\_SENT] [varchar](25) NULL,

[PRIN\_INCREASE] [decimal](19, 2) NULL,

[INCR\_EFF\_DATE] [varchar](8) NULL,

[ROLLOVER\_MARKER] [varchar](1) NULL,

[ROLLOVER\_DATE] [varchar](8) NULL,

[NEW\_INTEREST\_RATE] [decimal](9, 6) NULL,

[NEW\_INTRST\_SPREAD] [decimal](9, 6) NULL,

[CAPITALISATION] [varchar](1) NULL,

[CONF\_BY\_CUST] [varchar](35) NULL,

[CONF\_BY\_BROKER] [varchar](35) NULL,

[SWIFT\_COMMON\_REF] [varchar](16) NULL,

[STATUS] [varchar](6) NULL,

[TOT\_INT\_TAX] [decimal](19, 2) NULL,

[DEALER\_DESK] [varchar](2) NULL,

[ORIGINAL\_FUNCTION] [varchar](1) NULL,

[COMM\_RATE\_FLAG] [varchar](1) NULL,

[COMM\_AMT\_FLAG] [varchar](1) NULL,

[PORTFOLIO\_NUMBER] [varchar](3) NULL,

[BK\_TO\_BK\_INFO] [varchar](35) NULL,

[INT\_CCY] [varchar](3) NULL,

[INT\_CCY\_MKT] [varchar](1) NULL,

[CHG\_TAX\_CODE] [varchar](2) NULL,

[CHG\_TAX\_AMT] [decimal](19, 2) NULL,

[CHG\_TAX\_AMT\_LCY] [decimal](19, 2) NULL,

[TOTAL\_CHARGE] [decimal](19, 2) NULL,

[TOTAL\_TAX\_ON\_CHG] [decimal](19, 2) NULL,

[CHG\_VALUE\_DATE] [varchar](8) NULL,

[AUTO\_ROLLOVER] [varchar](1) NULL,

[AUTO\_CAPITALISE] [varchar](1) NULL,

[AUTO\_ROLL\_TERM] [varchar](11) NULL,

[ROLLOVER\_INT\_RATE] [varchar](11) NULL,

[LINKED\_CONTRACT] [varchar](12) NULL,

[LINK\_REFERENCE] [varchar](12) NULL,

[STMT\_NO] [varchar](29) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[LIQUIDATION\_MODE] [varchar](15) NULL,

[OVERDUE\_STATUS] [varchar](3) NULL,

[INT\_PERIOD\_START] [varchar](8) NULL,

[INT\_PERIOD\_END] [varchar](8) NULL,

[NEXT\_INT\_AMOUNT] [decimal](19, 2) NULL,

[NEXT\_INT\_TAX\_AMT] [decimal](19, 2) NULL,

[NEXT\_PRIN\_AMOUNT] [decimal](19, 2) NULL,

[ORIG\_START\_DATE] [varchar](8) NULL,

[ORIG\_MAT\_DATE] [varchar](8) NULL,

[PREV\_MAT\_DATE] [varchar](8) NULL,

[MI\_FUNDING\_DEPT] [varchar](4) NULL,

[TAX\_INTEREST\_TYPE] [varchar](17) NULL,

[CONTRACT\_GRP] [varchar](5) NULL,

[PI\_INT\_KEY] [varchar](2) NULL,

[MERGE\_CONTRACT] [varchar](14) NULL,

[MERGE\_AMOUNT] [decimal](19, 2) NULL,

[FINAL\_MATURITY] [varchar](11) NULL,

[MATURE\_AT\_SOD] [varchar](3) NULL,

[MAX\_LEGAL\_BAL] [decimal](19, 2) NULL,

[FV\_RATE\_KEY] [varchar](4) NULL,

[FV\_MARGIN\_KEY] [varchar](4) NULL,

[GR\_INT\_TAX\_LCY] [decimal](19, 2) NULL,

[GR\_INT\_TAX\_FCY] [decimal](19, 2) NULL,

[MATURED\_ONLINE] [varchar](3) NULL,

[FUND\_ID] [varchar](16) NULL,

[MIS\_COF\_TYPE] [varchar](35) NULL,

[MIS\_COF\_RATE] [decimal](9, 6) NULL,

[MIS\_COF\_DEPT] [varchar](35) NULL,

[NEGATIVE\_RATE] [varchar](3) NULL,

[REVOLVING] [varchar](1) NULL,

[SECURED] [varchar](1) NULL,

[OLD\_CONTRACT] [varchar](27) NULL,

[TELLER\_REF] [varchar](18) NULL,

[PENALTY\_DAYS] [varchar](5) NULL,

[MM\_JOINT\_HOLDER] [varchar](65) NULL,

[MM\_REL\_CODE] [varchar](20) NULL,

[MM\_JOINT\_NOTES] [varchar](35) NULL,

[LOST\_CD\_REASON] [varchar](2) NULL,

[CHG\_CNG\_WAIVE] [varchar](1) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[SEND\_PAYMENT] [varchar](3) NULL,

[RESERVED2] [varchar](1) NULL,

[RESERVED1] [varchar](1) NULL,

[INT\_SCHEDULE] [varchar](19) NULL,

[NEXT\_INT\_SCH\_AMT] [decimal](19, 2) NULL,

[ROUNDING\_RULE] [varchar](20) NULL,

[SPEC\_INT\_METHOD] [varchar](3) NULL,

[INT\_TAX\_LCY] [varchar](77) NULL,

[IAS\_CLASSIFICATION] [varchar](35) NULL,

[ACCRUAL\_PARAM] [varchar](35) NULL,

[MARKET\_MARGIN] [varchar](12) NULL,

[PRODCAT] [varchar](6) NULL,

[INT\_TAX\_FCY] [varchar](77) NULL,

[MARKET\_RATE] [varchar](11) NULL,

[SUB\_TYPE] [varchar](35) NULL,

[COMPOUND\_YLD\_MTHD] [varchar](8) NULL,

[RESERVED8] [varchar](1) NULL,

[RESERVED7] [varchar](1) NULL,

[RESERVED3] [varchar](1) NULL,

[RESERVED9] [varchar](1) NULL,

[RESERVED6] [varchar](1) NULL,

[COMPOUND\_TYPE] [varchar](5) NULL,

[EFF\_INT\_RATE] [varchar](16) NULL,

[RESERVED10] [varchar](1) NULL,

[RESERVED4] [varchar](1) NULL,

[RESERVED5] [varchar](1) NULL,

[NCB\_PRODUCT\_COD] [varchar](10) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[Multi\_MG\_MORTGAGE] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[Multi\_MG\_MORTGAGE](

[ID] [varchar](17) NOT NULL,

[CONTRACT\_NO] [varchar](17) NULL,

[CUSTOMER] [varchar](10) NULL,

[CONTRACT\_TYPE] [varchar](10) NULL,

[CONTRACT\_STATUS] [varchar](3) NULL,

[CURRENCY] [varchar](3) NULL,

[PRINCIPAL\_AMOUNT] [decimal](19, 2) NULL,

[TOTAL\_PRINCIPAL] [decimal](19, 2) NULL,

[ASSOC\_CONTRACT] [varchar](17) NULL,

[CATEGORY] [varchar](5) NULL,

[CONTRACT\_DATE] [varchar](8) NULL,

[VALUE\_DATE] [varchar](8) NULL,

[TERM] [varchar](4) NULL,

[ORIGINAL\_MAT\_DATE] [varchar](8) NULL,

[MATURITY\_DATE] [varchar](8) NULL,

[INTEREST\_BASIS] [varchar](2) NULL,

[UP\_TO\_PRINCIPAL] [decimal](19, 2) NULL,

[FIXED\_VAR\_IND] [varchar](8) NULL,

[INTEREST\_RATE] [decimal](9, 6) NULL,

[INTEREST\_KEY] [varchar](4) NULL,

[INTEREST\_SPREAD] [decimal](9, 6) NULL,

[EFFECTIVE\_INT] [decimal](9, 6) NULL,

[REPAYMENT\_DATE] [varchar](13) NULL,

[REDEMPTION\_DATE] [varchar](13) NULL,

[REDEM\_START\_DATE] [varchar](8) NULL,

[REPAYMENT\_AMOUNT] [decimal](19, 2) NULL,

[THEORETICAL\_AMOUNT] [decimal](19, 2) NULL,

[COMMITTED\_INTEREST] [decimal](19, 2) NULL,

[INT\_REVIEW\_FREQ] [varchar](17) NULL,

[REPAY\_REVIEW\_FREQ] [varchar](17) NULL,

[NEW\_INT\_EFF\_DATE] [varchar](8) NULL,

[NEW\_UP\_TO\_PRIN] [decimal](19, 2) NULL,

[NEW\_FIX\_VAR\_IND] [varchar](8) NULL,

[NEW\_INT\_RATE] [decimal](9, 6) NULL,

[NEW\_INT\_KEY] [varchar](3) NULL,

[NEW\_INT\_SPREAD] [decimal](9, 6) NULL,

[PAYMENT\_TYPE] [varchar](1) NULL,

[NEW\_REPAY\_AMT] [decimal](19, 2) NULL,

[NEW\_REDEM\_DATE] [varchar](8) NULL,

[NEW\_THEORETICAL] [decimal](19, 2) NULL,

[PENALTY\_RATE] [decimal](9, 6) NULL,

[RATE\_FIX\_FQY] [varchar](17) NULL,

[LIQUIDATION\_MODE] [varchar](15) NULL,

[OVERDUE\_STATUS] [varchar](3) NULL,

[RESIDUAL\_PRINCIPAL] [decimal](19, 2) NULL,

[LIMIT\_REF] [varchar](10) NULL,

[MORTGAGE\_ACCOUNT] [varchar](16) NULL,

[DEPOSIT\_ACCOUNT] [varchar](16) NULL,

[DRAWDOWN\_ACCOUNT] [varchar](16) NULL,

[FIRST\_PAY\_IN\_ADV] [varchar](1) NULL,

[ADD\_PAY\_TYPE] [varchar](7) NULL,

[ADD\_AMOUNT] [decimal](19, 2) NULL,

[ADD\_RESERVED] [varchar](1) NULL,

[ADD\_PAY\_DATE] [varchar](19) NULL,

[ADD\_CR\_ACCOUNT] [varchar](16) NULL,

[ADD\_NEW\_AMOUNT] [decimal](19, 2) NULL,

[ADD\_NEW\_DATE] [varchar](8) NULL,

[ADD\_END\_DATE] [varchar](8) NULL,

[ADD\_REFERENCE] [varchar](35) NULL,

[ADD\_THIRD\_PARTY] [varchar](17) NULL,

[ADD\_USER\_REF\_1] [varchar](35) NULL,

[CHARGE\_DATE] [varchar](8) NULL,

[CHARGE\_CODE] [varchar](11) NULL,

[CHARGE\_CCY] [varchar](3) NULL,

[CHARGE\_AMOUNT] [varchar](65) NULL,

[CHARGE\_ACCOUNT] [varchar](16) NULL,

[CHARGE\_AMT\_LCY] [varchar](65) NULL,

[CHARGE\_TAX\_AMT] [varchar](65) NULL,

[CHARGE\_TAX\_LCY] [varchar](65) NULL,

[CHARGE\_TAX\_CODE] [varchar](2) NULL,

[TOTAL\_CHARGE] [decimal](19, 2) NULL,

[TOTAL\_CHG\_TAX] [decimal](19, 2) NULL,

[PORTFOLIO\_NUMBER] [varchar](3) NULL,

[COLLATERAL\_CODE\_1] [varchar](2) NULL,

[COLLATERAL\_CODE\_2] [varchar](2) NULL,

[OUR\_REMARKS] [varchar](35) NULL,

[MIS\_INTEREST\_KEY] [varchar](2) NULL,

[MIS\_INTEREST\_RATE] [decimal](9, 6) NULL,

[MIS\_ACCT\_OFFICER] [varchar](10) NULL,

[FIRST\_DAY\_ACCRUAL] [varchar](3) NULL,

[PAYMENT\_REFERENCE] [varchar](10) NULL,

[CLOSURE\_DATE] [varchar](8) NULL,

[LINK\_REFERENCE] [varchar](12) NULL,

[JOINT\_HOLDER] [varchar](65) NULL,

[RELATION\_CODE] [varchar](20) NULL,

[JOINT\_NOTES] [varchar](35) NULL,

[MI\_FUNDING\_DEPT] [varchar](4) NULL,

[TAX\_E\_I] [varchar](1) NULL,

[TAX\_ON] [varchar](2) NULL,

[TAX\_TYPE\_KEY] [varchar](10) NULL,

[TAX\_RATE] [decimal](9, 6) NULL,

[TAX\_KEY] [varchar](2) NULL,

[COMM\_TAX] [decimal](19, 2) NULL,

[ADVICE\_TYPE] [varchar](4) NULL,

[DELIVERY\_REF] [varchar](20) NULL,

[CURRENCY\_MARKET] [varchar](1) NULL,

[POSITION\_TYPE] [varchar](2) NULL,

[CONSOL\_KEY] [varchar](70) NULL,

[FIRST\_INT\_PAY] [varchar](19) NULL,

[APPLIED\_TAX\_KEY] [varchar](2) NULL,

[TERM\_PRIORITY] [varchar](4) NULL,

[PRIN\_INCREASE] [decimal](19, 2) NULL,

[PRIN\_EFF\_DATE] [varchar](8) NULL,

[RECALC\_CURR\_AMT] [varchar](3) NULL,

[NEW\_TERM] [varchar](35) NULL,

[MAX\_INT\_RATE] [decimal](9, 6) NULL,

[MIN\_INT\_RATE] [decimal](9, 6) NULL,

[INITIAL\_RATE] [decimal](9, 6) NULL,

[MAX\_INT\_TOL] [decimal](9, 6) NULL,

[MIN\_INT\_TOL] [decimal](9, 6) NULL,

[MAX\_REPAY\_AMT] [decimal](19, 2) NULL,

[MIN\_REPAY\_AMT] [decimal](19, 2) NULL,

[INIT\_REPAY\_AMT] [decimal](19, 2) NULL,

[MAX\_REPAY\_TOL] [decimal](9, 6) NULL,

[MIN\_REPAY\_TOL] [decimal](9, 6) NULL,

[RESERVED\_4] [varchar](35) NULL,

[RESERVED\_3] [varchar](35) NULL,

[RESERVED\_2] [varchar](35) NULL,

[RESERVED\_1] [varchar](35) NULL,

[STMT\_NO] [varchar](20) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](8) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](8) NULL,

[DEF\_INT\_CAP] [varchar](8) NULL,

[MIS\_COF\_TYPE] [varchar](35) NULL,

[MIS\_COF\_RATE] [decimal](9, 6) NULL,

[MIS\_COF\_DEPT] [varchar](35) NULL,

[DD\_EXCG\_RATE] [decimal](9, 6) NULL,

[INT\_LIQ\_ACCOUNT] [varchar](16) NULL,

[REVOLVING] [varchar](1) NULL,

[SECURED] [varchar](1) NULL,

[OLD\_CONTRACT] [varchar](27) NULL,

[SCHEDULE\_N] [varchar](1) NULL,

[RESTRUCTURED\_YN] [varchar](1) NULL,

[PROPERTY\_ADDR] [varchar](65) NULL,

[BLOCK\_LOT] [varchar](12) NULL,

[PROPERTY\_TYPE] [varchar](2) NULL,

[CENSUS\_TRACK] [varchar](7) NULL,

[MSA] [varchar](5) NULL,

[MAX\_RATE\_INCR] [decimal](9, 6) NULL,

[MAX\_RATE\_DECR] [decimal](9, 6) NULL,

[PREV\_YR\_PAY\_TYP] [varchar](2) NULL,

[PREV\_YR\_AMOUNT] [decimal](19, 2) NULL,

[PRV\_YR\_DSB\_DT] [varchar](8) NULL,

[NXT\_YR\_PAY\_TYPE] [varchar](2) NULL,

[NXT\_YR\_PAY\_AMT] [decimal](19, 2) NULL,

[NXT\_YR\_DISB\_DTE] [varchar](8) NULL,

[REQ\_ESC\_ACT\_BAL] [decimal](19, 2) NULL,

[ESCROW\_ACCOUNT] [varchar](12) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[BROKER\_CODE] [varchar](10) NULL,

[BROKERAGE] [decimal](19, 2) NULL,

[BROKER\_CCY] [varchar](3) NULL,

[BROKER\_AMOUNT] [decimal](19, 2) NULL,

[NEW\_CUSTOMER] [varchar](10) NULL,

[OLD\_CUSTOMER] [varchar](10) NULL,

[REPAY\_METHOD] [varchar](2) NULL,

[DD\_MANDATE\_REF] [varchar](25) NULL,

[ASSET\_CLASS] [varchar](3) NULL,

[PROVISION] [varchar](11) NULL,

[PROVISION\_AMOUNT] [decimal](18, 2) NULL,

[PROVISION\_METHOD] [varchar](6) NULL,

[RESERVED\_6] [varchar](35) NULL,

[RESERVED\_5] [varchar](35) NULL,

[FWD\_BWD] [varchar](3) NULL,

[USER\_DEF\_INT] [varchar](3) NULL,

[OTHER\_REF] [varchar](35) NULL,

[THEIR\_REF] [varchar](35) NULL,

[MG\_VAR\_IND] [varchar](200) NULL,

[INT\_RATE\_BASE] [varchar](200) NULL,

[MARGIN\_RATE] [varchar](200) NULL,

[LOAN\_USAGE] [varchar](200) NULL,

[LOAN\_LOC] [varchar](200) NULL,

[REPRI\_DATE] [varchar](200) NULL,

[PROD\_TYPE] [varchar](200) NULL,

[LN\_REPAY\_TYPE] [varchar](200) NULL,

[TENOR] [varchar](200) NULL,

[LTV\_RATIO] [varchar](200) NULL,

[TERM\_DAY] [varchar](200) NULL,

[FIRST\_LEND\_DATE] [varchar](200) NULL,

[REPRI\_FREQ] [varchar](200) NULL,

[ORIGINAL\_AMT] [varchar](200) NULL,

[ORIG\_DATE] [varchar](200) NULL,

[COL\_TYPE] [varchar](300) NULL,

[GUARANTOR\_CIF] [varchar](300) NULL,

[COL\_INDICT] [varchar](300) NULL,

[S\_LD\_TYPE] [varchar](300) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[MULTI\_LD\_SCHEDULE\_DEFINE\_DETAIL] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[MULTI\_LD\_SCHEDULE\_DEFINE\_DETAIL](

[ID] [varchar](12) NOT NULL,

[SCH\_TYPE] [varchar](3) NOT NULL,

[DATE] [varchar](8) NOT NULL,

[CYCLED\_DATES] [varchar](3300) NULL,

[AMOUNT] [decimal](19, 2) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[MULTI\_LD\_SCHEDULE\_DEFINE] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[MULTI\_LD\_SCHEDULE\_DEFINE](

[ID] [varchar](12) NOT NULL,

[FORWARD\_BACKWARD] [varchar](1) NULL,

[BASE\_DATE\_KEY] [varchar](1) NULL,

[RATE] [varchar](50) NULL,

[CHARGE\_CODE] [varchar](3) NULL,

[NUMBER] [varchar](3) NULL,

[FREQUENCY] [varchar](5) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[CONTRACT\_ID] [varchar](12) NULL,

[CURRENCY] [varchar](3) NULL,

[DIARY\_ACTION] [varchar](35) NULL,

[NOTE\_DENOM] [varchar](50) NULL,

[RESERVED6] [varchar](55) NULL,

[RESERVED5] [varchar](55) NULL,

[RESERVED4] [varchar](55) NULL,

[RESERVED3] [varchar](55) NULL,

[RESERVED2] [varchar](55) NULL,

[RESERVED1] [varchar](55) NULL,

[NOTE\_QUANT] [varchar](11) NULL,

[CHG\_BASE\_AMT] [varchar](2) NULL,

[FREQ\_CODE] [varchar](5) NULL,

[INCLUSIVE\_CHG] [varchar](3) NULL,

[BILL\_NO] [varchar](15) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[CAP\_INT] [varchar](40) NULL,

[SCHED\_DATE] [varchar](36) NULL,

[SCHED\_TYPE] [varchar](30) NULL,

[SCHED\_AMT\_DUE] [varchar](60) NULL,

[ADVICE\_TYPE] [varchar](30) NULL,

[ADVICE\_DATE] [varchar](36) NULL,

[SCHED\_AMT] [varchar](60) NULL,

[PREV\_SCHED\_TYPE] [varchar](6) NULL,

[PREV\_SCHED\_DATE] [varchar](36) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[MULTI\_LD\_BALANCE] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[MULTI\_LD\_BALANCE](

[ID] [varchar](50) NULL,

[DATE] [varchar](50) NULL,

[TOTAL\_PAYMENT] [decimal](19, 2) NULL,

[PRINCIPAL] [decimal](19, 2) NULL,

[INTEREST] [decimal](19, 2) NULL,

[COMMISSION] [decimal](19, 2) NULL,

[FEES] [decimal](19, 2) NULL,

[CHARGES] [decimal](19, 2) NULL,

[OUTSTANDING\_BAL] [decimal](19, 2) NULL,

[CO\_CODE] [varchar](10) NULL,

[COMPANY] [varchar](3) NULL,

[ASOFDATE] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[MULTI\_LD] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[MULTI\_LD](

[ID] [varchar](25) NOT NULL,

[CONTRACT\_NO] [varchar](25) NULL,

[CUSTOMER\_ID] [varchar](10) NULL,

[CURRENCY] [varchar](3) NULL,

[CURRENCY\_MARKET] [varchar](2) NULL,

[AMOUNT] [decimal](19, 2) NULL,

[BUS\_DAY\_DEFN] [varchar](4) NULL,

[VALUE\_DATE] [varchar](8) NULL,

[FIN\_MAT\_DATE] [varchar](8) NULL,

[LIMIT\_REFERENCE] [varchar](10) NULL,

[THIRD\_PARTY\_REF] [varchar](16) NULL,

[CUSTOMER\_REF] [varchar](16) NULL,

[CATEGORY] [varchar](6) NULL,

[DRAWDOWN\_ACCOUNT] [varchar](16) NULL,

[INT\_RATE\_TYPE] [varchar](20) NULL,

[INTEREST\_BASIS] [varchar](10) NULL,

[INT\_PAYMT\_METHOD] [varchar](18) NULL,

[INTEREST\_RATE] [decimal](9, 6) NULL,

[INTEREST\_KEY] [varchar](4) NULL,

[INTEREST\_SPREAD] [varchar](12) NULL,

[FIRST\_DAY\_ACCRUAL] [varchar](3) NULL,

[INT\_REF\_NUMBER] [varchar](5) NULL,

[MIS\_INTEREST\_KEY] [varchar](3) NULL,

[MIS\_INTEREST\_RATE] [decimal](9, 6) NULL,

[CAPITALISATION] [varchar](3) NULL,

[TOT\_INTEREST\_AMT] [decimal](19, 2) NULL,

[INT\_START\_DATE] [varchar](8) NULL,

[INT\_END\_DATE] [varchar](8) NULL,

[BAND\_LEVEL\_IND] [varchar](1) NULL,

[TAX\_PRINCIPAL\_KEY] [varchar](2) NULL,

[TAX\_COMMISSION\_KEY] [varchar](2) NULL,

[W\_H\_TAX\_ALT] [varchar](2) NULL,

[W\_H\_TAX\_RATE\_KEY] [varchar](2) NULL,

[LIQUIDATION\_CODE] [varchar](18) NULL,

[COMM\_CALC\_METHOD] [varchar](10) NULL,

[COMM\_PAYMT\_METHOD] [varchar](18) NULL,

[COMM\_RATE] [decimal](9, 6) NULL,

[TOTAL\_COMM\_AMOUNT] [decimal](19, 2) NULL,

[CHRG\_CODE] [varchar](12) NULL,

[CHRG\_AMOUNT] [varchar](66) NULL,

[CHRG\_CLAIM\_DATE] [varchar](27) NULL,

[CHRG\_BOOKED\_ON] [varchar](27) NULL,

[CHRG\_CAPITALISE] [varchar](12) NULL,

[PENALTY\_KEY] [varchar](3) NULL,

[PENALTY\_SPREAD] [varchar](11) NULL,

[POSITION\_TYPE] [varchar](2) NULL,

[NEG\_REF\_NUMBER] [varchar](5) NULL,

[PRINC\_FX\_RATE] [decimal](9, 6) NULL,

[V\_DATE\_FOR\_P\_FX] [varchar](8) NULL,

[OTHER\_NEG\_REF] [varchar](5) NULL,

[OTHER\_FX\_RATE] [decimal](9, 6) NULL,

[V\_DATE\_OTHER\_FX] [varchar](8) NULL,

[MAX\_CONTRACT\_RATE] [decimal](9, 6) NULL,

[MIN\_CONTRACT\_RATE] [decimal](9, 6) NULL,

[ACCRUAL\_RATE] [decimal](9, 6) NULL,

[REDUCED\_RATE] [varchar](7) NULL,

[DELIVERY\_LINK] [varchar](3) NULL,

[WALK\_IN\_CUST] [varchar](35) NULL,

[PRIN\_LIQ\_ACCT] [varchar](16) NULL,

[PRIN\_BEN\_BANK\_1] [varchar](10) NULL,

[PRIN\_BEN\_BANK\_2] [varchar](10) NULL,

[PRIN\_ADDRESS] [varchar](35) NULL,

[BENEF\_PRIN\_ACCT] [varchar](35) NULL,

[INT\_LIQ\_ACCT] [varchar](16) NULL,

[INT\_BEN\_BANK\_1] [varchar](10) NULL,

[INT\_BEN\_BANK\_2] [varchar](10) NULL,

[INT\_ADDRESS] [varchar](35) NULL,

[BEN\_INT\_ACCT] [varchar](35) NULL,

[COM\_LIQ\_ACCT] [varchar](16) NULL,

[CHRG\_LIQ\_ACCT] [varchar](16) NULL,

[MIS\_ACCT\_OFFICER] [varchar](10) NULL,

[FEDERAL\_FUNDS] [varchar](3) NULL,

[CLEAN\_RISK] [varchar](3) NULL,

[RESERVES] [decimal](19, 2) NULL,

[SECURED\_Y\_N] [varchar](3) NULL,

[AGREEMENT\_DATE] [varchar](8) NULL,

[CROSS\_REF\_NO] [varchar](12) NULL,

[COUNTRY\_RISK] [varchar](2) NULL,

[COUNTRY\_EXPOSURE] [varchar](2) NULL,

[CENTRAL\_BANK\_CODE] [varchar](10) NULL,

[STATUS\_CONTROL] [varchar](9) NULL,

[STATUS] [varchar](3) NULL,

[COMMITMENT\_NO] [varchar](14) NULL,

[L\_C\_U\_TYPE] [varchar](25) NULL,

[STA\_Y\_N] [varchar](3) NULL,

[L\_C\_U\_CONVERSION] [decimal](9, 6) NULL,

[L\_C\_U\_CALC\_BASE] [varchar](17) NULL,

[COMMT\_AVAIL\_AMT] [decimal](19, 2) NULL,

[D\_D\_START\_DATE] [varchar](8) NULL,

[D\_D\_END\_DATE] [varchar](8) NULL,

[D\_D\_MULT\_AMT] [decimal](19, 2) NULL,

[PARTICIPANT\_NO] [varchar](10) NULL,

[PRINC\_DIS\_ACCT] [varchar](16) NULL,

[AMOUNT\_INCREASE] [decimal](19, 2) NULL,

[AMT\_V\_DATE] [varchar](8) NULL,

[NEW\_INT\_RATE] [decimal](9, 6) NULL,

[INT\_RATE\_V\_DATE] [varchar](8) NULL,

[NEW\_SPREAD] [varchar](12) NULL,

[SPREAD\_V\_DATE] [varchar](8) NULL,

[NEW\_COM\_RATE] [decimal](9, 6) NULL,

[COM\_RATE\_V\_DATE] [varchar](8) NULL,

[BK\_TO\_BK\_INFO] [varchar](35) NULL,

[LINK\_REFERENCE] [varchar](12) NULL,

[CUST\_REMARKS] [varchar](35) NULL,

[ALONE\_MERGE\_IND] [varchar](5) NULL,

[OUR\_REMARKS] [varchar](35) NULL,

[AUTO\_SCHEDS] [varchar](3) NULL,

[DEFINE\_SCHEDS] [varchar](3) NULL,

[INTEREST\_DUE\_DATE] [varchar](8) NULL,

[COMM\_DUE\_DATE] [varchar](8) NULL,

[INT\_REVISION\_DATE] [varchar](8) NULL,

[MARGIN\_REV\_DATE] [varchar](8) NULL,

[PORTFOLIO\_NO] [varchar](3) NULL,

[ACTIVITY] [varchar](8) NULL,

[ADV\_SENT] [varchar](25) NULL,

[TOT\_INT\_TAX] [decimal](19, 2) NULL,

[TOTAL\_W\_H\_T] [decimal](19, 2) NULL,

[TOT\_COM\_TAX] [decimal](19, 2) NULL,

[TOT\_PRINC\_TAX] [decimal](19, 2) NULL,

[BROKER\_CODE] [varchar](10) NULL,

[BROKERAGE] [decimal](19, 2) NULL,

[BROKER\_CCY] [varchar](3) NULL,

[BROKER\_AMOUNT] [decimal](19, 2) NULL,

[STMT\_NO] [varchar](29) NULL,

[OVERRIDE] [varchar](500) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](15) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[DRAWDOWN\_ISSUE\_PRC] [varchar](19) NULL,

[DD\_FEE\_CODE] [varchar](11) NULL,

[DD\_FEE\_AMT] [decimal](19, 2) NULL,

[DRAWDOWN\_NET\_AMT] [decimal](19, 2) NULL,

[VD\_DATE\_ADJUSTMENT] [varchar](20) NULL,

[TOTAL\_CAP\_CHARGE] [decimal](19, 2) NULL,

[ISSUE\_PL\_AMOUNT] [decimal](19, 2) NULL,

[LIQUIDATION\_MODE] [varchar](15) NULL,

[OVERDUE\_STATUS] [varchar](3) NULL,

[VD\_PRIOR\_ADJUST] [varchar](3) NULL,

[COMMIT\_DRAWING] [varchar](2) NULL,

[GRACE\_PERIOD] [varchar](2) NULL,

[GRACE\_AMOUNT] [decimal](19, 2) NULL,

[SEND\_PAYMENT] [varchar](2) NULL,

[SEND\_CONFIRMATION] [varchar](2) NULL,

[COM\_RESERVE\_1] [varchar](10) NULL,

[COM\_RESERVE\_2] [varchar](10) NULL,

[COM\_RESERVE\_3] [varchar](16) NULL,

[COM\_RESERVE\_4] [varchar](16) NULL,

[COM\_RESERVE\_5] [varchar](10) NULL,

[FEE\_PAY\_ACCOUNT] [varchar](16) NULL,

[FEE\_RESERVE4] [varchar](16) NULL,

[FEE\_RESERVE3] [varchar](16) NULL,

[FEE\_RESERVE2] [varchar](16) NULL,

[FEE\_RESERVE1] [varchar](16) NULL,

[CONTRACT\_TYPE] [varchar](6) NULL,

[MIS\_FUNDING\_DEPT] [varchar](4) NULL,

[PAYING\_AGENT] [varchar](10) NULL,

[TRADE\_DATE] [varchar](8) NULL,

[NOTE\_DENOM] [decimal](19, 2) NULL,

[NOTE\_QUANTITY] [decimal](19, 2) NULL,

[ISSUE\_PRICE] [decimal](9, 6) NULL,

[ISSUE\_ACCRUAL] [varchar](9) NULL,

[REIMBURSE\_PRICE] [decimal](19, 15) NULL,

[REIMBURSE\_AMOUNT] [decimal](19, 2) NULL,

[REIMBURSE\_PL\_AMT] [decimal](19, 2) NULL,

[REIMBURSE\_ACCRUAL] [varchar](9) NULL,

[DRAWDOWN\_ENT\_DATE] [varchar](8) NULL,

[CANCELLATION\_DATE] [varchar](8) NULL,

[CANCELLED\_INT] [decimal](19, 2) NULL,

[TOT\_CANCEL\_INT\_AMT] [decimal](19, 2) NULL,

[JOINT\_HOLDER] [varchar](65) NULL,

[RELATION\_CODE] [varchar](20) NULL,

[JOINT\_NOTES] [varchar](35) NULL,

[ANNUITY\_PAY\_METHOD] [varchar](5) NULL,

[RESIDUAL\_VALUE] [decimal](19, 2) NULL,

[ANNUITY\_REPAY\_AMT] [decimal](19, 2) NULL,

[TAX\_PRINCIPAL\_TYPE] [varchar](17) NULL,

[TAX\_COMMISSN\_TYPE] [varchar](17) NULL,

[CONTRACT\_GRP] [varchar](5) NULL,

[BILL\_NUMBER] [varchar](12) NULL,

[BILL\_REDISC\_RATE] [decimal](9, 6) NULL,

[BILL\_TXN\_AMOUNT] [decimal](19, 2) NULL,

[USE\_DEF\_PI\_RATE] [varchar](3) NULL,

[PRIN\_INC\_LIMAMT] [varchar](100) NULL,

[INCLUSIVE\_CSN] [varchar](3) NULL,

[CSN\_CALC\_TYPE] [varchar](2) NULL,

[ASSET\_REFERENCE] [varchar](12) NULL,

[FIN\_CHG\_CALC\_TYPE] [varchar](2) NULL,

[DAYS\_PRE\_ADVICE] [varchar](3) NULL,

[FV\_RATE\_KEY] [varchar](4) NULL,

[FV\_MARGIN\_KEY] [varchar](4) NULL,

[NET\_INT\_TAX\_LCY] [decimal](19, 2) NULL,

[NET\_INT\_TAX\_FCY] [decimal](19, 2) NULL,

[RETURN\_INT] [varchar](3) NULL,

[RETURN\_COM] [varchar](3) NULL,

[HB\_CURR\_PERIOD] [decimal](19, 2) NULL,

[FLAT\_COMM\_TYPE] [varchar](3) NULL,

[YIELD\_METHOD] [varchar](3) NULL,

[MAT\_AMOUNT] [decimal](19, 2) NULL,

[MATURE\_AT\_SOD] [varchar](3) NULL,

[MATURED\_ONLINE] [varchar](3) NULL,

[EXCG\_RATE] [decimal](9, 6) NULL,

[SETTLEMENT\_MARKET] [varchar](2) NULL,

[CONVERSION\_TYPE] [varchar](4) NULL,

[CURRENCY\_CODE] [varchar](3) NULL,

[TRANCHE\_AMT] [decimal](19, 2) NULL,

[REV\_CONTINGENT] [varchar](4) NULL,

[LAST\_DAY\_ACCRUAL] [varchar](5) NULL,

[DEFAULTED\_VALUE] [varchar](3) NULL,

[FWD\_BWD\_KEY] [varchar](3) NULL,

[BASE\_DATE\_KEY] [varchar](4) NULL,

[MIS\_COF\_TYPE] [varchar](35) NULL,

[MIS\_COF\_RATE] [decimal](9, 6) NULL,

[MIS\_COF\_DEPT] [varchar](35) NULL,

[RESERVED\_1] [varchar](35) NULL,

[RESERVED\_2] [varchar](35) NULL,

[RED\_RECOURSE] [varchar](3) NULL,

[THIRD\_PARTY\_ID] [varchar](10) NULL,

[THIRD\_LIM\_REF] [varchar](10) NULL,

[DEALER\_DESK] [varchar](2) NULL,

[NEGATIVE\_RATE] [varchar](3) NULL,

[PENALTY\_RATE] [decimal](9, 6) NULL,

[REVOLVING] [varchar](1) NULL,

[SECURED] [varchar](1) NULL,

[OLD\_CONTRACT] [varchar](27) NULL,

[SYNDICATE\_LDR] [varchar](32) NULL,

[ARRANGER] [varchar](32) NULL,

[BUYER\_SELLER] [varchar](32) NULL,

[CUSIP\_NO] [varchar](9) NULL,

[SCHEDULE\_N] [varchar](1) NULL,

[COMMITTED\_YN] [varchar](1) NULL,

[DISCREPANCY\_YN] [varchar](1) NULL,

[RESTRUCTURED\_YN] [varchar](1) NULL,

[DISCOUNT\_TYPE] [varchar](1) NULL,

[REF\_1] [varchar](12) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[INT\_COLL\_METHOD] [decimal](15, 2) NULL,

[ASSET\_CLASS] [varchar](3) NULL,

[PROVISION] [varchar](11) NULL,

[PROVISION\_AMOUNT] [decimal](18, 2) NULL,

[PROVISION\_METHOD] [varchar](6) NULL,

[REPAY\_CAP] [varchar](3) NULL,

[FWD\_PROJ] [varchar](2) NULL,

[ACCRUAL\_PARAM] [varchar](35) NULL,

[ROUNDING\_RULE] [varchar](35) NULL,

[REIMB\_RESERVED] [varchar](6) NULL,

[ONLINE\_EFF\_DATE] [varchar](8) NULL,

[ONLINE\_RPY\_TYPE] [varchar](10) NULL,

[ONLINE\_RPY\_AMT] [varchar](60) NULL,

[TAX\_INT\_KEY] [varchar](10) NULL,

[TAX\_INT\_TYPE] [varchar](55) NULL,

[INT\_TAX\_LCY] [varchar](60) NULL,

[INT\_TAX\_FCY] [varchar](60) NULL,

[LD\_RESERVED] [varchar](7) NULL,

[IAS\_CLASSIFICATION] [varchar](35) NULL,

[IAS\_SUBTYPE] [varchar](35) NULL,

[MARKET\_KEY] [varchar](35) NULL,

[MARKET\_MARGIN] [varchar](12) NULL,

[SPECIAL\_INT\_BASIS] [varchar](10) NULL,

[SPECIAL\_COMM\_BASIS] [varchar](10) NULL,

[COMMT\_REINITIATE] [varchar](3) NULL,

[CURR\_PD\_PR\_AMT] [decimal](19, 2) NULL,

[COMPOUND\_YIELD\_METHOD] [varchar](8) NULL,

[COMPOUND\_TYPE] [varchar](5) NULL,

[NCB\_PRODUCT\_COD] [varchar](10) NULL,

[BOCUS\_ACCT\_NO] [varchar](17) NULL,

[INT\_RATE\_BASE] [varchar](200) NULL,

[MARGIN\_RATE] [varchar](200) NULL,

[LOAN\_USAGE] [varchar](200) NULL,

[LOAN\_LOC] [varchar](200) NULL,

[REPRI\_DATE] [varchar](200) NULL,

[PROD\_TYPE] [varchar](200) NULL,

[LN\_REPAY\_TYPE] [varchar](200) NULL,

[TENOR] [varchar](200) NULL,

[LTV\_RATIO] [varchar](200) NULL,

[TERM\_DAY] [varchar](200) NULL,

[FIRST\_LEND\_DATE] [varchar](200) NULL,

[REPRI\_FREQ] [varchar](200) NULL,

[ORIGINAL\_AMT] [varchar](200) NULL,

[ORIG\_DATE] [varchar](200) NULL,

[COL\_TYPE] [varchar](300) NULL,

[GUARANTOR\_CIF] [varchar](300) NULL,

[COL\_INDICT] [varchar](300) NULL,

[S\_LD\_TYPE] [varchar](300) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[MULTI\_ACCOUNT] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[MULTI\_ACCOUNT](

[ID] [varchar](16) NOT NULL,

[ACCOUNT\_NUMBER] [varchar](16) NULL,

[CUSTOMER] [varchar](10) NULL,

[CATEGORY] [varchar](6) NULL,

[ACCOUNT\_TITLE\_1] [varchar](35) NULL,

[ACCOUNT\_TITLE\_2] [varchar](35) NULL,

[SHORT\_TITLE] [varchar](35) NULL,

[MNEMONIC] [varchar](10) NULL,

[POSITION\_TYPE] [varchar](2) NULL,

[CURRENCY] [varchar](3) NULL,

[CURRENCY\_MARKET] [varchar](1) NULL,

[LIMIT\_REF] [varchar](10) NULL,

[ACCOUNT\_OFFICER] [varchar](10) NULL,

[OTHER\_OFFICER] [varchar](10) NULL,

[POSTING\_RESTRICT] [varchar](2) NULL,

[RECONCILE\_ACCT] [varchar](1) NULL,

[INTEREST\_LIQU\_ACCT] [varchar](16) NULL,

[INTEREST\_COMP\_ACCT] [varchar](16) NULL,

[INT\_NO\_BOOKING] [varchar](8) NULL,

[REFERAL\_CODE] [varchar](6) NULL,

[WAIVE\_LEDGER\_FEE] [varchar](1) NULL,

[CONDITION\_GROUP] [varchar](4) NULL,

[INACTIV\_MARKER] [varchar](1) NULL,

[OPEN\_ACTUAL\_BAL] [decimal](19, 2) NULL,

[OPEN\_CLEARED\_BAL] [decimal](19, 2) NULL,

[ONLINE\_ACTUAL\_BAL] [decimal](19, 2) NULL,

[ONLINE\_CLEARED\_BAL] [decimal](19, 2) NULL,

[WORKING\_BALANCE] [decimal](19, 2) NULL,

[DATE\_LAST\_CR\_CUST] [varchar](8) NULL,

[AMNT\_LAST\_CR\_CUST] [decimal](19, 2) NULL,

[TRAN\_LAST\_CR\_CUST] [varchar](3) NULL,

[DATE\_LAST\_CR\_AUTO] [varchar](8) NULL,

[AMNT\_LAST\_CR\_AUTO] [decimal](19, 2) NULL,

[TRAN\_LAST\_CR\_AUTO] [varchar](3) NULL,

[DATE\_LAST\_CR\_BANK] [varchar](8) NULL,

[AMNT\_LAST\_CR\_BANK] [decimal](19, 2) NULL,

[TRAN\_LAST\_CR\_BANK] [varchar](3) NULL,

[DATE\_LAST\_DR\_CUST] [varchar](8) NULL,

[AMNT\_LAST\_DR\_CUST] [decimal](19, 2) NULL,

[TRAN\_LAST\_DR\_CUST] [varchar](3) NULL,

[DATE\_LAST\_DR\_AUTO] [varchar](8) NULL,

[AMNT\_LAST\_DR\_AUTO] [decimal](19, 2) NULL,

[TRAN\_LAST\_DR\_AUTO] [varchar](3) NULL,

[DATE\_LAST\_DR\_BANK] [varchar](8) NULL,

[AMNT\_LAST\_DR\_BANK] [decimal](19, 2) NULL,

[TRAN\_LAST\_DR\_BANK] [varchar](3) NULL,

[CAP\_DATE\_CHARGE] [varchar](8) NULL,

[CAP\_DATE\_CR\_INT] [varchar](8) NULL,

[CAP\_DATE\_C2\_INT] [varchar](8) NULL,

[CAP\_DATE\_DR\_INT] [varchar](8) NULL,

[CAP\_DATE\_D2\_INT] [varchar](8) NULL,

[CAP\_BACK\_VALUE] [varchar](8) NULL,

[ACCR\_CHG\_CATEG] [varchar](6) NULL,

[ACCR\_CHG\_TRANS] [varchar](3) NULL,

[ACCR\_CHG\_AMOUNT] [decimal](19, 2) NULL,

[ACCR\_CHG\_SUSP] [decimal](19, 2) NULL,

[ACCR\_CR\_CATEG] [varchar](6) NULL,

[ACCR\_CR\_TRANS] [varchar](3) NULL,

[ACCR\_CR\_AMOUNT] [decimal](19, 2) NULL,

[ACCR\_CR\_SUSP] [decimal](19, 2) NULL,

[ACCR\_CR2\_CATEG] [varchar](6) NULL,

[ACCR\_CR2\_TRANS] [varchar](3) NULL,

[ACCR\_CR2\_AMOUNT] [decimal](19, 2) NULL,

[ACCR\_CR2\_SUSP] [decimal](19, 2) NULL,

[ACCR\_DR\_CATEG] [varchar](6) NULL,

[ACCR\_DR\_TRANS] [varchar](3) NULL,

[ACCR\_DR\_AMOUNT] [decimal](19, 2) NULL,

[ACCR\_DR\_SUSP] [decimal](19, 2) NULL,

[ACCR\_DR2\_CATEG] [varchar](6) NULL,

[ACCR\_DR2\_TRANS] [varchar](3) NULL,

[ACCR\_DR2\_AMOUNT] [decimal](19, 2) NULL,

[ACCR\_DR2\_SUSP] [decimal](19, 2) NULL,

[CONSOL\_KEY] [varchar](65) NULL,

[BALANCE\_CCY\_MKT] [varchar](1) NULL,

[MKT\_ACTUAL\_BAL] [varchar](255) NULL,

[MKT\_CLEAR\_BAL] [varchar](255) NULL,

[PASSBOOK] [varchar](2) NULL,

[START\_YEAR\_BAL] [decimal](19, 2) NULL,

[OPENING\_DATE] [varchar](8) NULL,

[OVERRIDE] [varchar](35) NULL,

[RECORD\_STATUS] [varchar](4) NULL,

[CURR\_NO] [varchar](4) NULL,

[INPUTTER] [varchar](35) NULL,

[DATE\_TIME] [varchar](10) NULL,

[AUTHORISER] [varchar](20) NULL,

[CO\_CODE] [varchar](11) NULL,

[DEPT\_CODE] [varchar](10) NULL,

[AUDITOR\_CODE] [varchar](16) NULL,

[AUDIT\_DATE\_TIME] [varchar](15) NULL,

[VALUE\_DATE] [varchar](255) NULL,

[CREDIT\_MOVEMENT] [varchar](255) NULL,

[DEBIT\_MOVEMENT] [varchar](255) NULL,

[VALUE\_DATED\_BAL] [varchar](255) NULL,

[CONTINGENT\_BAL\_CR] [decimal](19, 2) NULL,

[CONTINGENT\_BAL\_DR] [decimal](19, 2) NULL,

[OPEN\_CATEGORY] [varchar](6) NULL,

[OPEN\_VAL\_DATED\_BAL] [decimal](19, 2) NULL,

[ACCOUNT\_CREDIT\_INT] [varchar](8) NULL,

[ACCOUNT\_DEBIT\_INT] [varchar](8) NULL,

[PREV\_OPEN\_BAL] [decimal](19, 2) NULL,

[CHARGE\_ACCOUNT] [varchar](16) NULL,

[LINK\_TO\_LIMIT] [varchar](3) NULL,

[CLOSURE\_DATE] [varchar](8) NULL,

[CHARGE\_CCY] [varchar](3) NULL,

[CHARGE\_MKT] [varchar](1) NULL,

[INTEREST\_CCY] [varchar](3) NULL,

[INTEREST\_MKT] [varchar](1) NULL,

[CON\_CHARGE\_ACCR] [varchar](65) NULL,

[CON\_INTEREST\_ACCR] [varchar](65) NULL,

[ALT\_ACCT\_TYPE] [varchar](9) NULL,

[ALT\_ACCT\_ID] [varchar](16) NULL,

[PREMIUM\_TYPE] [varchar](16) NULL,

[APR] [decimal](9, 6) NULL,

[CAP\_DATE\_PRM] [varchar](8) NULL,

[PREMIUM\_FREQ] [varchar](16) NULL,

[JOINT\_HOLDER] [varchar](255) NULL,

[RELATION\_CODE] [varchar](255) NULL,

[JOINT\_NOTES] [varchar](35) NULL,

[ALLOW\_NETTING] [varchar](3) NULL,

[LEDG\_RECO\_WITH] [varchar](16) NULL,

[STMT\_RECO\_WITH] [varchar](16) NULL,

[OUR\_EXT\_ACCT\_NO] [varchar](35) NULL,

[RECO\_TOLERANCE] [decimal](19, 2) NULL,

[PENDING\_ID] [varchar](12) NULL,

[TOTAL\_PENDING] [decimal](19, 2) NULL,

[SERIAL\_NO\_FORMAT] [varchar](16) NULL,

[STOCK\_CONTROL\_TYPE] [varchar](6) NULL,

[AUTO\_PAY\_ACCT] [varchar](16) NULL,

[ORIG\_CCY\_PAYMENT] [varchar](3) NULL,

[AUTO\_REC\_CCY] [varchar](3) NULL,

[ORIGINAL\_ACCT] [varchar](16) NULL,

[FROM\_DATE] [varchar](50) NULL,

[LOCKED\_AMOUNT] [varchar](128) NULL,

[DISPO\_OFFICER] [varchar](10) NULL,

[ICA\_MAIN\_ACCOUNT] [varchar](16) NULL,

[ICA\_DISTRIB\_RATIO] [varchar](3) NULL,

[ICA\_MAIN\_ACCT\_IND] [varchar](3) NULL,

[ICA\_DISTRIB\_TYPE] [varchar](10) NULL,

[ICA\_POST\_INTEREST] [varchar](4) NULL,

[ICA\_MAIN\_RATIO] [varchar](3) NULL,

[ICA\_NEW\_MAIN\_ACC] [varchar](16) NULL,

[ICA\_START\_DATE] [varchar](8) NULL,

[ICA\_ADD\_REMOVE] [varchar](6) NULL,

[ICA\_BACK\_VALUE] [varchar](8) NULL,

[ICA\_MAIN\_ACCT] [varchar](16) NULL,

[ICA\_MAIN\_DATE] [varchar](8) NULL,

[LIQUIDATION\_MODE] [varchar](14) NULL,

[OVERDUE\_STATUS] [varchar](3) NULL,

[SINGLE\_LIMIT] [varchar](1) NULL,

[CONTINGENT\_INT] [varchar](1) NULL,

[ALL\_IN\_ONE\_PRODUCT] [varchar](6) NULL,

[DISPO\_EXEMPT] [varchar](3) NULL,

[ER\_BALANCE] [decimal](19, 2) NULL,

[ER\_VALUE\_DATE] [varchar](8) NULL,

[SB\_GROUP\_ID] [varchar](18) NULL,

[OPEN\_AVAILABLE\_BAL] [decimal](19, 2) NULL,

[AVAILABLE\_DATE] [varchar](255) NULL,

[AV\_AUTH\_DB\_MVMT] [varchar](255) NULL,

[AV\_NAU\_DB\_MVMT] [varchar](255) NULL,

[AV\_AUTH\_CR\_MVMT] [varchar](255) NULL,

[AV\_NAU\_CR\_MVMT] [varchar](255) NULL,

[AVAILABLE\_BAL] [varchar](255) NULL,

[CREDIT\_CHECK] [varchar](9) NULL,

[AVAILABLE\_BAL\_UPD] [varchar](7) NULL,

[TAX\_SUSPEND] [varchar](1) NULL,

[TAX\_AT\_SETTLE] [varchar](1) NULL,

[EP\_BALANCE] [decimal](19, 2) NULL,

[CONSOLIDATE\_ENT] [varchar](20) NULL,

[MAX\_SUB\_ACCOUNT] [varchar](2) NULL,

[MASTER\_ACCOUNT] [varchar](35) NULL,

[CLOSED\_ONLINE] [varchar](1) NULL,

[ACCT\_COND\_PRIOR] [varchar](35) NULL,

[ACCT\_PURPOSE] [varchar](255) NULL,

[ACC\_OTH\_PURPOSE] [varchar](35) NULL,

[EXPECT\_BALANCE] [decimal](19, 2) NULL,

[REMITTANCE\_TO] [varchar](35) NULL,

[REMITTANCE\_FROM] [varchar](35) NULL,

[ACCT\_SOURCE] [varchar](35) NULL,

[ACCT\_REASON] [varchar](35) NULL,

[ACCT\_REMARKS] [varchar](35) NULL,

[POST\_RST\_REASON] [varchar](35) NULL,

[AC\_BGC\_BRANCH] [varchar](1) NULL,

[BOCUS\_ACCT\_NO] [varchar](17) NULL,

[LAST\_ACTIV\_DATE] [varchar](8) NULL,

[REVOLVING\_LIMIT] [varchar](1) NULL,

[SECURED\_LIMIT] [varchar](2) NULL,

[VENDOR\_1099] [varchar](3) NULL,

[US\_BSA\_NOTE] [varchar](35) NULL,

[ACH\_PROCESS] [varchar](1) NULL,

[US\_LEGACY\_CR\_DT] [varchar](8) NULL,

[US\_LEGACY\_DR\_DT] [varchar](8) NULL,

[MIN\_BAL\_SWEEP] [varchar](3) NULL,

[PD\_STATUS] [varchar](5) NULL,

[GPS\_PAYNOPAY] [varchar](5) NULL,

[BOCUS\_OPEN\_DATE] [varchar](8) NULL,

[ASOFDATE] [datetime] NOT NULL,

[COMPANY] [varchar](3) NOT NULL,

[NEXT\_AF\_DATE] [varchar](12) NULL,

[NEXT\_ACCT\_CAP] [varchar](11) NULL,

[DATE\_LAST\_UPDATE] [varchar](11) NULL,

[NEXT\_STMT\_DATE] [varchar](36) NULL,

[EXPOSURE\_DATES] [varchar](255) NULL,

[NEXT\_EXP\_DATE] [varchar](11) NULL,

[ENTRY\_HOLD] [varchar](78) NULL,

[FWD\_ENTRY\_HOLD] [varchar](78) NULL,

[FIRST\_AF\_DATE] [varchar](12) NULL,

[CASH\_POOL\_GROUP] [varchar](20) NULL,

[OPEN\_ASSET\_TYPE] [varchar](10) NULL,

[PORTFOLIO\_NO] [varchar](57) NULL,

[LAST\_COM\_CHG\_DATE] [varchar](11) NULL,

[IC\_CHARGE\_ID] [varchar](11) NULL,

[IC\_NEXT\_CAP\_DATE] [varchar](11) NULL,

[IC\_PRODUCT] [varchar](100) NULL,

[IC\_LST\_PROD\_CAP] [varchar](36) NULL,

[ACC\_DEB\_LIMIT] [varchar](36) NULL,

[ACCT\_CREDIT\_INT] [varchar](4000) NULL,

[ACCT\_DEBIT\_INT] [varchar](4000) NULL,

[FORWARD\_MVMTS] [varchar](1000) NULL,

[ARRANGEMENT\_ID] [varchar](17) NULL,

[PRODCAT] [varchar](6) NULL,

[INT\_LIQU\_TYPE] [varchar](15) NULL,

[INT\_LIQU\_ACCT] [varchar](55) NULL,

[INT\_LIQ\_CCY] [varchar](15) NULL,

[RESERVED11] [varchar](55) NULL,

[MANDATE\_APPL] [varchar](120) NULL,

[MANDATE\_REG] [varchar](120) NULL,

[MANDATE\_RECORD] [varchar](120) NULL,

[DR\_ADJ\_AMOUNT] [decimal](19, 2) NULL,

[DR2\_ADJ\_AMOUNT] [decimal](19, 2) NULL,

[CR\_ADJ\_AMOUNT] [decimal](19, 2) NULL,

[CR2\_ADJ\_AMOUNT] [decimal](19, 2) NULL,

[EVENT] [varchar](120) NULL,

[FIELD] [varchar](100) NULL,

[OPERAND] [varchar](120) NULL,

[VALUE] [varchar](120) NULL,

[REQUEST\_ID] [varchar](120) NULL,

[MV\_ALERT\_RES6] [varchar](150) NULL,

[MV\_ALERT\_RES5] [varchar](150) NULL,

[MV\_ALERT\_RES4] [varchar](150) NULL,

[MV\_ALERT\_RES3] [varchar](150) NULL,

[MV\_ALERT\_RES2] [varchar](150) NULL,

[MV\_ALERT\_RES1] [varchar](150) NULL,

[LOCKED\_WITH\_LIMIT] [varchar](3) NULL,

[LOCK\_INC\_THIS\_MVMT] [varchar](7) NULL,

[NCB\_PRODUCT\_COD] [varchar](10) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Map\_Tgt\_Agg\_BOC] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING OFF

GO

CREATE TABLE [app].[Map\_Tgt\_Agg\_BOC](

[mapID] [int] IDENTITY(1,1) NOT NULL,

[TgtID] [int] NULL,

[AggBOCID] [int] NULL,

[TargetTableName] [varchar](255) NULL,

[TargetFieldName] [varchar](255) NULL,

[AggregationReportName] [varchar](255) NULL,

[AggregationPortfolioName] [varchar](255) NULL,

[ProductID] [varchar](255) NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Map\_Tgt\_Agg] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Map\_Tgt\_Agg](

[mapID] [int] IDENTITY(1,1) NOT NULL,

[TgtID] [int] NULL,

[Agg\_2052ID] [int] NULL,

[TargetTableName] [varchar](255) NULL,

[TargetFieldName] [varchar](255) NULL,

[AggregationReportName] [varchar](255) NULL,

[AggregationPortfolioName] [varchar](255) NULL,

[ProductID] [varchar](255) NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[Map\_Src\_Tgt] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[Map\_Src\_Tgt](

[MapID] [int] IDENTITY(1,1) NOT NULL,

[SourceID] [varchar](4000) NULL,

[TargetID] [varchar](255) NULL,

[SourceDatabaseName] [varchar](255) NULL,

[SourceTableName] [varchar](255) NULL,

[SourceFieldName] [varchar](255) NULL,

[TargetReportName] [varchar](255) NULL,

[TargetPortfolioName] [varchar](255) NULL,

[Product] [varchar](255) NULL,

[AsofDate] [datetime] NULL,

[CreationDate] [datetime] NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[IssueTracer] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING OFF

GO

CREATE TABLE [app].[IssueTracer](

[issueID] [int] IDENTITY(1,1) NOT NULL,

[issueNumber] [int] NULL,

[issueState] [int] NULL,

[issueSeverity] [int] NULL,

[issueLine] [int] NULL,

[issueProc] [varchar](max) NULL,

[issueMsg] [varchar](max) NULL,

[UserName] [varchar](max) NULL,

[HostName] [varchar](max) NULL,

[issueDate] [datetime] NULL,

[ExecutionStatus] [varchar](20) NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [SRC].[HOLIDAYS] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [SRC].[HOLIDAYS](

[ID] [varchar](8) NOT NULL,

[Month] [varchar](2) NOT NULL,

[Days] [varchar](31) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[GlobalVariable] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[GlobalVariable](

[Variable\_ID] [int] IDENTITY(1,1) NOT NULL,

[Asofdate] [varchar](255) NULL,

[CreationDate] [varchar](255) NULL

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [app].[FRValidationTracer] Script Date: 07/11/2017 22:35:56 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [app].[FRValidationTracer](

[ID] [int] IDENTITY(1,1) NOT NULL,

[FRValidationID] [int] NULL,

[ResultType] [varchar](max) NULL,

[Logic] [varchar](max) NULL,

[ReportType] [varchar](max) NULL,

[PID] [varchar](max) NULL,

[UserName] [varchar](max) NULL,

[HostName] [varchar](max) NULL,

[IssueDate] [datetime] NULL,

[asofdate] [date] NULL,

[creationDate] [datetime] NULL,

[description] [varchar](max) NULL,

[AggFR2052aID] [int] NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[fn\_MaturityDateCalculater] Script Date: 07/11/2017 22:35:59 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE FUNCTION [dbo].[fn\_MaturityDateCalculater]

(@AsOfDate date, @MaturityDate Date, @Logic Varchar(20))

RETURNS Varchar (20)

AS

BEGIN

DECLARE @MatStartEnd VARCHAR(20)

--DECLARE variables for step by step calculation

DECLARE @CALC VARCHAR(20)

DECLARE @CALC1 VARCHAR(20)

DECLARE @CALC2 INT

-- logic for bucket 'open

IF @Logic = 'Open'

BEGIN

SET @MatStartEnd = 'P0D - P0D'

END

ELSE

--Logic for day 0 to day 30

IF @Logic LIKE 'DAY %' AND CAST((SELECT RIGHT(@Logic,2)) AS INT) < = 60

BEGIN

Set @MatStartEnd = 'P'+ (substring (@Logic,CHARINDEX(' ', @Logic)+1,50))+ 'D-P'+(substring (@Logic,CHARINDEX(' ', @Logic)+1,50))+'D'

END

ELSE

IF @Logic LIKE '%-%'

BEGIN

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 0 AND 60

Set @MatStartEnd = 'P'+ISNULL(CAST(DATEDIFF(DAY,@AsOfDate,@MaturityDate) AS VARCHAR),'0')+'D-P'+ISNULL(CAST(DATEDIFF(DAY,@AsOfDate,@MaturityDate) AS VARCHAR),'0')+'D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 61 AND 67

Set @MatStartEnd = 'P61D-P67D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 68 AND 74

Set @MatStartEnd = 'P68D-P74D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 75 AND 82

Set @MatStartEnd = 'P75D-P82D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 83 AND 90

Set @MatStartEnd = 'P83D-P90D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 91 AND 120

Set @MatStartEnd = 'P91D-P120D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 121 AND 150

Set @MatStartEnd = 'P121D-P150D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 151 AND 180

Set @MatStartEnd = 'P151D-P180D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 181 AND 270

Set @MatStartEnd = 'P181D-P270D'

IF DATEDIFF(DAY,@AsOfDate,@MaturityDate) BETWEEN 271 AND 365

Set @MatStartEnd = 'P271D-P365D'

END

ELSE

SET @CALC = (SELECT LEFT(@Logic,3))

SET @CALC1 = (SELECT RIGHT(@CALC,1))

SET @CALC2 = (SELECT CAST(@CALC1 AS INT))

--Logic for calendar days 271 to >5 years

IF @Logic LIKE '> %' AND (@CALC2 < 5)

BEGIN

Set @MatStartEnd = 'P' + CAST(((365 \*@CALC2)+1)AS VARCHAR) + 'D-P' + CAST(((365 \* (@CALC2+1))) AS VARCHAR)+ 'D'

END

ELSE

IF @Logic LIKE '> %' AND (@CALC2 = 5)

BEGIN

Set @MatStartEnd = 'P'+ CAST(((365 \* @CALC2)+2) AS VARCHAR) + 'D-'

END

RETURN @MatStartEnd

END

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[fn\_ExtractRawData] Script Date: 07/11/2017 22:35:59 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE FUNCTION [dbo].[fn\_ExtractRawData](@PID AS varchar (20), @Portfolio AS varchar (100), @AsOfDate AS Date)

RETURNS @trackingItems TABLE

(

[Agg\_2052ID] [int] NULL,

[AggBOCID] [int] NULL,

[Tgt\_ID] [int] NOT NULL,

[SrcID\_Description] [varchar](255) NULL,

[SrcID] [varchar](max) NULL,

[reportScope] [nvarchar](255) NULL,

[Currency] [nvarchar](255) NULL,

[Converted] [nvarchar](255) NULL,

[PID] [varchar](255) NULL,

[product] [nvarchar](255) NULL,

[SID] [varchar](255) NULL,

[SID2] [varchar](255) NULL,

[subProduct] [varchar](255) NULL,

[subProduct2] [varchar](255) NULL,

[CID] [decimal](33, 15) NULL,

[counterPartySector] [nvarchar](255) NULL,

[maturityValue] [decimal](30, 15) NULL,

[marketValue] [decimal](30, 15) NULL,

[lendableValue] [decimal](30, 15) NULL,

[maturityBucket] [nvarchar](255) NULL,

[collateralClass] [nvarchar](255) NULL,

[collateralValue] [decimal](30, 15) NULL,

[CollateralCurrency] [nvarchar](255) NULL,

[insured] [nvarchar](255) NULL,

[trigger] [nvarchar](255) NULL,

[Rehypothecated] [nvarchar](255) NULL,

[forwardStartValue] [decimal](30, 15) NULL,

[forwardStartBucket] [nvarchar](255) NULL,

[internal] [nvarchar](255) NULL,

[internalCounterParty] [nvarchar](255) NULL,

[primeBrokerage] [nvarchar](255) NULL,

[treasuryControl] [nvarchar](255) NULL,

[unencumbered] [nvarchar](255) NULL,

[effectiveMaturityStart] [nvarchar](255) NULL,

[effectiveMaturityEnd] [nvarchar](255) NULL,

[effectiveMaturityBucket] [nvarchar](255) NULL,

[customerNumber] [nvarchar](255) NULL,

[Portfolio] [varchar](255) NULL,

[DESC1] [varchar](255) NULL,

[CompanyName] [varchar](255) NULL,

[reportType] [nvarchar](255) NULL,

[maturityDate] [date] NULL,

[AsofDate] [date] NULL,

[event] [varchar](255) NULL,

[settlementMechanism] varchar (255) NULL,

[reasonableness\_Customer] [varchar](255) NULL,

[reasonableness\_Sector] [varchar](255) NULL,

[reasonableness\_Company] [varchar](255) NULL,

[reasonableness\_Contract\_DealNo] [varchar](255) NULL

)

AS

BEGIN

INSERT INTO @trackingItems

(

[Agg\_2052ID],

[AggBOCID],

[Tgt\_ID],

[SrcID\_Description],

[SrcID],

[reportScope],

[Currency],

[Converted],

[PID],

[product],

[SID],

[SID2],

[subProduct],

[subProduct2],

[CID],

[counterPartySector],

[maturityValue],

[marketValue],

[lendableValue],

[maturityBucket],

[collateralClass],

[collateralValue],

[CollateralCurrency],

[insured],

[trigger],

[Rehypothecated],

[forwardStartValue],

[forwardStartBucket],

[internal],

[internalCounterParty],

[primeBrokerage],

[treasuryControl],

[unencumbered],

[effectiveMaturityStart],

[effectiveMaturityEnd],

[effectiveMaturityBucket],

customerNumber,

[Portfolio],

[DESC1],

[CompanyName],

[reportType],

[maturityDate],

[AsofDate],

[event],

[settlementMechanism],

[reasonableness\_Customer],

[reasonableness\_Sector],

[reasonableness\_Company],

[reasonableness\_Contract\_DealNo]

)

SELECT RawData.\*

FROM

(

SELECT FR.Agg\_2052ID, Agg.AggBOCID,Tgt\_DO.Tgt\_depositsOutflowID,Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

, Tgt\_DO.SrcID

, Tgt\_DO.reportScope

, Tgt\_DO.Currency

, Tgt\_DO.Converted

, Tgt\_DO.PID

, Tgt\_DO.product

,NULL AS [SID]

,NULL AS [SID2]

,NULL AS subProduct

,NULL AS subProduct2

, Tgt\_DO.CID

, Tgt\_DO.counterPartySector

, Tgt\_DO.maturityValue

, NULL AS marketValue

, NULL AS lendableValue

, Tgt\_DO.maturityBucket

, Tgt\_DO.collateralClass

, Tgt\_DO.collateralValue

, Tgt\_DO.CollateralCurrency

, Tgt\_DO.insured

, Tgt\_DO.[trigger]

, Tgt\_DO.Rehypothecated

, NULL AS forwardStartValue

, NULL AS forwardStartBucket

, Tgt\_DO.internal

, Tgt\_DO.internalCounterParty

, NULL AS primeBrokerage

, NULL AS treasuryControl

, NULL AS [unencumbered]

, NULL AS [effectiveMaturityStart]

, NULL AS [effectiveMaturityEnd]

, NULL AS [effectiveMaturityBucket]

,customerNumber

, Tgt\_DO.Portfolio

, Tgt\_DO.DESC1

, Tgt\_DO.CompanyName

, Tgt\_DO.reportType

, tgt\_DO.[maturityDate]

, Tgt\_DO.AsofDate

, Tgt\_DO.[event]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_depositsOutflow Tgt\_DO

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_DO.PID = Agg.ProductID

AND Tgt\_DO.Tgt\_depositsOutflowID = Agg.TgtID

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_DO.PID = FR.ProductID

AND Tgt\_DO.Tgt\_depositsOutflowID = FR.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_DO.PID = SRC.Product

AND Tgt\_DO.Tgt\_depositsOutflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

WHERE ((Tgt\_DO.PID = @PID AND Tgt\_DO.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND Tgt\_DO.maturityValue <> 0 AND Tgt\_DO.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID,Tgt\_OO.Tgt\_otherOutflowID,Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

, Tgt\_OO.SrcID

, Tgt\_OO.reportScope

, Tgt\_OO.Currency

, Tgt\_OO.Converted

, Tgt\_OO.PID

, Tgt\_OO.product

,NULL AS [SID]

,NULL AS [SID2]

,NULL AS subProduct

,NULL AS subProduct2

, Tgt\_OO.CID

, Tgt\_OO.counterPartySector

, Tgt\_OO.maturityValue

, NULL AS marketValue

, NULL AS lendableValue

, Tgt\_OO.maturityBucket

, Tgt\_OO.collateralClass

, Tgt\_OO.collateralValue

, Tgt\_OO.CollateralCurrency

, NULL AS insured

, NULL AS [trigger]

, NULL AS Rehypothecated

, Tgt\_OO.forwardStartValue

, TGT\_OO.forwardStartBucket

, Tgt\_OO.internal

, Tgt\_OO.internalCounterParty

, Tgt\_OO.primeBrokerage

,NULL AS treasuryControl

, NULL AS [unencumbered]

, NULL AS [effectiveMaturityStart]

, NULL AS [effectiveMaturityEnd]

, NULL AS [effectiveMaturityBucket]

,customerNumber

, Tgt\_OO.Portfolio

, Tgt\_OO.DESC1

, Tgt\_OO.CompanyName

, Tgt\_OO.reportType

, Tgt\_OO.[maturityDate]

, Tgt\_OO.AsofDate

, Tgt\_OO.[event]

, NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_otherOutflow Tgt\_OO

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_OO.PID = Agg.ProductID

AND Tgt\_OO.Tgt\_otherOutflowID = Agg.TgtID

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_OO.PID = FR.ProductID

AND Tgt\_OO.Tgt\_otherOutflowID = FR.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_OO.PID = SRC.Product

AND Tgt\_OO.Tgt\_otherOutflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

WHERE ((Tgt\_OO.PID = @PID AND Tgt\_OO.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND Tgt\_OO.maturityValue <> 0 AND Tgt\_OO.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID,Tgt\_SO.Tgt\_securedOutflowID ,TablePrimaryKeyName AS SrcID\_Description

,Tgt\_SO.SrcID

, Tgt\_SO.reportScope

, Tgt\_SO.Currency

, Tgt\_SO.Converted

, Tgt\_SO.PID

, Tgt\_SO.product

, Tgt\_SO.[SID]

, NULL AS [SID2]

, Tgt\_SO.subProduct

, NULL AS subProduct2

, Tgt\_SO.CID

, Tgt\_SO.counterPartySector

, Tgt\_SO.maturityValue

, NULL AS marketValue

, NULL AS lendableValue

, Tgt\_SO.maturityBucket

, Tgt\_SO.collateralClass

, Tgt\_SO.collateralValue

, Tgt\_SO.CollateralCurrency

, NULL AS insured

, NULL AS [trigger]

, Tgt\_SO.rehypothecated

, Tgt\_SO.forwardStartValue

, Tgt\_so.forwardStartBucket

, Tgt\_SO.internal

, Tgt\_SO.internalCounterParty

, Tgt\_So.primeBrokerage

, Tgt\_So.treasuryControl AS treasuryControl

, NULL AS [unencumbered]

, NULL AS [effectiveMaturityStart]

, NULL AS [effectiveMaturityEnd]

, NULL AS [effectiveMaturityBucket]

,customerNumber

, Tgt\_SO.Portfolio

, Tgt\_SO.DESC1

, Tgt\_SO.CompanyName

, Tgt\_SO.reportType

, Tgt\_SO.[maturityDate]

, Tgt\_SO.AsofDate

, Tgt\_so.[event]

, Tgt\_SO.settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_securedOutflow Tgt\_SO

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_SO.PID = Agg.ProductID

AND Tgt\_SO.Tgt\_securedOutflowID = Agg.TgtID

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_SO.PID = FR.ProductID

AND Tgt\_SO.Tgt\_securedOutflowID = FR.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_SO.PID = SRC.Product

AND Tgt\_SO.Tgt\_securedOutflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_NB REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

WHERE ((Tgt\_so.PID = @PID AND Tgt\_so.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND Tgt\_SO.maturityValue <> 0 AND Tgt\_so.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_wholesaleOutflowID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,Tgt\_WO.PID

,Tgt\_WO.product

,NULL AS [SID]

,NULL AS [SID2]

,NULL AS subProduct

,NULL AS subProduct2

,CID

,counterPartySector

,maturityValue

,NULL AS marketValue

,NULL AS lendableValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,NULL AS TreasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_WO.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_WO.[maturityDate]

,Tgt\_WO.AsofDate

,[EVENT]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_wholesaleOutflow Tgt\_WO

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_WO.PID = Agg.ProductID

AND Tgt\_WO.Tgt\_wholesaleoutflowID = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_WO.PID = SRC.Product

AND Tgt\_WO.Tgt\_wholesaleoutflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_WO.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_WO.PID = FR.ProductID

AND Tgt\_WO.Tgt\_wholesaleoutflowID = FR.TgtID

WHERE ((Tgt\_WO.PID = @PID AND Tgt\_WO.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND maturityValue <> 0 AND Tgt\_WO.AsofDate = @AsOfDate

-----------------------

-- UNION ALL INFLOWS

-----------------------

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_AI.Tgt\_assetInflowID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,Tgt\_AI.PID

,Tgt\_AI.product

,Tgt\_AI.[SID]

,NULL AS [SID2]

,Tgt\_AI.subProduct

,NULL AS subProduct2

,NULL AS CID

,NULL AS counterPartySector

,NULL AS maturityValue

,Tgt\_AI.marketValue

,Tgt\_AI.lendableValue

,maturityBucket

,collateralClass

,NULL AS collateralValue

,NULL AS CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,forwardStartValue

,forwardStartBucket

,NULL AS internal

,NULL AS internalCounterParty

,NULL AS primeBrokerage

,Tgt\_AI.treasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_AI.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_AI.[maturityDate]

,Tgt\_AI.AsofDate

,[EVENT]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_assetInflow AS Tgt\_AI

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_AI.PID = Agg.ProductID

AND Tgt\_AI.Tgt\_assetInflowID = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_AI.PID = SRC.Product

AND Tgt\_AI.Tgt\_assetInflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_AI.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_AI.PID = FR.ProductID

AND Tgt\_AI.Tgt\_assetInflowID = FR.TgtID

WHERE ((Tgt\_AI.PID = @PID AND Tgt\_AI.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND Tgt\_AI.marketValue <> 0 AND Tgt\_AI.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_UI.TGT\_unsecuredInflowID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,Tgt\_UI.PID

,Tgt\_UI.product

,Tgt\_UI.[SID]

,NULL AS [SID2]

,NULL AS subProduct

,NULL AS subProduct2

,CID

,counterPartySector

,maturityValue

,NULL AS marketValue

,NULL AS lendableValue

,maturityBucket

,NULL AS collateralClass

,NULL AS collateralValue

,NULL AS CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,NULL AS treasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_UI.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_UI.[maturityDate]

,Tgt\_UI.AsofDate

,[EVENT]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_unsecuredInflow AS Tgt\_UI

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_UI.PID = Agg.ProductID

AND Tgt\_UI.TGT\_unsecuredInflowID = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_UI.PID = SRC.Product

AND Tgt\_UI.TGT\_unsecuredInflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_UI.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_UI.PID = FR.ProductID

AND Tgt\_UI.TGT\_unsecuredInflowID = FR.TgtID

WHERE ((Tgt\_UI.PID = @PID AND Tgt\_UI.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND maturityValue <> 0 AND Tgt\_UI.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, TGT\_SI.Tgt\_securedInflowID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,TGT\_SI.PID

,TGT\_SI.product

,[SID]

,NULL AS [SID2]

,Tgt\_SI.subProduct

,NULL AS subProduct2

,CID

,counterPartySector

,maturityValue

,NULL AS marketValue

,NULL AS lendableValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,treasuryControl

, [unencumbered]

, [effectiveMaturityStart]

, [effectiveMaturityEnd]

, [effectiveMaturityBucket]

,customerNumber

,TGT\_SI.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_SI.[maturityDate]

,TGT\_SI.AsofDate

,[EVENT]

,settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_securedInflow AS Tgt\_SI

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON TGT\_SI.PID = Agg.ProductID

AND TGT\_SI.Tgt\_securedInflowID = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON TGT\_SI.PID = SRC.Product

AND TGT\_SI.Tgt\_securedInflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = TGT\_SI.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_SI.PID = FR.ProductID

AND Tgt\_SI.Tgt\_securedInflowID = FR.TgtID

WHERE ((Tgt\_SI.PID = @PID AND Tgt\_SI.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND maturityValue <> 0 AND Tgt\_SI.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_OI.Tgt\_otherInflowID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,Tgt\_OI.PID

,Tgt\_OI.product

,NULL AS [SID]

,NULL AS [SID2]

,NULL AS subProduct

,NULL AS subProduct2

,NULL AS CID

,NULL AS counterPartySector

,maturityValue

,NULL AS marketValue

,NULL AS lendableValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,treasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_OI.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_OI.[maturityDate]

,Tgt\_OI.AsofDate

,[EVENT]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_otherInflow AS Tgt\_OI

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_OI.PID = Agg.ProductID

AND Tgt\_OI.Tgt\_otherInflowID = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_OI.PID = SRC.Product

AND Tgt\_OI.Tgt\_otherInflowID = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_OI.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_OI.PID = FR.ProductID

AND Tgt\_OI.Tgt\_otherInflowID= FR.TgtID

WHERE ((Tgt\_OI.PID = @PID AND Tgt\_OI.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND maturityValue <> 0 AND Tgt\_OI.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_IS.Tgt\_informationalSupplementalID, Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,Currency

,Converted

,Tgt\_IS.PID

,Tgt\_IS.product

,[SID]

,[SID2]

,subProduct

,subProduct2

,NULL AS CID

,NULL AS counterPartySector

,NULL AS maturityValue

,marketValue

,NULL AS lendableValue

,NULL AS maturityBucket

,collateralClass

,NULL AS collateralValue

,NULL AS CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,NULL AS forwardStartValue

,NULL AS forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,NULL AS treasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_IS.Portfolio

,DESC1

,CompanyName

,reportType

, Tgt\_IS.[maturityDate]

,Tgt\_IS.AsofDate

,[EVENT]

,NULL AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_informationalSupplemental AS Tgt\_IS

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_IS.PID = Agg.ProductID

AND Tgt\_IS.[Tgt\_informationalSupplementalID] = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_IS.PID = SRC.Product

AND Tgt\_IS.[Tgt\_informationalSupplementalID] = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_IS.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_IS.PID = FR.ProductID

AND Tgt\_IS.[Tgt\_informationalSupplementalID]= FR.TgtID

WHERE ((Tgt\_IS.PID = @PID AND Tgt\_IS.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND marketValue <> 0 AND Tgt\_IS.AsofDate = @AsOfDate

UNION

SELECT FR.Agg\_2052ID, Agg.AggBOCID, Tgt\_FX.[Tgt\_ForeignExchange\_ID], Ref\_tbl.TablePrimaryKeyName AS SrcID\_Description

,SrcID

,reportScope

,[Currency\_1] AS Currency

,[Currency\_1\_Converted] AS Converted

,Tgt\_FX.PID

,Tgt\_FX.product

,NULL [SID]

,NULL [SID2]

,NULL subProduct

,NULL subProduct2

,NULL AS CID

,NULL AS counterPartySector

,[MaturityAmountCurrency1] AS maturityValue

,NULL marketValue

,NULL AS lendableValue

,maturityBucket

,NULL collateralClass

,NULL AS collateralValue

,NULL AS CollateralCurrency

,NULL AS INSURED

,NULL AS [TRIGGER]

,NULL AS rehypothecated

,NULL AS forwardStartValue

,NULL AS forwardStartBucket

,internal

,internalCounterParty

,NULL primeBrokerage

,NULL AS treasuryControl

,NULL AS [unencumbered]

,NULL AS [effectiveMaturityStart]

,NULL AS [effectiveMaturityEnd]

,NULL AS [effectiveMaturityBucket]

,customerNumber

,Tgt\_FX.Portfolio

,NULL DESC1

,CompanyName

,reportType

, Tgt\_FX.[maturityDate]

,Tgt\_FX.AsofDate

,[EVENT]

,[Settlement] AS settlementMechanism

, [reasonableness\_Customer]

, [reasonableness\_Sector]

, [reasonableness\_Company]

, [reasonableness\_Contract\_DealNo]

FROM app.Tgt\_ForeignExchange AS Tgt\_FX

LEFT JOIN app.Map\_Tgt\_Agg\_BOC Agg

ON Tgt\_FX.PID = Agg.ProductID

AND Tgt\_FX.[Tgt\_ForeignExchange\_ID] = Agg.TgtID

LEFT JOIN app.Map\_Src\_Tgt SRC

ON Tgt\_FX.PID = SRC.Product

AND Tgt\_FX.[Tgt\_ForeignExchange\_ID] = SRC.TargetID

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_nb REF\_tbl

ON REF\_tbl.PID = SRC.Product

AND REF\_tbl.SourceTableName = SRC.SourceTableName

AND REF\_TBL.Portfolio = Tgt\_FX.Portfolio

LEFT JOIN app.Map\_Tgt\_Agg FR

ON Tgt\_FX.PID = FR.ProductID

AND Tgt\_FX.[Tgt\_ForeignExchange\_ID]= FR.TgtID

WHERE ((Tgt\_FX.PID = @PID AND Tgt\_FX.Portfolio = @Portfolio

AND Agg.AggregationPortfolioName = @Portfolio AND FR.AggregationPortfolioName = @Portfolio AND SRC.TargetPortfolioName = @Portfolio)

OR @Portfolio = 'All')

AND [MaturityAmountCurrency1] <> 0 AND Tgt\_FX.AsofDate = @AsOfDate

) AS RawData

WHERE ((RawData.PID = @PID AND RawData.Portfolio = @Portfolio) OR @Portfolio = 'All')

AND RawData.AsofDate = @AsOfDate

RETURN;

END;

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[fn\_CurrencyExchange] Script Date: 07/11/2017 22:35:59 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE FUNCTION [dbo].[fn\_CurrencyExchange]

(@Currency Varchar(20), @MonetaryValue Decimal (30,15), @AsOfDate Date)

RETURNS Decimal (30,15)

AS

BEGIN

DECLARE @DollarValue DECIMAL (30,15)

DECLARE @intermediateValue DECIMAL(30,15)

-- If currency parameter is NULL, return the same value

IF @CURRENCY IS NULL OR @Currency = ''

SET @DollarValue = @MonetaryValue

-- If currency parameter is not found in reference table, return the same value

ELSE IF @CURRENCY NOT IN (SELECT DISTINCT CURRENCY\_CODE

FROM SRC.FBNK\_CURRENCY cur

WHERE ASOFDATE = @ASOFDATE)

SET @DollarValue = @MonetaryValue

--Perform calculations

ELSE

IF (SELECT DISTINCT Quotation\_code

from SRC.FBNK\_CURRENCY cur

WHERE cur.Currency\_code = @CURRENCY AND CAST(cur.AsOfDate AS DATE) = @AsOfDate) = 0

SET @DollarValue = @MonetaryValue \* ( SELECT DISTINCT [MID\_REVAL\_RATE]

FROM SRC.FBNK\_CURRENCY cur

WHERE Cur.Currency\_code = @CURRENCY

AND CAST(cur.AsOfDate AS DATE) = @AsOfDate)

ELSE IF(SELECT DISTINCT Quotation\_code

from SRC.FBNK\_CURRENCY cur

WHERE cur.Currency\_code = @CURRENCY AND CAST(cur.AsOfDate AS DATE) = @AsOfDate) IS NULL

SET @DollarValue = ISNULL(@MonetaryValue / NULLIF((SELECT DISTINCT [MID\_REVAL\_RATE]

FROM SRC.FBNK\_CURRENCY cur

--FROM [9amhsadcsql106\SQL106$dvlp].[DATA\_MART\_US].dbo.FBNK\_CURRENCY cur

WHERE Cur.Currency\_code = @CURRENCY

AND CAST(cur.AsOfDate AS DATE) = @AsOfDate),'0'),0)

RETURN @DollarValue

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_RPT\_FR2052a] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_RPT\_FR2052a]

--@AsOfDate Date

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_secured\_Inflow]

\*\* Report Type: Secured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

--BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @monetaryValueThreshold DECIMAL (18,2)

SET @monetaryValueThreshold = 5000.00

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

--------------------------------------------------------------

-- Check all values that are less then 5000 / Per record

--

--------------------------------------------------------------

INSERT INTO [REPORT\_FR2052A].APP.[IssueTracer]

([IssueNumber]

,[IssueState]

,[IssueSeverity]

,[IssueLine]

,[IssueProc]

,[IssueMsg]

,[UserName]

,[HostName]

,[IssueDate]

,[ExecutionStatus])

SELECT

0 AS issueNumber

, 0 AS issueState

, 0 AS issueSeverity

, 0 AS issueLine

,CASE WHEN ReportType = 'Inflows-Assets' THEN 'sp\_RPT\_FR2052a - I.A.'+ PID

WHEN ReportType = 'Inflows-Unsecured' THEN 'sp\_RPT\_FR2052a - I.U.' + PID

WHEN ReportType = 'Inflows-Secured' THEN 'sp\_RPT\_FR2052a - I.S.' + PID

WHEN ReportType = 'Inflows-Other' THEN 'sp\_RPT\_FR2052a - I.O.' + PID

WHEN ReportType = 'Outflows-Secured' THEN 'sp\_RPT\_FR2052a - O.S.' + PID

WHEN ReportType = 'Outflows-Wholesale' THEN 'sp\_RPT\_FR2052a - O.W.' + PID

WHEN ReportType = 'Outflows-Deposits' THEN 'sp\_RPT\_FR2052a - O.D.' + PID

WHEN ReportType = 'Outflows-Other' THEN 'sp\_RPT\_FR2052a - O.O.' + PID

WHEN ReportType = 'Supplemental-Informational' THEN 'sp\_RPT\_FR2052a - S.I.' + PID

END AS issueProc

,CASE WHEN ISNULL(MarketValue, 0) < @monetaryValueThreshold AND ISNULL(MaturityValue, 0) < @monetaryValueThreshold AND ISNULL(LendableValue, 0) < @monetaryValueThreshold

AND ISNULL(CollateralValue, 0) < @monetaryValueThreshold AND ISNULL(ForwardStartValue, 0) < @monetaryValueThreshold

THEN 'Removed Monetary values less than 5000 :

\*\*Agg\_FR2052aID = ' +ISNULL(CAST(Agg\_FR2052aID AS VARCHAR(100)),'') + ' and MarketValue = ' + ISNULL(CAST(MarketValue AS VARCHAR(100)), '') + ' and MaturityValue = ' + ISNULL(CAST(MaturityValue AS VARCHAR(100)), '')

+ ', LendableValue = ' + ISNULL(CAST(LendableValue AS VARCHAR(100)), '')

+ ', CollateralValue = ' + ISNULL(CAST(CollateralValue AS VARCHAR(100)), '')

+ ', ForwardStartValue = ' + ISNULL(CAST(ForwardStartValue AS VARCHAR(100)), '')

WHEN ISNULL(MarketValue, 0) < 0 OR ISNULL(MaturityValue, 0) < 0 OR ISNULL(LendableValue, 0) < 0 OR ISNULL(CollateralValue, 0) < 0 OR ISNULL(ForwardStartValue, 0) < 0

THEN 'Removed PID containg negative Monetary values:

\*\*Agg\_FR2052aID = ' +ISNULL(CAST(Agg\_FR2052aID AS VARCHAR(100)),'') + ' and MarketValue = ' + ISNULL(CAST(MarketValue AS VARCHAR(100)), '') + ' and MaturityValue = ' + ISNULL(CAST(MaturityValue AS VARCHAR(100)), '')

+ ' and LendableValue = ' + ISNULL(CAST(LendableValue AS VARCHAR(100)), '')

+ ' and CollateralValue = ' + ISNULL(CAST(CollateralValue AS VARCHAR(100)), '')

+ ' and ForwardStartValue = ' + ISNULL(CAST(ForwardStartValue AS VARCHAR(100)), '')

ELSE ''

END AS IssueMsg

,SUSER\_SNAME()

,Host\_NAME()

,@CreationDate

, 'Warning' AS ExecutionStatus

FROM app.AGG\_FR2052a

WHERE (ISNULL(MarketValue, 0) < @monetaryValueThreshold AND ISNULL(MaturityValue, 0) < @monetaryValueThreshold AND ISNULL(LendableValue, 0) < @monetaryValueThreshold

AND ISNULL(CollateralValue, 0) < @monetaryValueThreshold AND ISNULL(ForwardStartValue, 0) < @monetaryValueThreshold)

OR

Product IN

(SELECT DISTINCT Product

FROM App.AGG\_FR2052a A

WHERE ISNULL(A.MarketValue, 0) < 0 OR ISNULL(A.MaturityValue, 0) < 0 OR ISNULL(A.LendableValue, 0) < 0

OR ISNULL(A.CollateralValue, 0) < 0 OR ISNULL(A.ForwardStartValue, 0) < 0)

/\*======================================================================

TEMP TABLE - FILTER OUT ALL VALUES LESS THAN 5000 AND Negative Values

======================================================================\*/

IF OBJECT\_ID ('TEMPDB..#TEMP\_AGG\_FR2052a') IS NOT NULL DROP TABLE #TEMP\_AGG\_FR2052a

SELECT \*

INTO #TEMP\_AGG\_FR2052a

FROM App.AGG\_FR2052a

WHERE NOT (ISNULL(MarketValue, 0) < @monetaryValueThreshold AND ISNULL(MaturityValue, 0) < @monetaryValueThreshold AND ISNULL(LendableValue, 0) < @monetaryValueThreshold

AND ISNULL(CollateralValue, 0) < @monetaryValueThreshold AND ISNULL(ForwardStartValue, 0) < @monetaryValueThreshold)

AND Product NOT IN

(SELECT DISTINCT Product

FROM App.AGG\_FR2052a A

WHERE ISNULL(A.MarketValue, 0) < 0 OR ISNULL(A.MaturityValue, 0) < 0 OR ISNULL(A.LendableValue, 0) < 0

OR ISNULL(A.CollateralValue, 0) < 0 OR ISNULL(A.ForwardStartValue, 0) < 0)

/\*======================================================================

All Inflows

======================================================================\*/

-- Inflow Assets

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,assetProduct

,assetSource

,reportScope

,assetInflowReport

,maturityBucket

,maturityStart

,maturityEnd

,forwardStartBucket

,forwardStartValue

,FmaturityEnd

,FmaturityStart

,lendableValue

,marketValue

,treasuryControl

,collateralClass

,Currency

,Converted

,asofdate

,creationDate

)

SELECT

AGG\_FR2052aID

,Product

,SubProduct

,reportscope

,ReportType

,maturityBucket

,maturityStart

,maturityEnd

,forwardStartBucket

,forwardStartValue

,forwardStart

,forwardEnd

,lendableValue

,marketValue

,treasuryControl

,collateralClass

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a

WHERE ReportType = 'Inflows-Assets'

--Inflows-Unsecured

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,unsecuredInflowReport

,unsecuredProduct

,reportScope

,maturityBucket

,maturityStart

,maturityEnd

,forwardStartBucket

,forwardStartValue

,FmaturityEnd

,FmaturityStart

,maturityValue

,treasuryControl

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,counterPartySector

,Currency

,Converted

,asofdate

,creationDate)

SELECT

AGG\_FR2052aID

,ReportType

,Product

,reportscope

,maturityBucket

,maturityStart

,maturityEnd

,forwardStartBucket

,forwardStartValue

,forwardStart

,forwardEnd

,maturityValue

,treasuryControl

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,counterPartySector

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a

WHERE ReportType = 'Inflows-Unsecured'

--Inflows-secured Select and insert statement is develop according to the XML report

INSERT INTO app.RPT\_FR2052a

(

AGG\_ID

,treasuryControl

,internal

,primeBrokerage

,unencumbered

,reportScope

,securedInflowProduct

,pledgeLevel

,counterPartySector

,maturityValue

,maturityStart

,maturityEnd

,EmaturityStart

,EmaturityEnd

,forwardStartValue

,FmaturityStart

,FmaturityEnd

,collateralClass

,collateralValue

,settlementMechanism

,internalCounterParty

,securedInflowReport

,Currency

,Converted

,asofdate

,creationDate )

SELECT

Agg.AGG\_FR2052aID

,agg.treasuryControl

,agg.internal

,agg.primeBrokerage

,agg.unencumbered

,agg.reportScope

,agg.product -- securedInflowProduct

,agg.SubProduct -- Pledgelevel

,agg.counterPartySector

,agg.maturityValue

,agg.maturityStart

,agg.maturityEnd

,agg.EffectiveMaturityStart

,agg.EffectiveMaturityEnd

,agg.forwardStartValue

,agg.ForwardStart

,agg.ForwardEnd

,agg.collateralClass

,agg.collateralValue

,agg.settlementMechanism

,agg.internalCounterParty

,agg.reportType

,agg.Currency

,agg.Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.ReportType = 'Inflows-secured'

--Inflows-Other Select and insert statement is develop according to the XML report

INSERT INTO app.RPT\_FR2052a

(

AGG\_ID

,internal

,primeBrokerage

,treasuryControl

,reportScope

,otherInflowProduct

,maturityValue

,maturityStart

,maturityEnd

,forwardStartValue

,FmaturityStart

,FmaturityEnd

,collateralClass

,collateralValue

,internalCounterParty

,otherInflowReport

,Currency

,Converted

,asofdate

,creationDate )

SELECT

Agg.AGG\_FR2052aID

,agg.internal

,agg.primeBrokerage

,Agg.TreasuryControl

,agg.reportScope

,product --OtherinflowProduct

,maturityValue

,maturityStart

,maturityEnd

,forwardStartValue

,ForwardStart

,ForwardEnd

,collateralClass

,collateralValue

,internalCounterParty

,reportType

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.ReportType = 'Inflows-Other'

/\*======================================================================

All Outflows

======================================================================\*/

-- Outflows Deposits

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,insured

,[trigger]

,internal

,rehypothecated

,reportScope

,depositsProduct

,counterPartySector

,maturityValue

,maturityStart

,maturityEnd

,collateralClass

,collateralValue

,internalCounterParty

,depositsOutflowReport

,Currency

,Converted

,asofdate

,creationDate)

SELECT

Agg.AGG\_FR2052aID

,agg.insured

,agg.[trigger]

,agg.internal

,agg.rehypothecated

,agg.reportScope

,agg.product -- DepositProduct in RPT\_2052a table

,agg.counterPartySector

,agg.maturityValue

,agg.maturityStart

,agg.maturityEnd

,agg.collateralClass

,agg.collateralValue

,agg.internalCounterParty

,agg.reportType

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.reportType = 'Outflows-Deposits'

-- Outflows wholesale

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,internal

,primeBrokerage

,reportScope

,wholesaleOutflowProduct

,counterPartySector

,maturityValue

,maturityStart

,maturityEnd

,collateralClass

,collateralValue

,forwardStartValue

,FmaturityStart

,FmaturityEnd

,internalCounterParty

,wholesaleOutflowReport

,Currency

,Converted

,asofdate

,creationDate)

SELECT

Agg.AGG\_FR2052aID

,agg.internal

,agg.primeBrokerage

,agg.reportScope

,agg.product -- Wholesale Produce

,agg.counterPartySector

,agg.maturityValue

,agg.maturityStart

,agg.maturityEnd

,agg.collateralClass

,agg.collateralValue

,agg.forwardStartValue

,agg.ForwardStart

,agg.ForwardEnd

,agg.internalCounterParty

,agg.reportType

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.reportType = 'Outflows-Wholesale'

-- Outflows Secured

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,treasuryControl

,rehypothecated

,internal

,primeBrokerage

,reportScope

,securedOutflowProduct

,securedOutflowCounterPartyOrCoverage

,counterPartySector

,maturityValue

,maturityStart

,maturityEnd

,forwardStartValue

,FmaturityStart

,FmaturityEnd

,collateralClass

,collateralValue

,settlementMechanism

,internalCounterParty

,securedOutflowReport

,Currency

,Converted

--,publicCounterParty --- MUST ADD

,asofdate

,creationDate

)

SELECT

Agg.AGG\_FR2052aID

,agg.treasuryControl

,agg.rehypothecated

,agg.internal

,agg.primeBrokerage

,agg.reportScope

,agg.product

,agg.SubProduct

,agg.counterPartySector

,agg.maturityValue

,agg.maturityStart

,agg.maturityEnd

,agg.forwardStartValue

,agg.ForwardStart

,agg.ForwardEnd

,agg.collateralClass

,agg.collateralValue

,agg.settlementMechanism

,agg.internalCounterParty

--,agg.publicCounterParty --\*\*\*\* MUST ADD

,agg.reportType

,agg.Currency

,agg.Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.reportType = 'Outflows-Secured'

-- Outflows Other

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,internal

,primeBrokerage

,reportScope

,otherOutflowProduct

,counterPartySector

,maturityValue

,maturityStart

,maturityEnd

,forwardStartValue

,FmaturityStart

,FmaturityEnd

,collateralClass

,collateralValue

,internalCounterParty

,otherOutflowReport

,Currency

,Converted

,asofdate

,creationDate)

SELECT

Agg.AGG\_FR2052aID

,agg.internal

,agg.primeBrokerage

,agg.reportScope

,agg.product

,agg.counterPartySector

,agg.maturityValue

,agg.maturityStart

,agg.maturityEnd

,agg.forwardStartValue

,agg.ForwardStart

,agg.ForwardEnd

,agg.collateralClass

,agg.collateralValue

,agg.internalCounterParty

,agg.reportType

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.reportType = 'Outflows-other'

/\*======================================================================

INFORMATIONAL SUPPLEMENTAL

======================================================================\*/

INSERT INTO app.RPT\_FR2052a

(AGG\_ID

,internal

,primeBrokerage

,reportScope

,supplementalInformationProduct

,segregationCategory

,settlementMechanism

,marketValue

,collateralClass

,internalCounterparty

,informationalSupplementalReport

,Currency

,Converted

,asofdate

,creationDate)

SELECT

Agg.AGG\_FR2052aID

,agg.internal

,agg.primeBrokerage

,agg.reportScope

,agg.product

,agg.SubProduct

,agg.SubProduct2

,agg.marketValue

,agg.collateralClass

,agg.internalCounterParty

,agg.reportType

,Currency

,Converted

,@ASOFDATE

,@CreationDate

FROM #TEMP\_AGG\_FR2052a Agg

WHERE agg.reportType = 'Supplemental-Informational'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_RequiredTableCheck] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_RequiredTableCheck]

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

INSERT INTO [REPORT\_FR2052A].APP.[IssueTracer]

([IssueNumber]

,[IssueState]

,[IssueSeverity]

,[IssueLine]

,[IssueProc]

,[IssueMsg]

,[UserName]

,[HostName]

,[IssueDate]

,[ExecutionStatus])

SELECT '','','','',

proc\_name + ' | ' + table\_schema+'.'+table\_name

,'Keytables need to be updated'

,SUSER\_SNAME()

,Host\_NAME()

,GETDATE()

,'Fail'

FROM

(SELECT

proc\_name,

table\_name,

table\_schema

FROM (SELECT

o.name AS proc\_name,

oo.name AS table\_name,

inf.table\_schema as table\_schema,

INF.TABLE\_CATALOG,

row\_number() OVER(partition by o.name,oo.name ORDER BY o.name,oo.name ) AS row

FROM sysdepends d

INNER JOIN sysobjects o

ON o.id=d.id

INNER JOIN sysobjects oo

ON oo.id=d.depid

INNER JOIN INFORMATION\_SCHEMA.COLUMNS inf

ON TABLE\_NAME = oo.name

WHERE o.xtype = 'P'

aND TABLE\_CATALOG = 'REPORT\_FR2052A'

AND O.name NOT IN ('sp\_alterdiagram','sp\_creatediagram','sp\_dropdiagram','sp\_helpdiagramdefinitioN','sp\_helpdiagrams','sp\_renamediagram','sp\_upgraddiagrams')

and table\_name not like '%delete%'

and TABLE\_NAME is not null

)stored\_procedures

where row =1

) sys\_sp\_tbls

WHERE NOT EXISTS (SELECT \* FROM SRC.US\_FR2052A\_REF\_RequiredTables tbl

WHERE substring(tbl.[Table Name ],CHARINDEX('.', tbl.[Table Name ])+1,LEN(tbl.[Table Name ])) = sys\_sp\_tbls.table\_name

and tbl.[Schema name]= sys\_sp\_tbls.table\_schema

and table\_name not like '%delete%'

and table\_name is not null)

INSERT INTO [REPORT\_FR2052A].[APP].[IssueTracer]

([IssueNumber]

,[IssueState]

,[IssueSeverity]

,[IssueLine]

,[IssueProc]

,[IssueMsg]

,[UserName]

,[HostName]

,[IssueDate]

,[ExecutionStatus])

SELECT

'','','','',

TBL.PID + ' | ' +TBl.Portfolio

,tbl.[Table Name ] + ' Doesnt exist in the system'

,SUSER\_SNAME()

,Host\_NAME()

,GETDATE()

,'Fail'

FROM SRC.US\_FR2052A\_REF\_RequiredTables TBL

WHERE NOT EXISTS (SELECT \*

FROM INFORMATION\_SCHEMA.COLUMNS INF

WHERE TABLE\_NAME = substring(tbl.[Table Name ],CHARINDEX('.', tbl.[Table Name ])+1,LEN(tbl.[Table Name ]))

AND [TABLE\_SCHEMA] =tbl.[Schema name]

and TABLE\_NAME is not null

)

and tbl.[Table Name ] is not null

INSERT INTO [REPORT\_FR2052A].[APP].[IssueTracer]

([IssueNumber]

,[IssueState]

,[IssueSeverity]

,[IssueLine]

,[IssueProc]

,[IssueMsg]

,[UserName]

,[HostName]

,[IssueDate]

,[ExecutionStatus])

SELECT '','','','',

TBL.PID + ' | ' +TBl.Portfolio

,tbl.[Table Name ] + ' has 0 records'

,SUSER\_SNAME()

,Host\_NAME()

,GETDATE()

,'Fail'

FROM SRC.US\_FR2052A\_REF\_RequiredTables TBL

WHERE EXISTS (SELECT \* FROM

sys.tables ta

INNER JOIN sys.partitions pa

ON pa.OBJECT\_ID = ta.OBJECT\_ID

AND ta.name = tbl.[Table Name ]

INNER JOIN sys.schemas sc

ON ta.schema\_id = sc.schema\_id

AND sc.name = tbl.[Schema name]

WHERE ta.is\_ms\_shipped = 0 AND pa.index\_id IN (1,0)

and tbl.[Schema name] = 'src'

and tbl.[Table Name ] is not null

GROUP BY sc.name,ta.name

HAVING SUM(pa.rows) = 0

)

and tbl.[Schema name] = 'src'

and tbl.[Table Name ] is not null

SELECT \* FROM APP.IssueTracer

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END

GO

/\*\*\*\*\*\* Object: View [dbo].[vw\_Summary\_Ref\_Tables] Script Date: 07/11/2017 22:35:57 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[vw\_Summary\_Ref\_Tables]

AS

SELECT TOP (100) PERCENT Table\_Name, COUNT(\*) AS Rec\_Count, ReportingEntity, Asofdate

FROM (SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_Entity' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_Entity

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_InternalCounterParty' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_InternalCounterParty

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_CounterParty' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_CounterParty

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_COLLATERAL' AS Table\_Name, Reportingentity, Asofdate

FROM SRC.US\_FR2052A\_REF\_COLLATERAL

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_EntitySTB' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_EntitySTB

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_CounterPartySTB' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_CounterPartySTB

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_CDARS' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_CDARS

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_InternalCounterPartySTB' AS Table\_Name, ReportingEntity, Asofdate

FROM SRC.US\_FR2052A\_REF\_InternalCounterPartySTB

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].US\_FR2052A\_REF\_OperationalDepositTag' AS Table\_Name, ReportingEntity, ASOFDATE

FROM SRC.US\_FR2052A\_REF\_OperationalDepositTag) AS NB\_Ref

GROUP BY Table\_Name, ReportingEntity, Asofdate

ORDER BY ReportingEntity, Asofdate, Table\_Name

GO

EXEC sys.sp\_addextendedproperty @name=N'MS\_DiagramPane1', @value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]

Begin DesignProperties =

Begin PaneConfigurations =

Begin PaneConfiguration = 0

NumPanes = 4

Configuration = "(H (1[40] 4[20] 2[20] 3) )"

End

Begin PaneConfiguration = 1

NumPanes = 3

Configuration = "(H (1 [50] 4 [25] 3))"

End

Begin PaneConfiguration = 2

NumPanes = 3

Configuration = "(H (1 [50] 2 [25] 3))"

End

Begin PaneConfiguration = 3

NumPanes = 3

Configuration = "(H (4 [30] 2 [40] 3))"

End

Begin PaneConfiguration = 4

NumPanes = 2

Configuration = "(H (1 [56] 3))"

End

Begin PaneConfiguration = 5

NumPanes = 2

Configuration = "(H (2 [66] 3))"

End

Begin PaneConfiguration = 6

NumPanes = 2

Configuration = "(H (4 [50] 3))"

End

Begin PaneConfiguration = 7

NumPanes = 1

Configuration = "(V (3))"

End

Begin PaneConfiguration = 8

NumPanes = 3

Configuration = "(H (1[56] 4[18] 2) )"

End

Begin PaneConfiguration = 9

NumPanes = 2

Configuration = "(H (1 [75] 4))"

End

Begin PaneConfiguration = 10

NumPanes = 2

Configuration = "(H (1[66] 2) )"

End

Begin PaneConfiguration = 11

NumPanes = 2

Configuration = "(H (4 [60] 2))"

End

Begin PaneConfiguration = 12

NumPanes = 1

Configuration = "(H (1) )"

End

Begin PaneConfiguration = 13

NumPanes = 1

Configuration = "(V (4))"

End

Begin PaneConfiguration = 14

NumPanes = 1

Configuration = "(V (2))"

End

ActivePaneConfig = 0

End

Begin DiagramPane =

Begin Origin =

Top = 0

Left = 0

End

Begin Tables =

Begin Table = "NB\_Ref"

Begin Extent =

Top = 6

Left = 38

Bottom = 110

Right = 218

End

DisplayFlags = 280

TopColumn = 0

End

End

End

Begin SQLPane =

End

Begin DataPane =

Begin ParameterDefaults = ""

End

Begin ColumnWidths = 9

Width = 284

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

End

End

Begin CriteriaPane =

Begin ColumnWidths = 12

Column = 1440

Alias = 900

Table = 1170

Output = 720

Append = 1400

NewValue = 1170

SortType = 1350

SortOrder = 1410

GroupBy = 1350

Filter = 1350

Or = 1350

Or = 1350

Or = 1350

End

End

End

' , @level0type=N'SCHEMA',@level0name=N'dbo', @level1type=N'VIEW',@level1name=N'vw\_Summary\_Ref\_Tables'

GO

EXEC sys.sp\_addextendedproperty @name=N'MS\_DiagramPaneCount', @value=1 , @level0type=N'SCHEMA',@level0name=N'dbo', @level1type=N'VIEW',@level1name=N'vw\_Summary\_Ref\_Tables'

GO

/\*\*\*\*\*\* Object: View [dbo].[vw\_Summary\_NB\_ReportType] Script Date: 07/11/2017 22:35:57 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[vw\_Summary\_NB\_ReportType]

AS

SELECT TOP (100) PERCENT Table\_Name, COUNT(\*) AS Rec\_Count, PID\_ProductID, Reportingentity, Asofdate, SUM(CASE WHEN ISNULL(NB.PortfolioName, '1')

= '1' THEN 1 ELSE 0 END) AS PortName\_Blank\_Count

FROM (SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_IO]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_IO

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_IU]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_IU

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_OO]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_OO

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_OS]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_OS

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_OW]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_OW

UNION ALL

SELECT '[REPORT\_FR2052A].[SRC].[US\_FR2052A\_NB\_SI]' AS Table\_Name, ID, Reportingentity, PID\_ProductID, Product, PortfolioName, ReportType, Asofdate,

EntityName, CounterpartyCode, CounterpartyName

FROM SRC.US\_FR2052A\_NB\_SI) AS NB

GROUP BY Table\_Name, PID\_ProductID, Reportingentity, Asofdate

ORDER BY Reportingentity, Asofdate

GO

EXEC sys.sp\_addextendedproperty @name=N'MS\_DiagramPane1', @value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]

Begin DesignProperties =

Begin PaneConfigurations =

Begin PaneConfiguration = 0

NumPanes = 4

Configuration = "(H (1[40] 4[20] 2[20] 3) )"

End

Begin PaneConfiguration = 1

NumPanes = 3

Configuration = "(H (1 [50] 4 [25] 3))"

End

Begin PaneConfiguration = 2

NumPanes = 3

Configuration = "(H (1 [50] 2 [25] 3))"

End

Begin PaneConfiguration = 3

NumPanes = 3

Configuration = "(H (4 [30] 2 [40] 3))"

End

Begin PaneConfiguration = 4

NumPanes = 2

Configuration = "(H (1 [56] 3))"

End

Begin PaneConfiguration = 5

NumPanes = 2

Configuration = "(H (2 [66] 3))"

End

Begin PaneConfiguration = 6

NumPanes = 2

Configuration = "(H (4 [50] 3))"

End

Begin PaneConfiguration = 7

NumPanes = 1

Configuration = "(V (3))"

End

Begin PaneConfiguration = 8

NumPanes = 3

Configuration = "(H (1[56] 4[18] 2) )"

End

Begin PaneConfiguration = 9

NumPanes = 2

Configuration = "(H (1 [75] 4))"

End

Begin PaneConfiguration = 10

NumPanes = 2

Configuration = "(H (1[66] 2) )"

End

Begin PaneConfiguration = 11

NumPanes = 2

Configuration = "(H (4 [60] 2))"

End

Begin PaneConfiguration = 12

NumPanes = 1

Configuration = "(H (1) )"

End

Begin PaneConfiguration = 13

NumPanes = 1

Configuration = "(V (4))"

End

Begin PaneConfiguration = 14

NumPanes = 1

Configuration = "(V (2))"

End

ActivePaneConfig = 0

End

Begin DiagramPane =

Begin Origin =

Top = 0

Left = 0

End

Begin Tables =

Begin Table = "NB"

Begin Extent =

Top = 6

Left = 38

Bottom = 301

Right = 367

End

DisplayFlags = 280

TopColumn = 0

End

End

End

Begin SQLPane =

End

Begin DataPane =

Begin ParameterDefaults = ""

End

Begin ColumnWidths = 9

Width = 284

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

Width = 1500

End

End

Begin CriteriaPane =

Begin ColumnWidths = 12

Column = 1440

Alias = 900

Table = 1170

Output = 720

Append = 1400

NewValue = 1170

SortType = 1350

SortOrder = 1410

GroupBy = 1350

Filter = 1350

Or = 1350

Or = 1350

Or = 1350

End

End

End

' , @level0type=N'SCHEMA',@level0name=N'dbo', @level1type=N'VIEW',@level1name=N'vw\_Summary\_NB\_ReportType'

GO

EXEC sys.sp\_addextendedproperty @name=N'MS\_DiagramPaneCount', @value=1 , @level0type=N'SCHEMA',@level0name=N'dbo', @level1type=N'VIEW',@level1name=N'vw\_Summary\_NB\_ReportType'

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Proc\_InsertIssueDetails] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[Proc\_InsertIssueDetails]

(

--Declare variables for custom errors (i.e. not system errors)

@@ISSUENumber VARCHAR(MAX) = null,

@@ISSUEState VARCHAR(MAX) = null ,

@@ISSUESeverity VARCHAR(MAX) = null,

@@ISSUELine VARCHAR(MAX) = null,

@@ISSUEProc VARCHAR(MAX) = null,

@@ISSUEMesg VARCHAR(MAX) = null

)

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Insert system errors into error tracer table

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [Proc\_InsertErrorDetails]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

--Declare variables for system errors

DECLARE @@UserName VARCHAR(MAX)

DECLARE @@HostName VARCHAR(MAX)

DECLARE @@Status VARCHAR(20)

--If there are no custom errors, check for system errors

IF (

@@ISSUENumber IS NULL

AND @@ISSUEState IS NULL

AND @@ISSUESeverity IS NULL

AND @@ISSUELine IS NULL

AND @@ISSUEProc IS NULL

AND @@ISSUEMesg IS NULL

)

BEGIN

(

SELECT

@@ISSUENumber = ERROR\_NUMBER()

,@@ISSUEState = ERROR\_STATE()

,@@ISSUESeverity = ERROR\_SEVERITY()

,@@ISSUELine = ERROR\_LINE()

,@@ISSUEProc = ERROR\_PROCEDURE()

,@@ISSUEMesg = ERROR\_MESSAGE()

)

END

--Set the following variables for both custom and system erros

SELECT

@@UserName = SUSER\_SNAME()

,@@HostName = Host\_NAME()

--Insert all the values in the ErrorTracer Table

INSERT INTO app.ISSUETracer

(

ISSUENumber

,ISSUEState

,ISSUESeverity

,ISSUELine

,ISSUEProc

,ISSUEMsg

,UserName

,HostName

,ISSUEDate

,ExecutionStatus

)

VALUES

(

@@ISSUENumber

,@@ISSUEState

,@@ISSUESeverity

,@@ISSUELine

,@@ISSUEProc

,@@ISSUEMesg

,@@UserName

,@@HostName

,GETDATE()

,'FAIL'

)

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_eventHandler] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_eventHandler]

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_assetInflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_Asset\_Inflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_assetInflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_ForeignExchange tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_ForeignExchange | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_ForeignExchange TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_unsecuredInflow tgt1

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_Unsecured\_Inflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_unsecuredInflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_securedInflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_secured\_Inflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedInflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_otherInflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_other\_Inflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherInflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_wholesaleOutflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_wholesale\_outflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_wholesaleOutflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_securedOutflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_secured\_outflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedOutflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_otherOutflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_other\_outflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherOutflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_depositsOutflow tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_deposit\_outflow | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_depositsOutflow TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

IF EXISTS (SELECT DISTINCT [EVENT]

FROM app.Tgt\_informationalSupplemental tgt

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> '')

INSERT INTO app.ISSUETracer

SELECT '','','','','dbo.sp\_InformationalSupplemental | '+ tgt.PID,

tgt.[event],SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_informationalSupplemental TGT

WHERE [EVENT] IS NOT NULL

AND [EVENT] <> ''

-- for testing purposes: expect 24 records

-- Add nonclustered indexes to improve performance

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_XML\_GENERATOR] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_XML\_GENERATOR]

--@AsOfDate Date

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_secured\_Inflow]

\*\* Report Type: Secured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

--BEGIN TRY

--Declare to enter PID and Portfolio dynamically

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Purpose: SQL code to produce a sample XML file that is to be submitted to the FRB.

\*\* PID:

\*\* Portfolio:

\*\* Date:

\*\* Status:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- Declare variables to be used as attribute names

DECLARE @unitMultiplier INT

DECLARE @unit VARCHAR(50)

DECLARE @header VARCHAR(50)

DECLARE @version VARCHAR (1)

DECLARE @reporter VARCHAR(50)

DECLARE @listingExchange VARCHAR(50)

DECLARE @reportingDate DATE

DECLARE @PID1 XML

DECLARE @XMLheader XML

DECLARE @decimals INT

DECLARE @comment VARCHAR (MAX)

DECLARE @valueThreshold FLOAT

DECLARE @Multiplier FLOAT

-- INFLOWS

DECLARE @assetInflowReport XML

DECLARE @unsecuredInflowReport XML

DECLARE @otherInflowReport XML

DECLARE @securedInflowReport XML

-- OUTFLOWS

DECLARE @depositsOutflowReport XML

DECLARE @otherOutflowReport XML

DECLARE @securedOutflowReport XML

DECLARE @wholesaleOutflowReport XML

-- Supplemental

DECLARE @InformationalSupplemental XML

-- Comments

DECLARE @comments XML

DECLARE @ASOFDATE DATE

SET @unitMultiplier = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'unitMultiplier')

SET @decimals = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'decimals')

SET @unit = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'unit')

SET @header = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'header')

SET @version = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'version')

SET @reporter = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'reporter')

SET @listingExchange = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'listingExchange')

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

SET @comment = ''

SET @valueThreshold = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'valueThreshold')

SET @Multiplier = (SELECT FILTER

FROM SRC.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'XML'

AND FilterName = 'Multiplier')

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Purpose: Set each report to variables which will be used to execute

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @XMLheader = (

SELECT @version AS [@version]

,(SELECT @ListingExchange AS 'reporter/@listingExchange'

,@reporter AS 'reporter'

For XML Path(''), Type

)

,(SELECT @ASOFDATE AS reportingDate

For XML Path(''), Type

)

For XML Path('header'), Type);

-- Asset Inflow Report

SET @assetInflowReport = CAST((SELECT

Lower(treasuryControl) AS [@treasuryControl]

,RPT\_ID AS [@reportID]

,reportScope --AS 'testme/@treasuryControl'

,assetProduct

,assetSource

,(select UPPER(Currency) AS 'marketValue/@unit'

,@unitMultiplier AS 'marketValue/@unitMultiplier'

,Lower(converted) AS 'marketValue/@converted'

,@decimals AS 'marketValue/@decimals'

,CAST((ABS(marketvalue/@Multiplier))AS DECIMAL (18,2)) AS 'marketValue'

For XML Path(''), Type)

,(select UPPER(Currency) AS 'lendableValue/@unit'

,@unitMultiplier AS 'lendableValue/@unitMultiplier'

,Lower(converted) AS 'lendableValue/@converted'

,@decimals AS 'lendableValue/@decimals'

,CAST((ABS(lendableValue/@Multiplier))AS DECIMAL (18,2)) AS 'lendableValue'

For XML Path(''), Type)

,(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,ABS(forwardStartValue/@Multiplier) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,collateralClass

-- ,@comment AS 'comment'

FROM app.RPT\_FR2052a

WHERE assetInflowReport = 'Inflows-Assets'

-- AND marketValue >= @valueThreshold

FOR XML PATH('assetInflowReport'), Type

) AS VARCHAR(MAX));

-- Unsecured Inflow Report

SET @unsecuredInflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,LOWER(primeBrokerage) AS [@primeBrokerage]

,reportScope

,case when unsecuredProduct = 'Required Nostro Balances' then 'Required Nostro Accounts'

when unsecuredProduct = 'Excess Nostro Balances' then 'Excess Nostro Accounts'

WHEN unsecuredProduct = 'Outstanding Draws on Revolving Facilities' THEN 'Outstanding Draws on Revolving Credit Facilities'

else unsecuredProduct end as unsecuredProduct

,LTRIM(RTRIM(counterPartySector)) AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((ABS(maturityValue/@Multiplier))AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

, CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

WHEN maturityEnd IS NULL OR LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE NULL

END

,CASE WHEN forwardStartValue IS NOT NULL AND FmaturityStart IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN forwardStartValue IS NOT NULL AND FmaturityStart IS NOT NULL THEN (select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

-- ,@comment AS 'comment'

FROM app.RPT\_FR2052a

WHERE unsecuredInflowReport = 'Inflows-Unsecured'

-- and maturityValue > = @valueThreshold

FOR XML PATH('unsecuredInflowReport'), Type--, root ('')

) AS VARCHAR(MAX));

-- Other Inflow Report

SET @otherInflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,LOWER(primeBrokerage) AS [@primeBrokerage]

,Lower(treasuryControl) AS [@treasuryControl]

,reportScope --AS 'testme/@treasuryControl'

,CASE WHEN otherInflowProduct = 'Principal Payments on Unencumbered Investment Securities' THEN 'Principal Payments Receivable on Unencumbered Investment Securities'

ELSE otherInflowProduct END AS otherInflowProduct

,assetSource

,LTRIM(RTRIM(counterPartySector)) AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((ABS(maturityValue/@Multiplier))AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,ABS(forwardStartValue/@Multiplier) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN collateralClass IS NOT NULL OR LEN(collateralClass) > 0 OR collateralClass <> ''

THEN (SELECT collateralClass For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(ABS(collateralValue/@Multiplier) AS DECIMAL (19,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE otherInflowReport = 'Inflows-Other'

-- AND maturityValue > = @valueThreshold

FOR XML PATH('otherInflowReport'), Type

) AS VARCHAR(MAX));

-- SECURED Inflow Report

SET @securedInflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,LOWER(primeBrokerage) AS [@primeBrokerage]

,Lower(treasuryControl) AS [@treasuryControl]

,lower(unencumbered) AS [@unencumbered]

,reportScope --AS 'testme/@treasuryControl'

,securedInflowProduct

,pledgeLevel

,LTRIM(RTRIM(counterPartySector)) AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((ABS(maturityValue/@Multiplier))AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

-- Maturity Bucket

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

-- Effective Maturity

,CASE

WHEN EmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(EmaturityStart, '') AS maturityStart

,ISNULL(EmaturityEnd,'') AS maturityEnd

For XML Path('effectiveMaturityBucket'), Type)

WHEN EmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(EmaturityStart, '') AS maturityStart

For XML Path('effectiveMaturityBucket'), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(forwardStartValue/@Multiplier AS DECIMAL(18,2)) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN collateralClass IS NOT NULL OR LEN(collateralClass) > 0

THEN (SELECT collateralClass AS collateralClass for XML Path ('') , type)

ELSE NULL END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(collateralValue/@Multiplier AS DECIMAL (18,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,settlementMechanism AS 'settlementMechanism'

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE securedInflowReport = 'Inflows-secured'

-- AND maturityValue > = @valueThreshold

FOR XML PATH('securedInflowReport'), Type

) AS VARCHAR(MAX));

-- Deposit outflow Report

SET @depositsOutflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,insured AS [@insured]

,lower([trigger]) AS [@trigger]

,lower(internal) AS [@internal]

,lower(rehypothecated) AS [@rehypothecated]

,reportScope

,depositsProduct

,LTRIM(RTRIM(counterPartySector)) AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((ABS(maturityValue/@Multiplier))AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

-- Maturity Bucket

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

,CASE WHEN collateralClass IS NOT NULL OR LEN(collateralClass) > 0

THEN (SELECT collateralClass AS collaterallClass for XML Path ('') , type)

ELSE NULL END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(collateralValue/@Multiplier AS DECIMAL (18,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE depositsOutflowReport = 'Outflows-Deposits'

-- AND maturityValue > = @valueThreshold

FOR XML PATH('depositsOutflowReport'), Type

) AS VARCHAR(MAX));

-- Other outflow Report

SET @otherOutflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,LOWER(primeBrokerage) AS [@primeBrokerage]

,reportScope --AS 'testme/@treasuryControl'

,otherOutflowProduct

,LTRIM(RTRIM(counterPartySector)) AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((maturityValue/@Multiplier)AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(forwardStartValue/@Multiplier AS DECIMAL (18,2)) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN collateralClass IS NOT NULL THEN

(SELECT collateralClass AS collateralClass

For XML Path (''), type)

ELSE NULL

END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(ABS(collateralValue/@Multiplier) AS DECIMAL (19,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE otherOutflowReport = 'Outflows-other'

--AND maturityValue > = @valueThreshold

FOR XML PATH('otherOutflowReport'), Type

) AS VARCHAR(MAX));

-- SECURED OUTFLOWS

SET @securedOutflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(treasuryControl) AS [@treasuryControl]

,lower(rehypothecated) AS [@rehypothecated]

,lower(internal) AS [@internal]

,lower(primeBrokerage) AS [@primeBrokerage]

,reportScope --AS 'testme/@treasuryControl'

,securedOutflowProduct

,securedOutflowCounterpartyOrCoverage

,ISNULL(counterPartySector, '') AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((maturityValue/@Multiplier)AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(forwardStartValue/@Multiplier AS DECIMAL (18,2)) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN collateralClass IS NOT NULL THEN

(SELECT collateralClass AS collateralClass

For XML Path (''), type)

ELSE NULL

END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(collateralValue/@Multiplier AS DECIMAL (18,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,settlementMechanism

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE securedOutflowReport = 'Outflows-secured'

--AND maturityValue > = @valueThreshold

FOR XML PATH('securedOutflowReport'), Type

) AS VARCHAR(MAX));

-- WHOLESALE

SET @wholesaleOutflowReport = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,lower(primeBrokerage) AS [@primeBrokerage]

,reportScope

,wholesaleOutflowProduct

,CASE WHEN counterPartySector IS NOT NULL THEN LTRIM(RTRIM(counterPartySector))

END AS counterpartySector

,(select Currency AS 'maturityValue/@unit'

,@unitMultiplier AS 'maturityValue/@unitMultiplier'

,Lower(converted) AS 'maturityValue/@converted'

,@decimals AS 'maturityValue/@decimals'

,CAST((maturityValue/@Multiplier)AS DECIMAL (18,2)) AS 'maturityValue'

For XML Path(''), Type)

,CASE WHEN LEN(maturityEnd) > 0 AND maturityStart IS NOT NULL THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

,LTRIM(RTRIM(maturityEnd)) AS maturityEnd

For XML Path('maturityBucket'), Type)

when LEN(maturityEnd) = 0 THEN

(select LTRIM(RTRIM(maturityStart)) AS maturityStart

For XML Path('maturityBucket'), Type)

ELSE null

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL THEN (select UPPER(Currency) AS 'forwardStartValue/@unit'

,@unitMultiplier AS 'forwardStartValue/@unitMultiplier'

,Lower(converted) AS 'forwardStartValue/@converted'

,@decimals AS 'forwardStartValue/@decimals'

,CAST(forwardStartValue/@Multiplier AS DECIMAL (18,2)) AS 'forwardStartValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND LEN(FmaturityEnd) > 0 THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

,ISNULL(FmaturityEnd,'') AS maturityEnd

For XML Path('forwardStartBucket'), Type)

WHEN FmaturityStart IS NOT NULL AND forwardStartValue IS NOT NULL AND (LEN(FmaturityEnd) = 0 OR FmaturityEnd IS NULL) THEN

(select ISNULL(FmaturityStart, '') AS maturityStart

For XML Path('forwardStartBucket'), Type)

ELSE NULL

END

,CASE WHEN collateralClass IS NOT NULL THEN

(SELECT collateralClass AS collateralClass

For XML Path (''), type)

ELSE NULL

END

,CASE WHEN collateralValue IS NOT NULL AND collateralClass IS NOT NULL THEN (select UPPER(Currency) AS 'collateralValue/@unit'

,@unitMultiplier AS 'collateralValue/@unitMultiplier'

,Lower(converted) AS 'collateralValue/@converted'

,@decimals AS 'collateralValue/@decimals'

-- ,CAST((ABS(forwardStartValue/@Multiplier))AS DECIMAL (18,2)) AS 'forwardStartValue'

,CAST(collateralValue/@Multiplier AS DECIMAL (18,2)) AS 'collateralValue'

For XML Path(''), Type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE wholesaleOutflowReport = 'Outflows-wholesale'

-- AND maturityValue > = @valueThreshold

FOR XML PATH('wholesaleOutflowReport'), Type

) AS VARCHAR(MAX));

SET @InformationalSupplemental = CAST((SELECT

RPT\_ID AS [@reportID]

,lower(internal) AS [@internal]

,lower(primeBrokerage) AS [@primeBrokerage]

,reportScope

,supplementalInformationProduct

,segregationCategory

,settlementMechanism

,(select Currency AS 'marketValue/@unit'

,@unitMultiplier AS 'marketValue/@unitMultiplier'

,Lower(converted) AS 'marketValue/@converted'

,@decimals AS 'marketValue/@decimals'

,CAST((marketValue/@Multiplier)AS DECIMAL (18,2)) AS 'marketValue'

For XML Path(''), Type)

,CASE WHEN collateralClass IS NOT NULL THEN

(SELECT collateralClass AS collateralClass

For XML Path (''), type)

ELSE NULL

END

,CASE WHEN internalCounterparty IS NOT NULL OR LEN(internalCounterparty) = 0 OR internalCounterparty = ''

THEN (SELECT internalCounterparty For XML Path(''), Type)

ELSE NULL

END

FROM app.RPT\_FR2052a

WHERE informationalSupplementalReport = 'Supplemental-Informational'

-- AND marketvalue > = @valueThreshold

FOR XML PATH('informationalSupplementalReport'), Type

) AS VARCHAR(MAX));

WITH XMLNAMESPACES('http://www.w3.org/2001/XMLSchema-instance' AS xsi)

SELECT '5GSchema\_Combined.xsd' AS '@xsi:noNamespaceSchemaLocation'

,@XMLheader

----OUTFLOW REPORTS----

,@wholesaleOutflowReport

,@depositsOutflowReport

,@securedOutflowReport

,@otherOutflowReport

--INFLOW REPORTS----

,@securedInflowReport

,@otherInflowReport

,@assetInflowReport

,@unsecuredInflowReport

,@InformationalSupplemental

FOR XML PATH('fiveGSubmission');

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

--EXEC dbo.sp\_Ctrl\_Secured\_Inflow

--EXEC dbo.sp\_other\_Inflow @ASOFDATE = @ASOFDATE

--END TRY

--BEGIN CATCH

-- BEGIN

--Call procedure to handle ISSUEs

-- EXEC [dbo].[Proc\_InsertISSUEDetails]

-- END

-- END CATCH

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Purpose: Set each report to variables which will be used to execute

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @XMLheader =(

SELECT @version AS [@version]

,(SELECT @ListingExchange AS 'reporter/@listingExchange'

,@reporter AS 'reporter'

For XML Path(''), Type)

,(SELECT @ASOFDATE AS reportingDate

For XML Path(''), Type)

For XML Path('header'), Type);

SET @comments = CAST((SELECT

reportScope AS reportScope

,(select [type] AS [@type]

,(select product AS product For XML Path(''), Type)

,CASE WHEN LEN(subproduct) > 0 OR subproduct <> ''

THEN (SELECT subproduct For XML Path(''), Type)

ELSE NULL

END

FOR XML Path('report'), Type)

,text as text

FROM SRC.US\_FR2052A\_REF\_Comments

FOR XML PATH('comment'), Type, root ('comments')

) AS VARCHAR(MAX));

WITH XMLNAMESPACES('http://www.w3.org/2001/XMLSchema-instance' AS xsi)

--WITH XmlNamespaceManager ('http://www.w3.org/2001/XMLSchema-instance' AS xsi)

--SELECT 'C:\Users\KBLACK\Documents\SQL Server Management Studio\Projects\XML\5GSchema.xsd' AS '@xsi:noNamespaceSchemaLocation',@XMLheader, @assetInflowReport, @unsecuredInflowReport

--FOR XML PATH('fiveGSubmission');

SELECT '5GSchema.xsd' AS '@xsi:noNamespaceSchemaLocation',@XMLheader, @comments

FOR XML PATH('fiveGComments');

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_wholesale\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_wholesale\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_wholesale\_outflow]

\*\* Report Type: Wholesale Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate\_1 Date

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR(10)

DECLARE @Settlement VARCHar(20)

DECLARE @CompanyName VARCHAR (50) -- NEW Variable 20170327 KBLack O.W.9

DECLARE @CompanyName1 VARCHAR (50) -- NEW Variable 20170327 KBLack O.W.9

DECLARE @AL VARCHAR(5)

DECLARE @BR VARCHAR(5)

DECLARE @CountryCode VARCHAR(50)

DECLARE @SCHTYPE VARCHAR(10)

DECLARE @REPORTENTITY VARCHAR(50)

DECLARE @CCYAMT INT

DECLARE @REVREASON VARCHAR(10)

DECLARE @OFFSHORE VARCHAR(10)

DECLARE @Onshore VARCHAR(10)

DECLARE @PrinAmt INT

DECLARE @UnamortAmt INT

DECLARE @SettQty INT

DECLARE @SettavgCost INT

DECLARE @InvType VARCHAR(10)

DECLARE @AcctngType VARCHAR(10)

DECLARE @MktVal INT

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.9 (NYB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.W.9'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @CountryCode = (SELECT DISTINCT Filter -- CountryCode = 'US'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CountryCode'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REVREASON = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'REVREASON'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT'

)

SET @SCHTYPE = (SELECT DISTINCT Filter -- SCHDTYPE = 'M'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SCHTYPE'

)

SET @AL = 'AL'

SET @BR = 'BR'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'SCHD | '+schd.DEALNO+' | '+schd.BR+' | '+CAST(schd.INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ

AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

,REF\_REP.Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.CounterParty AS [CounterParty]

,DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day 60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

-- Report Totpayamt only if VDATE > reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.TotPayAmt), @AsofDate)

ELSE NULL

END AS [ForwardStart]

--Report VDATE only if VDATE>reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

(

CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, '> 1')

END

END

END

)

ELSE NULL

END AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterparty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName] -- Reasonableness Test

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN DLDT.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN DLDT.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

, ((SELECT CASE WHEN DLDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.PrinAmt AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,DLDT.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_OPICS\_DLDT DLDT

LEFT JOIN src.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND DLDT.ASOFDATE = CUST.ASOFDATE

LEFT JOIN src.[US\_OPICS\_SCHD] SCHD

ON SCHD.DEALNO = DLDT.DEALNO

AND CUST.ASOFDATE = SCHD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = DLDT.CNO

WHERE

-- To avoid FR2052a Validation fail - where Reporting Entity is not the same as internal counterparty

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @REVREASON

AND DLDT.CCYAMT <> @CCYAMT

AND SCHD.SCHDTYPE = @SCHTYPE

AND DLDT.MDATE > @AsofDate

AND DLDT.ASOFDATE = @AsofDate

AND CUST.CCODE = @CountryCode

AND SCHD.CCY = @Currency

AND DLDT.AL IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding

-- \*\* PID: O.W.9 (CHB and LAB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.W.9'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @Category = (SELECT DISTINCT Filter --Category = 21038

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Category'

AND Module = 'O.W.9\_CHB\_LAB'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @CompanyName1 = (SELECT DISTINCT Filter -- CompanyName = CHI

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CompanyName'

AND Module = 'CHI')

SET @CompanyName = (SELECT DISTINCT Filter -- CompanyName = LAB

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CompanyName'

AND Module = 'LAB')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterparty]

,CASE WHEN @PrimeBrokeRage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName] -- Reasonableness Test

,CASE WHEN CUST.CUSTOMER\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo-- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,CUST.CUSTOMER\_CODE AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.[t\_fmddata] FMD

INNER JOIN SRC.COMPANY COMP

ON COMP.COMPANY\_CODE = FMD.COMPANY\_CODE

AND COMP.ASOFDATE = FMD.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = COMP.company\_code

INNER JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = FMD.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = FMD.Customer\_No

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = fmd.CUSTOMER\_NO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.ASOFDATE = @ASOFDATE

AND FMD.Category = @Category

AND FMD.Company IN (@CompanyName, @CompanyName1)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.W.10'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @CountryCode = (SELECT DISTINCT Filter -- CountryCode = 'US'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CountryCode'

)

SET @REVREASON = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'REVREASON'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMt'

)

SET @SCHTYPE = (SELECT DISTINCT Filter -- SCHDTYPE = 'M'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SCHTYPE'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @AL = 'AL'

SET @BR = 'BR'

SET @Onshore = 'Onshore'

SET @OFFSHORE = 'Offshore'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ

AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, REF\_REP.Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [Counterparty]

,DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

-- Report Totpayamt only if VDATE > reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.TotPayAmt), @AsofDate)

ELSE NULL

END AS [ForwardStart]

--Report VDATE only if VDATE>reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

(

CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day 60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, '> 1')

END

END

END

)

ELSE NULL

END AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterparty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN DLDT.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN DLDT.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN DLDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.PrinAmt AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,DLDT.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_DLDT DLDT

LEFT JOIN SRC.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND DLDT.ASOFDATE = CUST.ASOFDATE

LEFT JOIN SRC.[US\_OPICS\_SCHD] SCHD

ON SCHD.DEALNO = DLDT.DEALNO

AND CUST.ASOFDATE = SCHD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = DLDT.CNO

-- To avoid FR2052a Validation fail - where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @REVREASON

AND DLDT.CCYAMT <> @CCYAMT

AND SCHD.SCHDTYPE = @SCHTYPE

AND DLDT.MDATE > @AsofDate

AND DLDT.ASOFDATE = @AsofDate

AND DLDT.AL IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND (

(

DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

AND Module = @Onshore

)

AND (

(

CUST.CCODE = @CountryCode AND SCHD.CCY <> @Currency

)

OR

(

CUST.CCODE <> @CountryCode AND SCHD.CCY = @Currency

)

OR

(

CUST.CCODE <> @CountryCode AND SCHD.CCY <> @Currency

)

)

)

OR

(

DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

AND Module = @Offshore

)

)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.11

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.W.11'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'Y'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @InvType = (SELECT DISTINCT Filter -- InvType = 'I'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'InvType'

)

SET @AcctngType = (SELECT DISTINCT Filter -- ACCTNGTYPE = Issue (representing bond)

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'AcctngType'

)

SET @PrinAmt = (SELECT DISTINCT Filter -- Prinamt is not 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrinAmt'

)

SET @UnamortAmt = (SELECT DISTINCT Filter -- unamortamt is not 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'UnamortAmt'

)

SET @MktVal = (SELECT DISTINCT Filter -- mktval is not 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'MktVal'

)

SET @SettQty = (SELECT DISTINCT Filter -- SettQTY is not 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SettQty'

)

SET @SettavgCost = (SELECT DISTINCT Filter -- settavgcost is not 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SettavgCost'

)

SET @REVREASON = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'REVREASON'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT'

)

SET @SCHTYPE = (SELECT DISTINCT Filter -- SCHDTYPE = 'M'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SCHTYPE'

)

SET @AL = 'AL'

SET @BR = 'BR'

IF OBJECT\_ID(N'tempdb..#Temp\_STBRPTCD') IS NOT NULL

BEGIN

DROP TABLE #Temp\_STBRPTCD

END

CREATE TABLE #Temp\_STBRPTCD

(

ID VARCHAR(255)

,br VARCHAR(5)

,ccy VARCHAR(20)

,secid VARCHAR(20)

,CNO VARCHAR(20)

,ctype VARCHAR(20)

,mdate DATE

,TPOS\_Prinamt VARCHAR(20) -- Maturity Date

,prinamt DECIMAL(30,15)

,ASOFDATE DATE

)

INSERT INTO #Temp\_STBRPTCD

(

ID

,br

,ccy

,secid

,CNO

,ctype

,mdate

,TPOS\_Prinamt -- Maturity Date

,Prinamt

,ASOFDATE

)

SELECT

'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(B.BR)) + ' | ' + LTRIM(RTRIM(B.SECID)) + ' | ' + CAST(LTRIM(RTRIM(B.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(B.PORT)) + ' | ' + LTRIM(RTRIM(B.INVTYPE)) + ' | ' + CAST(B.COST AS VARCHAR (50))))

, B.br

, B.ccy'CCY'

, B.secid

,D.CNO

,D.CTYPE

, A.mdate

,B.PrinAmt

,DBO.fn\_CurrencyExchange

(B.CCY,(B.PrinAmt), @AsofDate) AS [maturityValue]

, A.ASOFDATE

FROM SRC.US\_OPICS\_secm A,

SRC.US\_OPICS\_tpos B,

SRC.US\_OPICS\_cust D

WHERE B.invtype = @InvType -- Issued investment type

AND A.issuer = D.cno

AND A.acctngtype = @AcctngType

AND A.secid = B.secid

AND B.br in (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

AND (B.prinamt <> @PrinAmt

OR B.unamortamt <> @UnamortAmt

OR B.settavgcost <> @SettavgCost

OR B.mktval <> @mktval)

AND B.settqty <> @SettQty

AND A.MDATE >@AsofDate

--AND D.CMNE IN ('BOCNY' ) --Customer name is BOCNY or BOCNYIBF to identify BOC issued BONDs

--Above D.CMNE condition is temporary and should be changed to new criteria using A.DESCR.

--A.DESCR field is currently being created by OSD (Liu Li) in Production environment

AND A.MDATE > @AsofDate

AND A.ASOFDATE = B.ASOFDATE

AND B.ASOFDATE = D.ASOFDATE

AND A.ASOFDATE = @AsofDate

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

STB.ID AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN STB.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,NULL AS [CID]

,NULL

AS [Counterparty]

,STB.PRINAMT AS [MaturityAmount] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,STB.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,STB.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,STB.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,STB.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,STB.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,STB.MDATE, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterparty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN STB.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN STB.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN STB.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN STB.CCY IS NULL OR STB.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN STB.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,STB.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,STB.PRINAMT AS orig\_maturityValue -- Reasonableness Test

,STB.CCY AS orig\_currency -- Reasonableness Test

,STB.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM #Temp\_STBRPTCD STB

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = STB.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = STB.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = STB.CNO

--AND ref\_Internal\_cp.InternalCounterParty <> ref\_en.reportingentityDescription

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND STB.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.W.16 (Principal -STB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.W.16'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate\_1 = @asofdate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @InvType = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND FilterName = 'INVTYPE')

SET @AcctngType = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND FilterName = 'ACCTNGTYPE')

SET @BR = 'BR'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101)) AS [SrcID] -- Represents Principal

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

, NULL AS [CID]

, NULL AS [counterPartySector]

,CASE WHEN TPOS.CCY = 'USD' THEN (TPOS.PRINAMT)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,(TPOS.PRINAMT), @AsofDate)) END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate\_1,SECM.MDATE, 'Day 1')

AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage ='Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate\_1 AS [AsofDate]

,(

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

-- SECS is a schedule table that shows the payment schedule

-- SECM is security master table like T\_FMDDATA table. Give description or attribute data related to securities

--SELECT TPOS.\*, 'TPOS---BREAK---TPOS' AS SEPARATOR, SECM.\*--, 'SECS---BREAK---SECS' AS SEPARATOR, SECM.\*

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,TPOS.PRINAMT AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,SECM.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = TPOS.BR

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND TPOS.ASOFDATE = @AsofDate

AND SECM.MDATE > = @asofdate

AND TPOS.BR IN (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR)

AND TPOS.INVTYPE = @InvType

AND SECM.ACCTNGTYPE = @AcctngType

-- AND TPOS.SETTQTY <> 0

-- AND TPOS.SETTAVGCOST <> 0

AND TPOS.PRINAMT <> 0

AND TPOS.PRINAMT IS NOT NULL

AND TPOS.ASOFDATE = SECM.ASOFDATE

AND TPOS.ASOFDATE = @AsofDate

AND SECM.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.W.16 (Interest -STB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.W.16'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @InvType = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND FilterName = 'INVTYPE')

SET @AcctngType = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND FilterName = 'ACCTNGTYPE')

SET @BR = 'BR'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101)) AS [SrcID] -- Represents Interest

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

,NULL AS [CID]

,NULL AS [counterPartySector]

,CASE WHEN TPOS.CCY = 'USD' THEN (TPOS.PRINAMT)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,(TPOS.QTY\*SECS.INTPAYAMT\_8), @AsofDate)) END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,SECS.IPAYDATE, 'Day 1')

AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage ='Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,TPOS.PRINAMT AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,SECS.IPAYDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECS SECS

ON SECS.SECID = SECM.SECID

AND SECS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = 'O.W.16'

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = TPOS.BR

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND TPOS.ASOFDATE = @AsofDate

AND SECM.MDATE > = @AsofDate

AND TPOS.BR IN (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR)

AND TPOS.INVTYPE = @InvType

AND SECM.ACCTNGTYPE = @AcctngType

AND TPOS.ASOFDATE = SECM.ASOFDATE

AND SECS.ASOFDATE = SECM.ASOFDATE

AND SECS.INTPAYAMT\_8 <> 0

AND SECS.INTPAYAMT\_8 IS NOT NULL

AND TPOS.QTY <> 0

AND TPOS.QTY IS NOT NULL

AND SECS.IPAYDATE >= @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Trade Finance

-- \*\* PID: TSD - O.W.19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.W.19'

SET @Portfolio = 'Trade Finance'

SET @Currency = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter -- TreasuryControl = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @DESC1 = 'DESC1'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,NULL AS [CID]

,NULL AS [Counterparty]

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterparty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.t\_fmddata FMD

LEFT JOIN SRC.COMPANY COMP

ON COMP.COMPANY\_CODE = FMD.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = COMP.COMPANY\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = FMD.sector

AND ref\_CP.ReportingEntity = ref\_en.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND FMD.DEAL\_BALANCE IS NOT NULL

-- AND FMD.Category = @Category

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.ASOFDATE = @ASOFDATE

AND FMD.DESC1 IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

and Portfolio = @Portfolio

AND FilterName = @DESC1)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Non - Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other payables and margin Collaterals

-- \*\* PID: O.W.18

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.W.18'

SET @Portfolio = 'Other Payables and Margin CollateralS'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'NB\_OW | '+CAST(NB\_OW.ID AS VARCHAR) + ' | O.W.'+ CAST(NB\_OW.PID\_ProductID AS VARCHAR) + ' | '+NB\_OW.ReportingEntity +' | '+CAST(NB\_OW.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [CID]

,NULL AS [CounterParty]

,dbo.fn\_CurrencyExchange

(NB\_OW.OriginalCurrencyOfMaturityAmount,(NB\_OW.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OW.MaturityDate, 'Open')

AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [ColletaralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NB\_OW.MaturityDate AS [MaturityDATE]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_OW.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,'N/A' AS [DESC1]

,REF\_EN.Company\_name AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_OW.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_OW.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_OW.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_OW.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_OW NB\_OW

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_OW.Reportingentity

AND NB\_OW.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_OW.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_OW.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_OW.MaturityAmountOriginalCurrency <> 0

AND NB\_OW.Asofdate = @ASOFDATE

AND 'O.W.'+CAST(NB\_OW.PID\_ProductID AS VARCHAR) = @PID

and NB\_OW.PortfolioName = @Portfolio

AND nb\_OW.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: O.W.19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.W.19'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_wholesaleOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'NB\_OW | '+CAST(NB\_OW.ID AS VARCHAR) + ' | O.W.'+ CAST(NB\_OW.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OW.ReportingEntity +' | '+CAST(NB\_OW.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [CID]

,NULL AS [CounterParty]

,dbo.fn\_CurrencyExchange

(NB\_OW.OriginalCurrencyOfMaturityAmount,(NB\_OW.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue]-- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OW.MaturityDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OW.MaturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OW.MaturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OW.MaturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OW.MaturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OW.MaturityDate, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [ColletaralValue]

,NULL AS [CollateralCurrency]

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_OW.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_OW.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_OW.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OW.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_OW.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_OW.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,NB\_OW.MaturityDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_OW NB\_OW

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_OW.Reportingentity

AND NB\_OW.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_OW.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_OW.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_OW.MaturityAmountOriginalCurrency <> 0

AND NB\_OW.Asofdate = @ASOFDATE

AND 'O.W.'+CAST(NB\_OW.PID\_ProductID AS VARCHAR) = @PID

and NB\_OW.PortfolioName = @Portfolio

AND NB\_OW.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Unsecured\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Unsecured\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Unsecured\_Inflow]

\*\* Report Type: Unsecured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Entity') AND NAME ='IX\_Ref\_En')

DROP INDEX IX\_Ref\_En ON app.US\_FR2052A\_REF\_Entity;

CREATE NONCLUSTERED INDEX IX\_Ref\_En

ON app.US\_FR2052A\_REF\_Entity(Company\_Code);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_T24\_MULTI\_LD\_BALANCE') AND NAME ='IX\_T24\_BAL')

DROP INDEX IX\_T24\_BAL ON SRC.US\_T24\_MULTI\_LD\_BALANCE;

CREATE NONCLUSTERED INDEX IX\_T24\_BAL

ON src.US\_T24\_MULTI\_LD\_BALANCE(ID,CO\_CODE,ASOFDATE);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.MULTI\_LD') AND NAME ='IX\_MULTI\_LD')

DROP INDEX IX\_MULTI\_LD ON SRC.MULTI\_LD ;

CREATE NONCLUSTERED INDEX IX\_MULTI\_LD

ON src.MULTI\_LD(ID,CO\_CODE,ASOFDATE, CATEGORY);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.FBNK\_CUSTOMER') AND NAME ='IX\_CUST')

DROP INDEX IX\_CUST ON SRC.FBNK\_CUSTOMER;

CREATE NONCLUSTERED INDEX IX\_CUST

ON src.FBNK\_CUSTOMER(ASOFDATE, ID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @DESC11 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate\_1 Date

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @BR VARCHAR(5)

DECLARE @AL VARCHAR(5)

--The following variables are for Commercial and Retail Loans Portfolio

--declaRe categories to be excluded

DECLARE @TradeFinanceCategory VARCHAR(10) -- These are already captured in Trade finance Portfolio

DECLARE @SyndicatedLoanCategory VARCHAR(10) -- syndicated term loans that need to be excluded

DECLARE @CollateralCode VARCHAR(50)

DECLARE @CountryCode VARCHAR(50)

DECLARE @ReportEntity VARCHAR(50)

DECLARE @RevReason VARCHAR(10)

DECLARE @CCYAMT INT

DECLARE @SCHDTYPE VARCHAR(10)

DECLARE @OnshoreBR VARCHAR(10)

DECLARE @OffshoreBR VARCHAR(10)

DECLARE @SCHTYPE VARCHAR(10)

DECLARE @Revolving INT

DECLARE @Nonrevolving INT

DECLARE @FixedVariable VARCHAR(20)

DECLARE @Module VARCHAR(20)

DECLARE @CustomerCode VARCHAR(20)

DECLARE @Contact\_id VARCHAR(20)

DECLARE @CustomerCode1 VARCHAR(20)

DECLARE @Contact\_id1 VARCHAR(20)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: I.U.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'I.U.1'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @CountryCode = (SELECT DISTINCT Filter -- CountryCode = 'US'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CountryCode'

)

SET @RevReason = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'RevReason'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT'

)

SET @SCHDTYPE = (SELECT DISTINCT Filter -- SCHDTYPE = 'M'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SCHDTYPE'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @BR = 'BR'

SET @AL = 'AL'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_unsecuredInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'SCHD | '+ schd.DEALNO+' | '+schd.BR+' | '+CAST(INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,dbo.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) <= 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

-- Report PROCEEDAMT only if VDATE > reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

dbo.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.TOTPAYAMT), @AsofDate)

ELSE NULL

END AS [ForwardStartAmount]

--Report VDATE only if VDATE>reporting date

,CASE WHEN DLDT.VDATE>@AsofDate

THEN

(CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, '> 5')

END

END

END

)

ELSE NULL

END AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN DLDT.BR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN DLDT.DEALNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN DLDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.PrinAmt AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,DLDT.VDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM src.US\_OPICS\_DLDT DLDT

LEFT JOIN src.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND DLDT.ASOFDATE = CUST.ASOFDATE

LEFT JOIN src.[US\_OPICS\_SCHD] SCHD

ON SCHD.DEALNO = DLDT.DEALNO

AND SCHD.ASOFDATE = DLDT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.cno = DLDT.CNO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @RevReason

AND DLDT.CCYAMT <> @CCYAMT

AND SCHD.SCHDTYPE = @SCHDTYPE

AND DLDT.MDATE > @AsofDate

AND DLDT.ASOFDATE = @AsofDate

AND DLDT.ASOFDATE = SCHD.AsOfDate

AND CUST.CCODE = @CountryCode

AND SCHD.CCY = @Currency

AND DLDT.AL IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: I.U.2 (STB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'I.U.2'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @CountryCode = (SELECT DISTINCT Filter -- CountryCode = 'USA'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CountryCode'

)

SET @RevReason = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'RevReason'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT'

)

SET @SCHDTYPE = (SELECT DISTINCT Filter -- SCHDTYPE = 'M'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SCHDTYPE'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @BR = 'BR'

SET @OnshoreBR = 'Onshore'

SET @OffshoreBR = 'Offshore'

SET @AL = 'AL'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_unsecuredInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'SCHD | '+schd.DEALNO+' | '+schd.BR+' | '+CAST(INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ

AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.MDATE, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

-- Report PROCEEDAMT only if VDATE > reporting date

,CASE WHEN DLDT.VDate>@AsofDate

THEN

DBO.fn\_CurrencyExchange(SCHD.CCY,(SCHD.TotPayAmt), @AsofDate)

ELSE NULL

END AS [ForwardStart]

--Report VDATE only if VDATE>reporting date

,CASE WHEN DLDT.VDate>@AsofDate

THEN

(

CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate)<=60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,DLDT.VDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,DLDT.VDate, '> 5')

END

END

END

)

ELSE NULL

END AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN DLDT.BR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN DLDT.DEALNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN DLDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.PrinAmt AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,DLDT.VDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_OPICS\_DLDT DLDT

LEFT JOIN src.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND DLDT.ASOFDATE = CUST.ASOFDATE

LEFT JOIN src.[US\_OPICS\_SCHD] SCHD

ON SCHD.DEALNO = DLDT.DEALNO

AND DLDT.ASOFDATE = SCHD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.cno = DLDT.CNO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @RevReason

AND DLDT.CCYAMT <> @CCYAMT

AND SCHD.SCHDTYPE = @SCHDTYPE

AND DLDT.MDATE > @AsofDate

AND DLDT.ASOFDATE = @AsofDate

--AND CUST.CCODE = @CountryCode

AND DLDT.AL IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND (

(

DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

AND Module = @OnshoreBR

)

AND (

(

CUST.CCODE = @CountryCode AND SCHD.CCY <> @Currency

)

OR

(

CUST.CCODE <> @CountryCode AND SCHD.CCY = @Currency

)

OR

(

CUST.CCODE <> @CountryCode AND SCHD.CCY <> @Currency

)

)

)

OR

(

DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

AND Module = @OffshoreBR

)

)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: I.U.2 (DataMart)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'I.U.2'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @DESC1 = 'DESC1'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @BR = 'BR'

SET @OnshoreBR = 'Onshore'

SET @OffshoreBR = 'Offshore'

SET @AL = 'AL'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_unsecuredInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN FMD.CURRENCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,DBO.fn\_CurrencyExchange

(FMD.CURRENCY,(FMD.DEAL\_BALANCE), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,FMD.DEAL\_MAT\_DATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,FMD.DEAL\_MAT\_DATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,FMD.DEAL\_MAT\_DATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,FMD.DEAL\_MAT\_DATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,FMD.DEAL\_MAT\_DATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,FMD.DEAL\_MAT\_DATE, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN FMD.CUSTOMER\_NO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN FMD.DEAL\_MAT\_DATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = FMD.COMPANY\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = FMD.SECTOR

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.CUSTOMER\_NO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND FMD.ASOFDATE = @AsOfDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.3 - Multi - Account

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--Enter PID and Portfolio for CASH

SET @PID = 'I.U.3'

SET @Portfolio = 'CASH'

SET @Sector = 'Sector'

set @Module = 'Multi-Account'

set @DESC11 = (SELECT DISTINCT Filter -- DESC1 = 0060

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = 'DESC1\_Exception'

and Module = @Module

)

SET @DESC1 = 'DESC1' -- DESC1 = 0075, 0090, 0100, 0105, 2360, 2410, 2550, or 2560

SET @Category = 'Category'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'Multi-Account | '+cast(MA.ID as varchar) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,dbo.fn\_CurrencyExchange(MA.CURRENCY, MA.EXPECT\_BALANCE, @ASOFDATE)

AS [MaturityValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] --Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,[DESC1] AS [DESC1]

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,fmd.Deal\_Mat\_date AS [MaturityDate]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN MA.ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(SELECT

CASE WHEN MA.CURRENCY IS NULL OR MA.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN MA.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

INNER JOIN SRC.MULTI\_ACCOUNT MA

ON MA.CUSTOMER = FMD.CUSTOMER\_NO

AND MA.CATEGORY = FMD.CATEGORY

AND MA.ID = FMD.CONTACT\_ID

AND FMD.ASOFDATE = MA.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = fmd.CUSTOMER\_NO

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO

WHERE

(ISNULL(ref\_Internal\_cp.InternalCounterParty,''))<> ref\_en.ReportingEntityDescription

AND FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND ((DESC1 = @DESC11 AND FMD.SECTOR <> (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = @Sector

))

OR (DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = @desc1

and Module = @Module

)

)

)

AND EXPECT\_BALANCE IS NOT NULL

AND EXPECT\_BALANCE <> 0

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.3 - FMD

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.U.3'

SET @Portfolio = 'CASH'

SET @Sector = 'Sector'

SET @DESC1 = (SELECT DISTINCT Filter -- DESC1 = 2360

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC1'

AND Module = 'FMD'

)

SET @CustomerCode = (SELECT DISTINCT Filter -- Customerid = 1000791

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CUST'

AND Module = 'FMD'

)

SET @Contact\_id = (SELECT DISTINCT Filter -- Contact id = 01019796

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Contact'

AND Module = 'FMD'

)

SET @DESC11 = (SELECT DISTINCT Filter -- DESC1 = 2410

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC11'

AND Module = 'FMD'

)

SET @CustomerCode1 = (SELECT DISTINCT Filter -- Customerid = 1000798

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CUST1'

AND Module = 'FMD'

)

SET @Contact\_id1 = (SELECT DISTINCT Filter -- Contact id = 01019797

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Contact1'

AND Module = 'FMD'

)

SET @Category = 'Category'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,dbo.fn\_CurrencyExchange(fmd.CURRENCY, fmd.deal\_balance, @ASOFDATE)

AS [MaturityValue] --Calcuation performed with Foreign Currency Exchange functi

,dbo.fn\_MaturityDateCalculater(@ASOFDATE,(SELECT case when @ASOFDATE between '1/1/'+cast(YEAR(@ASOFDATE)as varchar) and '1/25/'+cast(YEAR(@ASOFDATE)as varchar)

then '1/25/'+cast(YEAR(@ASOFDATE)as varchar)

else case when @ASOFDATE between '1/26/'+cast(YEAR(@ASOFDATE)as varchar) and '4/25/'+cast(YEAR(@ASOFDATE)as varchar)

then '4/25/'+cast(YEAR(@ASOFDATE)as varchar)

else case when @ASOFDATE between '4/26/'+cast(YEAR(@ASOFDATE)as varchar) and '7/25/'+cast(YEAR(@ASOFDATE)as varchar)

then '7/25/'+cast(YEAR(@ASOFDATE)as varchar)

else case when @ASOFDATE between '7/26/'+cast(YEAR(@ASOFDATE)as varchar) and '10/25/'+cast(YEAR(@ASOFDATE)as varchar)

then '7/25/'+cast(YEAR(@ASOFDATE)as varchar)

else case when @ASOFDATE between '10/26/'+cast(YEAR(@ASOFDATE)as varchar) and '1/25/'+cast(YEAR(@ASOFDATE)+1 as varchar)

then '1/25/'+cast(YEAR(@ASOFDATE)+1 as varchar)

end

end

end

END

end ),'0-100') AS [MaturityBucket] --Matuity date = the next future roll-over date which is among April 25, July 25, October 25, and January 25. Maturity date needs to be bucketed into the appropriate FR2052a bucket

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,[DESC1] AS [DESC1]

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,fmd.Deal\_Mat\_date AS [MaturityDate]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.Contact\_ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN fmd.CURRENCY IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = fmd.CUSTOMER\_NO

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO

WHERE

(ISNULL(ref\_Internal\_cp.InternalCounterParty,''))<> ref\_en.ReportingEntityDescription

AND

FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND ((DESC1 = @DESC1

AND CUSTOMER\_NO = @CustomerCode

AND CONTACT\_ID = @Contact\_ID

)

or

(DESC1 = @DESC11

AND CUSTOMER\_NO = @CustomerCode1

AND CONTACT\_ID = @Contact\_id1

)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--Enter PID and Portfolio for CASH

-- SET VARIABLES

-- SET VARIABLES

SET @PID = 'I.U.4'

SET @PID1 = 'I.U.3'

SET @Portfolio = 'CASH'

SET @Sector = 'Sector'

set @DESC11 = (SELECT DISTINCT Filter -- DESC1 = 0060

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid1

AND Portfolio = @Portfolio

AND FilterName = 'DESC1\_Exception'

and Module = @Module

)

SET @DESC1 = 'DESC1'

SET @Category = 'Category'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'Multi-Account | '+cast(MA.ID as varchar) AS [SrcID] -- Represents I.U.3 Multi-Account

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,dbo.fn\_CurrencyExchange(MA.CURRENCY, MA.EXPECT\_BALANCE, @ASOFDATE)

AS [MaturityValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] --Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,[DESC1] AS [DESC1]

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,fmd.Deal\_Mat\_date AS [MaturityDate]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN MA.ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(SELECT

CASE WHEN MA.CURRENCY IS NULL OR MA.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN MA.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,MA.EXPECT\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS orig\_date -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

INNER JOIN SRC.MULTI\_ACCOUNT MA

ON MA.CUSTOMER = FMD.CUSTOMER\_NO

AND MA.CATEGORY = FMD.CATEGORY

AND MA.ID = FMD.CONTACT\_ID

AND FMD.ASOFDATE = MA.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = fmd.CUSTOMER\_NO

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO

WHERE

(ISNULL(ref\_Internal\_cp.InternalCounterParty,''))<> ref\_en.ReportingEntityDescription

AND FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND ((DESC1 = @DESC11 AND FMD.SECTOR <> (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = @Sector

))

OR (DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = @desc1

)

)

)

AND EXPECT\_BALANCE IS NOT NULL

AND EXPECT\_BALANCE <> 0

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*I.U.3 FMD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID1 = 'I.U.3'

SET @DESC1 = (SELECT DISTINCT Filter -- DESC1 = 2360

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'DESC1'

AND Module = 'FMD'

)

SET @CustomerCode = (SELECT DISTINCT Filter -- Customerid = 1000791

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'CUST'

AND Module = 'FMD'

)

SET @Contact\_id = (SELECT DISTINCT Filter -- Contact id = 01019796

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'I.U.3'

AND Portfolio = 'CASH'

AND FilterName = 'Contact'

AND Module = 'FMD'

)

SET @DESC11 = (SELECT DISTINCT Filter -- DESC1 = 2410

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'DESC11'

AND Module = 'FMD'

)

SET @CustomerCode1 = (SELECT DISTINCT Filter -- Customerid = 1000798

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = 'CASH'

AND FilterName = 'CUST1'

AND Module = 'FMD'

)

SET @Contact\_id1 = (SELECT DISTINCT Filter -- Contact id = 01019797

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'Contact1'

AND Module = 'FMD'

)

SET @Category = 'Category'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR) AS [SrcID] -- Represents IU3 FMD

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,dbo.fn\_CurrencyExchange(fmd.CURRENCY, fmd.deal\_balance, @ASOFDATE)

AS [MaturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,[DESC1] AS [DESC1]

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,fmd.Deal\_Mat\_date AS [MaturityDate]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.Contact\_ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN fmd.CURRENCY IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = fmd.CUSTOMER\_NO

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO

WHERE

(ISNULL(ref\_Internal\_cp.InternalCounterParty,''))<> ref\_en.ReportingEntityDescription

AND

FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND ((DESC1 = @DESC1

AND CUSTOMER\_NO = @CustomerCode

AND CONTACT\_ID = @Contact\_ID

)

or

(DESC1 = @DESC11

AND CUSTOMER\_NO = @CustomerCode1

AND CONTACT\_ID = @Contact\_id1

)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*i.u.4 Total\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.4'

SET @PID1 = 'i.u.3'

SET @Portfolio = 'CASH'

SET @Sector = (SELECT DISTINCT Filter -- Sector = 3200

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'Sector'

)

set @DESC1 = (SELECT DISTINCT Filter -- DESC1 = 0060

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'DESC1\_Exception'

and Module = 'Multi-Account'

)

SET @DESC11 = 'DESC1' -- DESC1 = 0075, 0090, 0100, 0105, 2550, or 2560

SET @Module = 'Total'

SET @Category = 'Category'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FMD\_ALL | '+ CAST(fmd.n\_ID AS VARCHAR) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,(dbo.fn\_CurrencyExchange(fmd.CURRENCY,fmd.DEAL\_BALANCE, @ASOFDATE) )

AS [MaturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,[DESC1] AS [DESC1]

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,fmd.Deal\_Mat\_date AS [MaturityDate]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.Contact\_ID IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN fmd.CURRENCY IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

INNER JOIN SRC.MULTI\_ACCOUNT MA

ON MA.CUSTOMER = FMD.CUSTOMER\_NO

AND MA.CATEGORY = FMD.CATEGORY

AND MA.ID = FMD.CONTACT\_ID

AND MA.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = fmd.CUSTOMER\_NO

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO COLLATE Latin1\_General\_CI\_AS

WHERE

(ISNULL(ref\_Internal\_cp.InternalCounterParty,''))<> ref\_en.ReportingEntityDescription

AND

FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND(( DESC1 = @desc1 AND FMD.SECTOR <> @sector)

OR DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = @desc11

and Module = @module

)

)

ORDER BY N\_ID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.U.5 - Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.5'

SET @PID1 = 'I.S.7' -- To identofy secured loans and eliminate them

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @CollateralCode = 'CollateralCode' -- Collateral\_Type = 501, 502, 503, 504, 600, 700 or 800

SET @FixedVariable = (SELECT DISTINCT Filter -- Fixed variable = 'FIXED'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter -- Synicated Loan Catoegory = 21067

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

SET @TradeFinanceCategory = 'CATEGORY'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @SCHTYPE = 'SCHTYPE'

SET @Revolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @DESC1 = 'DESC1'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

IF OBJECT\_ID('tempdb..#Temp\_Unsecured') IS NOT NULL -- Temp Table that contains all secured loans

DROP TABLE #Temp\_Unsecured

SELECT CREDIT\_LINE

into #Temp\_Unsecured

FROM (

select lim.credit\_line

,CASE WHEN LIM.FIXED\_Variable = @FixedVariable

AND COL.Collateral\_Type not in (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Collateral\_Type is not NULL

THEN 'N'

ELSE 'Y'

END AS [Unsecured]

from src.FBNK\_LIMIT LIM

INNER JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.id

AND LIM.collateral\_code = RIG.collateral\_code

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

INNER JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND COL.ASOFDATE = RIG.ASOFDATE

WHERE RECORD\_PARENT IS NULL

--ORDER BY CREDIT\_LINE

)SUB

GROUP BY SUB.CREDIT\_LINE

HAVING MIN([Unsecured]) = 'Y'

--Insert scripts for target table

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,customerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ BAL.ID+' | '+BAL.DATE

AS [SrcID] -- Represents Principal amounts

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(LD.currency,(BAL.PRINCIPAL), @AsofDate) AS [MaturityAmount] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101))BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), '> 5')

END

END

END AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,fmd.desc1 AS [DESC1]

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(BAL.DATE as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS orig\_marketValue -- Reasonableness Test

,BAL.PRINCIPAL AS orig\_maturityValue -- Reasonableness Test

,LD.currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN src.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID+'.00'+LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN #Temp\_Unsecured TMP -- Join to eliminate all secured loans

ON TMP.CREDIT\_LINE = LIM.CREDIT\_LINE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = LD.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LD.CUSTOMER\_ID

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(BAL.DATE as DATE),101) > @AsofDate

AND BAL.PRINCIPAL <> 0

AND BAL.PRINCIPAL IS NOT NULL

AND BAL.ASOFDATE = @ASOFDATE

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND ((RIGHT(LEFT(SUBSTRING(lim.CREDIT\_LINE\_NO,CHARINDEX('.', lim.CREDIT\_LINE\_NO), LEN(lim.CREDIT\_LINE\_NO)),4),3) =@revolving)

AND LD.Category <> @SyndicatedLoanCategory)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.U.5 - Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.5'

SET @PID1 = 'I.S.7' -- To identofy secured loans and eliminate them

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @CollateralCode = 'CollateralCode' -- Collateral\_Type = 501, 502, 503, 504, 600, 700 or 800

SET @FixedVariable = (SELECT DISTINCT Filter -- Fixed variable = 'FIXED'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter -- Syndicated Loan Category = 21067

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

SET @TradeFinanceCategory = 'CATEGORY'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @SCHTYPE = 'SCHTYPE'

SET @Revolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @DESC1 = 'DESC1'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

IF OBJECT\_ID('tempdb..#Temp\_Unsecured1') IS NOT NULL -- Temp Table that contains all secured loans

DROP TABLE #Temp\_Unsecured1

SELECT CREDIT\_LINE

into #Temp\_Unsecured1

FROM (

select lim.credit\_line

,CASE WHEN LIM.FIXED\_Variable = @FixedVariable

AND COL.Collateral\_Type not in (SELECT DISTINCT Filter -- Fixed variable = 'FIXED'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Collateral\_Type is not NULL

THEN 'N'

ELSE 'Y'

END AS [Unsecured]

from src.FBNK\_LIMIT LIM

INNER JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.id

AND LIM.collateral\_code = RIG.collateral\_code

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

INNER JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND COL.ASOFDATE = RIG.ASOFDATE

WHERE RECORD\_PARENT IS NULL

--ORDER BY CREDIT\_LINE

)SUB

GROUP BY SUB.CREDIT\_LINE

HAVING MIN([Unsecured]) = 'Y'

--Insert scripts for target table

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,customerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | INTEREST | '+ BAL.ID+' | '+BAL.DATE

AS [SrcID] -- Represents Interest

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(LD.currency,(BAL.INTEREST), @AsofDate) AS [MaturityAmount] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101))BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), '> 5')

END

END

END AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,fmd.desc1 AS [DESC1]

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer]

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType] -- Reasonableness Test

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(BAL.DATE as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS orig\_marketValue -- Reasonableness Test

,BAL.INTEREST AS orig\_maturityValue -- Reasonableness Test

,LD.currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN src.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID+'.00'+LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN #Temp\_Unsecured TMP -- To eliminate secured loans

ON TMP.CREDIT\_LINE = LIM.CREDIT\_LINE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = LD.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LD.CUSTOMER\_ID

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(BAL.DATE as DATE),101) > @AsofDate

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

AND BAL.ASOFDATE = @ASOFDATE

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND ((RIGHT(LEFT(SUBSTRING(lim.CREDIT\_LINE\_NO,CHARINDEX('.', lim.CREDIT\_LINE\_NO), LEN(lim.CREDIT\_LINE\_NO)),4),3) =@revolving)

AND LD.Category <> @SyndicatedLoanCategory)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.U.6 - Loan Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.6'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter -- syndicated loan category = 21067

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

SET @TradeFinanceCategory = 'CATEGORY'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Category = 'CATEGORY'

SET @Revolving = (SELECT DISTINCT Filter -- revolving = 001

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @NonRevolving = (SELECT DISTINCT Filter -- non - revolving = 002

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving'

)

SET @SchType = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SchType'

AND Module = 'Principal'

)

SET @FixedVariable = (SELECT DISTINCT Filter -- Fixed Variable = FIXED

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @CollateralCode = 'CollateralCode'

--Insert scripts for target table

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

IF OBJECT\_ID('tempdb..#Temp\_Unsecured2') IS NOT NULL -- Temp table that contains all secured loans

DROP TABLE #Temp\_Unsecured2

SELECT DISTINCT LD.ID

INTO #Temp\_Unsecured2

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN SRC.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID + '.00' + LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = --LIM.COLLATERAL\_CODE

(SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7)+'.'+substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1) = RIG.ID

AND COL.ASOFDATE = RIG.ASOFDATE

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = LD.Co\_Code

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = LD.CUSTOMER\_ID

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) > @AsofDate

and bal.ASOFDATE = @asofdate

AND BAL.PRINCIPAL <> 0

AND BAL.PRINCIPAL IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

and lim.MAXIMUM\_TOTAL <> 0

AND RIG.ID IS NOT NULL

AND LIM.Fixed\_Variable = @FixedVariable

AND Col.Collateral\_Type NOT IN

(SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Col.Collateral\_Type is not null

--To exclude all unsecured loans

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @NonRevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,CustomerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[Event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ BAL.ID+' | '+BAL.DATE AS [SrcID] -- Represents Principal amounts

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(LD.currency,(BAL.principal), @AsofDate) AS [MaturityAmount] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), '> 5')

END

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,fmd.desc1 AS [DESC1]

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(BAL.DATE as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @asofdate)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS orig\_marketValue -- Reasonableness Test

,BAL.PRINCIPAL AS orig\_maturityValue -- Reasonableness Test

,LD.currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN src.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID+'.00'+LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

left JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = --LIM.COLLATERAL\_CODE

(SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL

AND LIMIT.ASOFDATE = @ASOFDATE)

AND RIG.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = LD.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LD.CUSTOMER\_ID

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(BAL.DATE as DATE),101) > @AsofDate

AND BAL.ASOFDATE = @AsOfDate

AND BAL.PRINCIPAL <> 0

AND BAL.PRINCIPAL IS NOT NULL

--AND TMP.CREDIT\_LINE IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Nonrevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

AND LD.ID NOT IN (SELECT DISTINCT ID FROM #Temp\_Unsecured2)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.U.6 - loan Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.6'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter -- Syndicated loan = 21067

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

SET @TradeFinanceCategory = 'CATEGORY'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Category = 'CATEGORY'

SET @Revolving = (SELECT DISTINCT Filter -- Revolving = '001'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @NonRevolving = (SELECT DISTINCT Filter -- Non Revolving = '002'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving'

)

SET @SchType = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SchType'

AND Module = 'Principal'

)

SET @FixedVariable = (SELECT DISTINCT Filter -- Fixed Variable = FIXED

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @CollateralCode = 'CollateralCode'

--Insert scripts for target table

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

IF OBJECT\_ID('tempdb..#Temp\_Unsecured4') IS NOT NULL

DROP TABLE #Temp\_Unsecured4

SELECT DISTINCT LD.ID

INTO #Temp\_Unsecured4

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN SRC.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID + '.00' + LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = --LIM.COLLATERAL\_CODE

(SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7)+'.'+substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1) = RIG.ID

AND COL.ASOFDATE = RIG.ASOFDATE

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = LD.Co\_Code

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = LD.CUSTOMER\_ID

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) > @AsofDate

and bal.ASOFDATE = @asofdate

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

and lim.MAXIMUM\_TOTAL <> 0

AND RIG.ID IS NOT NULL

AND LIM.Fixed\_Variable = @FixedVariable

AND Col.Collateral\_Type NOT IN

(SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Col.Collateral\_Type is not null

--To exclude all unsecured loans

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @NonRevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[Portfolio]

,maturityDate

,CustomerNumber

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[Event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | INTEREST | '+ BAL.ID+' | '+BAL.DATE AS [SrcID] -- Represents Interest

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(LD.currency,(BAL.INTEREST), @AsofDate) AS [MaturityAmount] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101), '> 5')

END

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,NULL AS [DESC1]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(BAL.DATE as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @asofdate)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS orig\_marketValue -- Reasonableness Test

,BAL.PRINCIPAL AS orig\_maturityValue -- Reasonableness Test

,LD.currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN src.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID+'.00'+LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

left JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = --LIM.COLLATERAL\_CODE

(SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL

AND LIMIT.ASOFDATE = @ASOFDATE)

AND RIG.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = LD.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LD.CUSTOMER\_ID

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(BAL.DATE as DATE),101) > @AsofDate

AND BAL.ASOFDATE = @AsOfDate

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

--AND TMP.CREDIT\_LINE IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Nonrevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

AND LD.ID NOT IN (SELECT DISTINCT ID FROM #Temp\_Unsecured4)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Trade Finance

-- \*\* PID: TSD - I.U.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.U.6'

SET @Portfolio = 'Trade Finance'

SET @Sector = 'Sector'

SET @DESC1 = 'DESC1'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

AND Module = 'I.U.6 - TSD'

)

SET @AsofDate\_1 = @AsofDate

SET @Module = 'I.U.6 - TSD'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStart]

,[forwardEnd]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,[AsofDate]

,[Event]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [MarketValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.DEAL\_MAT\_DATE) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.DEAL\_MAT\_DATE, 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.DEAL\_MAT\_DATE, '> 5')

END

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [Sector]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN FMD.DEAL\_BALANCE = 0

THEN 'Deal\_Balance = 0 identified'

ELSE '' END )

) AS [EVENT]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNO -- Reasonableness Test

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.[t\_fmddata] fmd

LEFT JOIN SRC.COMPANY COMP

ON COMP.COMPANY\_CODE = FMD.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.sector

AND ref\_Cp.ReportingEntity = Ref\_en.ReportingEntity

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = FMD.Customer\_No

AND CUST.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = fmd.CUSTOMER\_NO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND fmd.[deal\_LOCAL\_bal] IS NOT NULL

AND FMD.[deal\_LOCAL\_bal] <> 0

AND FMD.ASOFDATE = @asofdate

AND fmd.DESC1 IN (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND FilterName = @DESC1

AND Module = @module

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* NON BRANCH

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: CASH

-- \*\* PID: I.U.3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.3'

SET @Portfolio = 'CASH'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStartValue]

,[ForwardStartBucket]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

--,[DESC1]

,[CompanyName]

,[Portfolio]

,maturityDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,[AsofDate]

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+ CAST(NB\_IU.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,CASE WHEN ISNUMERIC(NB\_IU.CounterpartyTypeCode) = 1 THEN NB\_IU.CounterpartyTypeCode

else 0 end AS [CID] -- CID exception Logic

,NB\_IU.CounterpartyTypeName AS [CounterParty]

,DBO.fn\_CurrencyExchange

(NB\_IU.OriginalCurrencyOfMaturityAmount,(NB\_IU.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IU.MaturityDate, 'Open')AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,ref\_en.Company\_name AS [CompanyName]

,@Portfolio AS Portfolio

,NB\_IU.MaturityDate AS [MaturityDate]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN NB\_IU.reportingentity IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_IU.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN ISNUMERIC(NB\_IU.CounterpartyTypeCode) <> 1

THEN 'data type mismatch in CID |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_IU.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_IU.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM src.US\_FR2052A\_NB\_IU NB\_IU

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IU.Reportingentity

AND NB\_IU.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_IU.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_IU.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_IU.MaturityAmountOriginalCurrency <> 0

AND NB\_IU.Asofdate = @ASOFDATE

AND 'I.U.'+CAST(NB\_IU.PID\_ProductID AS VARCHAR) = @PID

and NB\_IU.PortfolioName = @Portfolio

AND nb\_iu.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Receivable

-- \*\* PID: I.U.3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.3'

SET @Portfolio = 'Other Receivable'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

Set @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

-- CID commented out since the value is varchar in the table

--,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStart]

,[forwardEnd]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[Portfolio]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,[DESC1]

,companyname

,[ReportType]

,[AsofDate]

,[Event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+CAST(NB\_IU.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

-- CID commented out since the value is varchar in the table

-- ,NB\_IU.CounterpartyTypeCode AS [CID]

,NB\_IU.CounterpartyTypeName AS [CounterParty]

,DBO.fn\_CurrencyExchange

(NB\_IU.OriginalCurrencyOfMaturityAmount,(NB\_IU.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IU.MaturityDate, 'Open')

AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,'N/A' AS [reasonableness\_Customer]-- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN NB\_IU.reportingentity IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_Contract\_DealNo]

,NB\_IU.MaturityDate AS [Maturity Date]

,'N/A' AS [DESC1]

,ref\_en.company\_name AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_IU.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_IU.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_IU.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_IU NB\_IU

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IU.Reportingentity

AND NB\_IU.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_IU.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_IU.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_IU.MaturityAmountOriginalCurrency <> 0

AND NB\_IU.Asofdate = @ASOFDATE

AND 'I.U.'+CAST(NB\_IU.PID\_ProductID AS VARCHAR) = @PID

and NB\_IU.PortfolioName = @Portfolio+ 's and deposit'

AND nb\_iu.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: I.U.6 - no data since all the data in the table is as of 12/31/2016

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.U.6'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

Set @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_unsecuredInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

-- CID commented out since the value is varchar in the table

--,[CID]

,[counterPartySector]

,[maturityValue]

,[maturityBucket]

,[forwardStart]

,[forwardEnd]

,[internal]

,[internalCounterParty]

,[primeBrokerage]

,[Portfolio]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[ReportType]

,CompanyName

,[AsofDate]

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+CAST(NB\_IU.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

-- CID commented out since the value is varchar in the table

--,NB\_IU.CounterpartyTypeCode AS [CID]

,NB\_IU.CounterpartyTypeName AS [CounterParty]

,DBO.fn\_CurrencyExchange

(NB\_IU.OriginalCurrencyOfMaturityAmount,(NB\_IU.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IU.MaturitydATE) <= 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IU.MaturitydATE, 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IU.MaturitydATE, '> 5')

END

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN NB\_IU.reportingentity IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_Contract\_DealNo] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,ref\_en.Company\_Name AS CompanyName

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_IU.MaturitydATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_IU.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IU.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_IU.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_IU.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,NB\_IU.MaturityDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_IU NB\_IU

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IU.Reportingentity

AND NB\_IU.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_IU.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_IU.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_IU.MaturityAmountOriginalCurrency <> 0

AND NB\_IU.Asofdate = @ASOFDATE

AND 'I.U.'+CAST(NB\_IU.PID\_ProductID AS VARCHAR) = @PID

and NB\_IU.PortfolioName = @Portfolio

AND nb\_iu.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_secured\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_secured\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_secured\_outflow]

\*\* Report Type: Secured Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_securedOutflow') AND NAME ='IX\_Tgt\_securedOutflow')

DROP INDEX IX\_Tgt\_securedOutflow ON app.Tgt\_securedOutflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_securedOutflow

ON app.Tgt\_securedOutflow(Tgt\_securedOutflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate\_1 Date

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR(10)

DECLARE @Settlement VARCHar(20)

DECLARE @Rehypothecated VARCHAR(20)

DECLARE @BR VARCHAR(5)

DECLARE @BR1 VARCHAR(5)

DECLARE @PS VARCHAR(5)

DECLARE @REPORTENTITY VARCHAR(50)

DECLARE @REVREASON VARCHAR (5)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: O.S.1-- TreasuryControl = 'N'

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.S.1'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter -- TreasuryControl = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @AsofDate\_1 = @AsofDate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Settlement = (SELECT DISTINCT Filter -- Settlement = 'Bilateral'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement'

)

SET @Rehypothecated = (SELECT DISTINCT Filter -- Rehypothecated = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Rehypothecated'

)

SET @BR = 'BR'

SET @PS = 'PS'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,SubProduct

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,TreasuryControl

,internal

,internalCounterParty

,primeBrokerage

,settlementMechanism

,Rehypothecated

,CID

,counterPartySector

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'RPDT | '+RPDT.DEALNO+ ' | '+RPDT.BR + ' | '+CAST(RPDT.DEALTIME AS VARCHAR(50))+' | '+CAST(RPDT.YIELD\_8 AS VARCHAR(100))+' | '+CAST(RPDT.CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(RPDT.DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100)) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN RPDT.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,DBO.fn\_CurrencyExchange

(RPDT.CCY,(RPDT.MatProcAmt), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.MDATE, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

-- Report PROCEEDAMT only if VDATE > reporting date

,DBO.fn\_CurrencyExchange( RPDT.CCY,(RPDT.ProceedAmt), @AsofDate)AS [ForwardStart] -- Calcuation performed with Foreign Currency Exchange function

--Report VDATE only if VDATE>reporting date

,CASE WHEN RPDT.VDate>@AsofDate

THEN (CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.VDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.VDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.VDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.VDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,RPDT.VDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,RPDT.VDATE, '> 1')

END

END

END

)

ELSE NULL

END AS [ForwardBucket] -- Calculate maturity bucket using maturity bucket function

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass] -- Collateral Class To be sourced from SECM.DESR for values described in the DAST

,RPDT.QTY\*DBO.fn\_CurrencyExchange(RPDT.CCY,(SECL.CLSGPrice\_8),@AsofDate)

AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Settlement AS [Settlement]

,CASE WHEN @Rehypothecated = 'y' THEN 'True'

ELSE 'False' END AS [Rehypothecated]

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN RPDT.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN RPDT.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN RPDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN RPDT.CCY IS NULL OR RPDT.CCY = '' OR RPDT.CCY IS NULL OR RPDT.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN RPDT.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

OR RPDT.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,CUST.CNO AS customerNumber

,NULL AS orig\_marketValue

,RPDT.MatProcAmt AS orig\_maturityValue

,RPDT.CCY AS orig\_currency

,RPDT.MDATE AS maturitydate

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_RPDT RPDT

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN SRC.US\_OPICS\_CUST CUST

ON RPDT.CNO = CUST.CNO

AND CUST.ASOFDATE = RPDT.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECL SECL

ON RPDT.SECID = SECL.SECID

AND SECL.ASOFDATE = RPDT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = RPDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON SECM.SECID = RPDT.SECID

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = RPDT.CNO

-- To avoid FR2052a Validation fail - where Reporting Entity is not the same as internal counterparty

WHERE ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND RPDT.PS IN (SELECT DISTINCT Filter -- PS = 'PS'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @PS

)

AND RPDT.BR IN (SELECT DISTINCT Filter -- BR = '01' or '02'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

AND RPDT.ASOFDATE = @AsofDate

AND RPDT.MDATE > @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Treasury Investment

-- \*\* PID: O.S.8

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.S.8'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter -- TreasuryControl = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Settlement = (SELECT DISTINCT Filter -- Settlement = 'Triparty'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement'

)

SET @Rehypothecated = (SELECT DISTINCT Filter -- Rehypothecated = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Rehypothecated'

)

SET @BR = 'BR'

SET @PS = 'PS'

SET @AsofDate\_1 = @ASOFDATE

SET @REVREASON = (SELECT DISTINCT FILTER -- Revreason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = 'Revreason')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,SubProduct

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,TreasuryControl

,internal

,internalCounterParty

,primeBrokerage

,settlementMechanism

,Rehypothecated

,CID

,counterPartySector

,Portfolio

,CompanyName

,reportType

,AsofDate

,maturityDate

,[event]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

SPSH.BR+' - '+SPSH.DEALNO AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SPSH.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, ref\_rep.Product AS Product

,ref\_rep.[SID] AS [SID]

,ref\_rep.SubProduct AS [SubProduct]

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,CASE WHEN SPSH.CCY = 'USD' THEN (SPSH.SETTAMT)

ELSE (DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsofDate)) END AS [ForwardStart] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,SPSH.SETTDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SPSH.SETTDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SPSH.SETTDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SPSH.SETTDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SPSH.SETTDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SPSH.SETTDATE, '1-2')

END

END

END AS [ForwardBucket] -- Calculate maturity bucket using maturity bucket function

-- Collateral Class To be sourced from SECM.DESR for values described in the DAST

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass]

,DBO.fn\_CurrencyExchange

(SPSH.CCY ,(SPSH.SETTAMT),@AsofDate) AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Settlement AS [Settlement]

,CASE WHEN @Rehypothecated = 'y' THEN 'True'

ELSE 'False' END AS [Rehypothecated]

,ref\_cp.CID AS [CID]

,ref\_cp.CounterParty AS [CounterParty]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,SECM.MDATE AS maturityDate

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY IS NULL OR SPSH.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN SPSH.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN SPSH.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SPSH.SETTAMT AS orig\_maturityValue -- Reasonableness Test

,SPSH.CCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_SPSH SPSH

INNER JOIN SRC.US\_OPICS\_SECM SECM

ON SPSH.SECID = SECM.SECID

AND SPSH.ASOFDATE = SECM.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_CUST CUST

ON CUST.CNO = SECM.ISSUER

AND CUST.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = SPSH.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_CP

ON REF\_CP.Ctype = CUST.CTYPE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SPSH.CNO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND SPSH.PS = (SELECT FILTER --Purchase/Sale Indicator SET TO P

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @PS)

AND SPSH.BR IN(SELECT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR)

AND SPSH.SETTDATE > @AsofDate -- Current Reporting date'12/30/2016'

AND SPSH.REVREASON = @REVREASON

AND SPSH.ASOFDATE = @ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Non - Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: O.S.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.S.9'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @CURRENCY = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter -- TreasuryControl = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Settlement = (SELECT DISTINCT Filter -- Settlement = 'Bilateral'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement'

)

SET @Rehypothecated = (SELECT DISTINCT Filter -- Rehypothecated = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Rehypothecated'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,SubProduct

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,TreasuryControl

,internal

,internalCounterParty

,primeBrokerage

,settlementMechanism

,Rehypothecated

,CID

,counterPartySector

,Portfolio

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,CompanyName

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'NB\_OS | '+CAST(NB\_OS.ID AS VARCHAR) + ' | O.S.'+ CAST(NB\_OS.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OS.ReportingEntity +' | '+CAST(NB\_OS.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_OS.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,dbo.fn\_CurrencyExchange

(NB\_OS.OriginalCurrencyOfMaturityAmount,(NB\_OS.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OS.MaturityDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OS.MaturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OS.MaturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OS.MaturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OS.MaturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OS.MaturityDate, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,dbo.fn\_CurrencyExchange

(NB\_OS.OriginalCurrencyOfCollateralValue,NB\_OS.CollateralValueOriginalCurrency, @AsofDate)

AS [ColletaralValue]

,NB\_OS.OriginalCurrencyOfCollateralValue AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Settlement AS [Settlement]

,CASE WHEN @Rehypothecated = 'y' THEN 'True'

ELSE 'False' END AS [Rehypothecated]

-- To be uncommented after data is cleaned for CID

,NULL

--,NB\_OS.CounterpartyTypeCode AS [CID]

,NB\_OS.CounterpartyTypeName AS [CounterParty]

,@Portfolio AS Portfolio

,'N/A' AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN NB\_OS.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N/A' AS reasonableness\_Contract\_DealNo

,ref\_rep.ReportType AS [ReportType]

,ref\_en.company\_name AS CompanyName

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_OS.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OS.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_OS.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OS.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_OS.MaturityAmountOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_OS.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,NB\_OS.MaturityDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_OS NB\_OS

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_OS.reportingentity

AND NB\_OS.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_OS.CollateralCode

and ref\_col.reportingentity =NB\_OS.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_OS.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_OS.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_OS.MaturityAmountOriginalCurrency <> 0

AND NB\_OS.ASOFDATE = @ASOFDATE

AND 'O.S.'+CAST(NB\_OS.PID\_ProductID AS VARCHAR) = @PID

and NB\_OS.PortfolioName = @Portfolio

AND nb\_OS.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_secured\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_secured\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_secured\_Inflow]

\*\* Report Type: Secured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_T24\_MULTI\_LD\_BALANCE') AND NAME ='IX\_T24\_BAL')

DROP INDEX IX\_T24\_BAL ON SRC.US\_T24\_MULTI\_LD\_BALANCE;

CREATE NONCLUSTERED INDEX IX\_T24\_BAL

ON src.US\_T24\_MULTI\_LD\_BALANCE(ID,CO\_CODE,ASOFDATE);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.MULTI\_LD') AND NAME ='IX\_MULTI\_LD')

DROP INDEX IX\_MULTI\_LD ON SRC.MULTI\_LD ;

CREATE NONCLUSTERED INDEX IX\_MULTI\_LD

ON src.MULTI\_LD(ID,CO\_CODE,ASOFDATE, CATEGORY);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.TMP\_MG\_Mortgage\_Schedule') AND NAME ='IX\_TMP')

DROP INDEX IX\_TMP ON SRC.TMP\_MG\_Mortgage\_Schedule;

CREATE NONCLUSTERED INDEX IX\_TMP

ON src.TMP\_MG\_Mortgage\_Schedule(CONTRACT\_NO, ASOFDATE);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.Multi\_MG\_MORTGAGE') AND NAME ='IX\_MULTI\_MG')

DROP INDEX IX\_MULTI\_MG ON SRC.Multi\_MG\_MORTGAGE;

CREATE NONCLUSTERED INDEX IX\_MULTI\_MG

ON src.Multi\_MG\_MORTGAGE(ID, CATEGORY, CO\_CODE, ASOFDATE);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR(10)

DECLARE @Settlement VARCHar(20)

DECLARE @BR VARCHAR(5)

DECLARE @PS VARCHAR(5)

--The following variables are for Commercial and Retail Loans Portfolio

--declaRe categories to be excluded

DECLARE @TradeFinanceCategory VARCHAR(10) -- These are already captured in Trade finance Portfolio

DECLARE @SyndicatedLoanCategory VARCHAR(10) -- syndicated term loans that need to be excluded

DECLARE @CollateralCode VARCHAR(50)

DECLARE @ReportEntity VARCHAR(50)

DECLARE @SCHTYPE VARCHAR(10)

DECLARE @FixedVariable VARCHAR(10)

DECLARE @Revolving INT

DECLARE @NonRevolving INT

-- Global Variables

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding

-- \*\* PID: I.S.1

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.S.1'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @BR = 'BR'

SET @PS = 'PS'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @PrimeBrokerage = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @Unencumbered = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Unencumbered')

SET @Settlement = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement')

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,[SubProduct]

,maturityValue

,maturityBucket

,EffectiveMaturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,Unencumbered

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,SettlementMechanism

,CID

,counterPartySector

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,CreationDate)

SELECT

'RPDT | '+DEALNO+' | '+ rpdt.BR + ' | '+CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100)) AS SRCID --Create Source ID

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN RPDT.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,DBO.fn\_CurrencyExchange

(RPDT.CCY,(RPDT.MatProcAmt), @AsofDate)

AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.MDATE, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL

,CASE WHEN RPDT.VDate>@AsofDate

THEN

(DBO.fn\_CurrencyExchange

(RPDT.CCY,(RPDT.ProceedAmt), @AsofDate)

)

ELSE NULL

END AS [ForwardStart] -- Report PROCEEDAMT only if VDATE > reporting date

,CASE WHEN RPDT.VDate>@AsofDate

THEN (CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.VDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.VDate, 'Day 60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.VDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.VDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,RPDT.VDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,RPDT.VDate, '> 5')

END

END

END)

ELSE NULL

END AS [ForwardBucket] --Report VDATE only if VDATE>reporting date

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass]

,RPDT.QTY\* DBO.fn\_CurrencyExchange

(RPDT.CCY,(SECL.CLSGPrice\_8), @AsofDate) AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN @Unencumbered = 'Y' THEN 'True'

ELSE 'False' END AS [Unencumbered]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.internal = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.internalcounterparty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Settlement AS [Settlement]

,ref\_cp.CID AS [CID]

,ref\_cp.CounterParty AS [CounterPary]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] --Reasonableness Test

,'N/A' AS [reasonableness\_Sector] --Reasonableness Test

,CASE WHEN RPDT.BR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN RPDT.DEALNO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] --Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

, ((SELECT CASE WHEN RPDT.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN RPDT.CCY IS NULL OR RPDT.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN RPDT.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] --Reasonableness Test

,CUST.CNO AS customerNumber --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,RPDT.MatProcAmt AS orig\_maturityValue --Reasonableness Test

,RPDT.CCY AS orig\_currency --Reasonableness Test

,RPDT.VDate AS maturitydate --Reasonableness Test

,@CreationDate AS CreationDate --Reasonableness Test

FROM SRC.US\_OPICS\_RPDT RPDT

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN SRC.US\_OPICS\_CUST CUST

ON RPDT.CNO = CUST.CNO

AND CUST.ASOFDATE = RPDT.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECL SECL

ON RPDT.SECID = SECL.SECID

AND SECL.ASOFDATE = RPDT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = RPDT.BR

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON SECM.SECID = RPDT.SECID

AND SECM.ASOFDATE = RPDT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = RPDT.CNO

AND ref\_Internal\_cp.InternalCounterParty <> ref\_en.reportingentityDescription

WHERE

RPDT.MDATE > @AsofDate

AND RPDT.ASOFDATE = @ASOFDATE

AND RPDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR)

AND RPDT.PS IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @PS)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.S.7 Loan Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.S.7'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @BR = 'BR'

SET @CollateralCode = 'CollateralCode'

SET @Category = 'Category'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @PrimeBrokerage = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @Unencumbered = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Unencumbered')

SET @Settlement = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement')

SET @FixedVariable = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable')

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory')

SET @SCHTYPE = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'schtype'

AND Module = 'Interest')

SET @Revolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving')

SET @NonRevolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving')

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,[SubProduct]

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,Unencumbered

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,DESC1

,SettlementMechanism

,CID

,counterPartySector

,Portfolio

,maturityDate

,customerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,CompanyName

,reportType

,AsofDate

,[event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ bal.ID+' | '+BAL.[DATE] AS [SrcID] --create Unique Source ID

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,Ref\_Rep.Product AS [Product]

,NULL AS [SID]

,NULL AS [SubProduct]

,(DBO.fn\_CurrencyExchange (LD.currency,(bal.PRINCIPAL), @AsofDate)) AS [MaturityAmount]

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101)) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101), 'Day 30')

ELSE dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101), '> 5')

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE'

THEN NULL

ELSE ref\_col.CollateralClassFR2052A

END AS [CollateralClass]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE'

THEN NULL

ELSE (ISNULL(((DBO.fn\_CurrencyExchange (LD.CURRENCY,bal.PRINCIPAL, @AsofDate)

/NULLIF(DBO.fn\_CurrencyExchange

(LIM.LIMIT\_currency,LIM.MAXIMUM\_TOTAL, @AsofDate),0)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL

THEN 0

ELSE (DBO.fn\_CurrencyExchange(COL.currency,(COL.NOMINAL\_VALUE), @AsofDate))

END ))),0))

END AS [CollateralValue]

,@currency AS [CollateralCurrency]

,CASE WHEN @Unencumbered = 'Y' THEN 'True'

ELSE 'False' END AS [Unencumbered]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.internal = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.internalcounterparty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,FMD.DESC1 AS [DESC1]

,@Settlement AS [Settlement]

,ISNULL((SELECT EXC.CID

FROM app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID]

,ISNULL((SELECT EXC.COunterparty

FROM app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo] --Reasonableness Test

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] --Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] --Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] --Reasonableness Test

,REF\_EN.COMPANY\_NAME AS [CompanyName] --Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIM.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT] --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,bal.PRINCIPAL AS orig\_maturityValue --Reasonableness Test

,LD.currency AS orig\_currency --Reasonableness Test

,@CreationDate AS CreationDate --Reasonableness Test

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN SRC.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID + '.00' + LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = (SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7)+'.'+substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1) = RIG.ID

AND COL.ASOFDATE = RIG.ASOFDATE

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

ON ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = LD.Co\_Code

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = LD.CUSTOMER\_ID

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) > @AsofDate

and bal.ASOFDATE = @asofdate

AND BAL.PRINCIPAL <> 0

AND BAL.PRINCIPAL IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

and lim.MAXIMUM\_TOTAL <> 0

AND RIG.ID IS NOT NULL AND

LIM.Fixed\_Variable = @FixedVariable

AND

Col.Collateral\_Type NOT IN

(SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Col.Collateral\_Type is not null--To exclude all unsecured loans

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @NonRevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

GROUP BY

ref\_en.reportingentityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,lim.CO\_CODE

,lim.limit\_currency

,RIG.ID

,lim.MAXIMUM\_TOTAL

,ld.CURRENCY

,bal.PRINCIPAL

,bal.id

,Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101)

,BAL.[DATE]

,CUST.CUSTOMER\_CODE

--,cust.SECTOR

,LIM.CO\_CODE

,LD.Contract\_no

,LIM.CREDIT\_LINE

,FMD.DESC1

,REF\_EN.COMPANY\_NAME

--,col.COLLATERAL\_TYPE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.S.7 Loan Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.S.7'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @BR = 'BR'

SET @CollateralCode = 'CollateralCode'

SET @Category = 'Category'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @PrimeBrokerage = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @Unencumbered = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Unencumbered')

SET @Settlement = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement')

SET @FixedVariable = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable')

SET @SyndicatedLoanCategory = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory')

SET @SCHTYPE = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'schtype'

AND Module = 'Interest')

SET @Revolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving')

SET @NonRevolving = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving')

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

--Insert scripts for target table

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,[SubProduct]

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,Unencumbered

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,desc1

,SettlementMechanism

,CID

,counterPartySector

,Portfolio

,maturityDate

,[CompanyName]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | INTEREST | '+ bal.ID+' | '+BAL.[DATE] AS [SrcID] --create Unique Source ID

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,NULL AS [SID]

,NULL AS [SubProduct]

,(DBO.fn\_CurrencyExchange (LD.currency,(bal.INTEREST), @AsofDate)) AS [MaturityAmount]

,CASE WHEN DATEDIFF(DAY,@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) ) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) , 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) , '> 5')

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

--Collateral Value is calculated as (INTEREST\_Amount/Maximum\_Total)\*Nominal\_Value

-- To assign collateral proportionately to the available amount

, CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE (ISNULL(((DBO.fn\_CurrencyExchange (LD.CURRENCY,bal.INTEREST, @AsofDate)

/NULLIF(DBO.fn\_CurrencyExchange

(LIM.LIMIT\_currency,LIM.MAXIMUM\_TOTAL, @AsofDate),0)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (DBO.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsofDate))

END ))),0))END AS [CollateralValue]

,@currency AS [CollateralCurrency]

,CASE WHEN @Unencumbered = 'Y' THEN 'True'

ELSE 'False' END AS [Unencumbered]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.internal = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.internalcounterparty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,FMD.DESC1 AS [DESC1]

,@Settlement AS [Settlement]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID]

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,@Portfolio AS [Portfolio]

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101) AS [MaturityDate]

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] --Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] --Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN LD.Contract\_no IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] --Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN Convert(Varchar(20),CAST(BAL.DATE as DATE),101) < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency IS NULL OR LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIM.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT] --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,bal.INTEREST AS orig\_maturityValue --Reasonableness Test

,LD.currency AS orig\_currency --Reasonableness Test

,@CreationDate AS CreationDate --Reasonableness Test

FROM src.US\_T24\_MULTI\_LD\_BALANCE BAL

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

LEFT JOIN src.MULTI\_LD LD

ON BAL.ID = LD.ID

AND BAL.ASOFDATE = LD.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

LEFT JOIN SRC.FBNK\_LIMIT LIM

ON LD.CUSTOMER\_ID + '.00' + LD.LIMIT\_REFERENCE = LIM.ID

AND LIM.ASOFDATE = LD.ASOFDATE

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = LIM.CREDIT\_LINE

AND RIG.collateral\_code = --LIM.COLLATERAL\_CODE

(SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

AND RIG.ASOFDATE = LIM.ASOFDATE

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7)+'.'+substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1) = RIG.ID

AND COL.ASOFDATE = RIG.ASOFDATE

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = LD.Co\_Code

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LD.CUSTOMER\_ID

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = LD.CUSTOMER\_ID

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND Convert(Varchar(20),CAST(BAL.DATE as DATE),101) > @AsofDate

and bal.ASOFDATE = @asofdate

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

AND FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND lim.MAXIMUM\_TOTAL <> 0

AND RIG.ID IS NOT NULL AND

LIM.Fixed\_Variable = @FixedVariable

AND

Col.Collateral\_Type NOT IN

(SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @CollateralCode

)

AND Col.Collateral\_Type is not null

--To exclude all unsecured loans

AND

(

(RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @NonRevolving)

OR

((RIGHT(LEFT(SUBSTRING(LIM.CREDIT\_LINE\_NO,CHARINDEX('.', LIM.CREDIT\_LINE\_NO), LEN(LIM.CREDIT\_LINE\_NO)),4),1) = @Revolving)

AND LD.Category = @SyndicatedLoanCategory

))

GROUP BY

ref\_en.reportingentityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,FMD.DESC1

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,lim.CO\_CODE

,lim.limit\_currency

,RIG.ID

,lim.MAXIMUM\_TOTAL

,ld.CURRENCY

,bal.INTEREST

,bal.id

,Convert(Varchar(20),CAST(BAL.DATE as DATE),101)

,BAL.DATE

,CUST.CUSTOMER\_CODE

--,cust.SECTOR

,LIM.CO\_CODE

,LD.Contract\_no

,LIM.CREDIT\_LINE

,REF\_EN.COMPANY\_NAME

--,col.COLLATERAL\_TYPE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.S.7 Mortgage Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.S.7'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @SyndicatedLoanCategory = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

--SET @TradeFinanceCategory = 'CATEGORY'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @FixedVariable = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @PrimeBrokerage = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Revolving = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @NonRevolving = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving'

)

SET @Unencumbered = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Unencumbered'

)

SET @Settlement = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement'

)

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

SET @Category = 'Category'

SET @CollateralCode = 'CollateralCode'

--Insert scripts for target table

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,[SubProduct]

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,Unencumbered

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,desc1

,SettlementMechanism

,CID

,counterPartySector

,Portfolio

,maturityDate

,customerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,CompanyName

,reportType

,AsofDate

,[Event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate)

SELECT

'TMP\_MG\_Mortgage\_Schedule | Principal | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE AS [SrcID] --create Unique Source ID

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN TMP.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,NULL AS [SID]

,NULL AS [SubProduct]

,DBO.fn\_CurrencyExchange

(TMP.currency,(TMP.PRINCIPAL\_AMOUNT), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,TMP.[DUE\_DATE]) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,TMP.[DUE\_DATE], 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,TMP.[DUE\_DATE], '> 5')

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

--Collateral Value is calculated as (Principal\_Amount/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE (ISNULL(((DBO.fn\_CurrencyExchange (TMP.CURRENCY,TMP.PRINCIPAL\_AMOUNT, @AsofDate)

/NULLIF(DBO.fn\_CurrencyExchange

(LIM.LIMIT\_currency,LIM.MAXIMUM\_TOTAL, @AsofDate),0)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (DBO.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsofDate))

END ))),0)) END AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN @Unencumbered = 'Y' THEN 'True'

ELSE 'False' END AS [Unencumbered]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.internal = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.internalcounterparty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,'N/A' AS [DESC1]

,@Settlement AS [Settlement]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID]

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,@Portfolio AS [Portfolio]

,TMP.[DUE\_DATE] AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] --Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] --Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN TMP.CONTRACT\_NO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] --Reasonableness Test

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN TMP.[DUE\_DATE] < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN TMP.currency IS NULL OR TMP.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TMP.currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIM.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT] --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,TMP.PRINCIPAL\_AMOUNT AS orig\_maturityValue --Reasonableness Test

,TMP.currency AS orig\_currency --Reasonableness Test

,@CreationDate AS CreationDate --Reasonableness Test

FROM

-- Access the linked server

src.TMP\_MG\_Mortgage\_Schedule TMP

INNER JOIN src.Multi\_MG\_MORTGAGE MG

ON TMP.CUSTOMER = MG.CUSTOMER

AND TMP.CONTRACT\_NO = MG.ID

AND TMP.ASOFDATE = MG.ASOFDATE

LEFT JOIN src.FBNK\_LIMIT LIM

ON MG.CUSTOMER+'.00'+MG.LIMIT\_REF = LIM.ID

AND LIM.ASOFDATE = MG.ASOFDATE

LEFT JOIN src.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.LIMIT\_REFERENCE = LIM.CREDIT\_LINE

and rig.ASOFDATE = lim.ASOFDATE

AND RIG.COLLATERAL\_CODE = (SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

------join logic on only first two parts of collateral ID

LEFT JOIN src.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

and col.ASOFDATE = rig.ASOFDATE

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = MG.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = MG.CUSTOMER

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = MG.CUSTOMER

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND tmp.[due\_date] > @AsofDate

AND TMP.ASOFDATE = @ASOFDATE

AND MG.PRINCIPAL\_AMOUNT <> 0

AND MG.PRINCIPAL\_AMOUNT IS NOT NULL

AND LIM.MAXIMUM\_TOTAL <> 0

GROUP BY

ref\_en.reportingentityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,lim.CO\_CODE

,lim.limit\_currency

,RIG.ID

,lim.MAXIMUM\_TOTAL

,RIG.id

,tmp.CONTRACT\_NO

,TMP.CURRENCY

,TMP.PRINCIPAL\_AMOUNT

,TMP.DUE\_DATE

,CUST.CUSTOMER\_CODE

,cust.SECTOR

,LIM.CO\_CODE

,TMP.Contract\_no

,REF\_EN.COMPANY\_NAME

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.S.7 Mortgage Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.S.7'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @SyndicatedLoanCategory = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SyndicatedLoanCategory'

)

--SET @TradeFinanceCategory = 'CATEGORY'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @FixedVariable = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FixedVariable'

)

SET @PrimeBrokerage = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @Revolving = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Revolving'

)

SET @NonRevolving = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'NonRevolving'

)

SET @Unencumbered = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Unencumbered'

)

SET @Settlement = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Settlement'

)

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

SET @Category = 'Category'

SET @CollateralCode ='CollateralCode'

--Insert scripts for target table

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_securedInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[SID]

,[SubProduct]

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,Unencumbered

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,desc1

,SettlementMechanism

,CID

,counterPartySector

,Portfolio

,maturityDate

,CustomerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,CompanyName

,reportType

,AsofDate

,[Event]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'TMP\_MG\_Mortgage\_Schedule | Interest | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE AS [SrcID] --create Unique Source ID

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN TMP.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

--Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,NULL AS [SID]

,NULL AS [SubProduct]

,DBO.fn\_CurrencyExchange

(TMP.currency,(TMP.INTEREST\_AMOUNT), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,TMP.[DUE\_DATE]) BETWEEN 1 AND 30

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,TMP.[DUE\_DATE], 'Day 30')

ELSE

dbo.fn\_MaturityDateCalculater(@AsofDate,TMP.[DUE\_DATE], '> 5')

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

--Collateral Value is calculated as (Principal\_Amount/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,CASE WHEN ref\_col.CollateralClassFR2052A = 'UNAVAILABLE' THEN NULL

ELSE (ISNULL(((DBO.fn\_CurrencyExchange (TMP.CURRENCY,TMP.INTEREST\_AMOUNT, @AsofDate)

/NULLIF(DBO.fn\_CurrencyExchange

(LIM.LIMIT\_currency,LIM.MAXIMUM\_TOTAL, @AsofDate),0)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (DBO.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsofDate))

END ))),0)) END AS [CollateralValue]

,@currency AS [CollateralCurrency]

,CASE WHEN @Unencumbered = 'Y' THEN 'True'

ELSE 'False' END AS [Unencumbered]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.internal = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.internalcounterparty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,'N/A' AS [DESC1]

,@Settlement AS [Settlement]

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID]

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

,@Portfolio AS [Portfolio]

,TMP.[DUE\_DATE] AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Customer] --Reasonableness Test

,CASE WHEN cust.SECTOR IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Sector] --Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN TMP.CONTRACT\_NO IS NULL THEN 'N' ELSE 'Y' END AS [reasonableness\_Contract\_DealNo] --Reasonableness Test

,REF\_EN.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN TMP.[DUE\_DATE] < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN TMP.currency IS NULL OR TMP.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TMP.currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIM.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT] --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,TMP.INTEREST\_AMOUNT AS orig\_maturityValue --Reasonableness Test

,TMP.currency AS orig\_currency --Reasonableness Test

,@CreationDate AS CreationDate --Reasonableness Test

FROM

-- Access the linked server

src.TMP\_MG\_Mortgage\_Schedule TMP

INNER JOIN src.Multi\_MG\_MORTGAGE MG

ON TMP.CUSTOMER = MG.CUSTOMER

AND TMP.CONTRACT\_NO = MG.ID

AND TMP.ASOFDATE = MG.ASOFDATE

LEFT JOIN src.FBNK\_LIMIT LIM

ON MG.CUSTOMER+'.00'+MG.LIMIT\_REF = LIM.ID

AND LIM.ASOFDATE = MG.ASOFDATE

LEFT JOIN src.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.LIMIT\_REFERENCE = LIM.CREDIT\_LINE

and rig.ASOFDATE = lim.ASOFDATE

AND RIG.COLLATERAL\_CODE = (SELECT DISTINCT COLLATERAL\_CODE

from SRC.FBNK\_LIMIT LIMIT

WHERE LIMIT.CREDIT\_LINE = RIG.LIMIT\_ID

and limit.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

AND LIMIT.ASOFDATE = RIG.ASOFDATE

AND LIMIT.RECORD\_PARENT IS NULL)

------join logic on only first two parts of collateral ID

LEFT JOIN src.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

and col.ASOFDATE = rig.ASOFDATE

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

and col.COLLATERAL\_CODE = rig.COLLATERAL\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_Code = MG.Co\_Code

LEFT JOIN src.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = MG.CUSTOMER

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CUST.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = MG.CUSTOMER

--where Reporting Entity is not the same as internal counterparty

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND tmp.[due\_date] > @AsofDate

AND TMP.ASOFDATE = @ASOFDATE

AND MG.PRINCIPAL\_AMOUNT <> 0

AND MG.PRINCIPAL\_AMOUNT IS NOT NULL

AND LIM.MAXIMUM\_TOTAL <> 0

GROUP BY

ref\_en.reportingentityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,lim.CO\_CODE

,lim.limit\_currency

,RIG.ID

,lim.MAXIMUM\_TOTAL

,RIG.id

,tmp.CONTRACT\_NO

,TMP.CURRENCY

,TMP.INTEREST\_AMOUNT

,TMP.DUE\_DATE

,CUST.CUSTOMER\_CODE

,cust.SECTOR

,LIM.CO\_CODE

,TMP.Contract\_no

,REF\_EN.COMPANY\_NAME

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_deposit\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_deposit\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_deposit\_outflow]

\*\* Report Type: Deposit Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_depositsOutflow') AND NAME ='IX\_Tgt\_depositsOutflow')

DROP INDEX IX\_Tgt\_depositsOutflow ON app.Tgt\_depositsOutflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_depositsOutflow

ON app.Tgt\_depositsOutflow(Tgt\_depositsOutflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR (100)

DECLARE @PID1 VARCHAR (100)

DECLARE @PID2 VARCHAR (10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR (100)

DECLARE @DESC1 VARCHAR (100)

DECLARE @Sector VARCHAR (10)

DECLARE @Currency VARCHAR (3)

DECLARE @MaturityBucket VARCHAR (10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR (10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR (3)

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR (10)

DECLARE @Settlement VARCHAR (20)

DECLARE @CompanyName VARCHAR (50)

DECLARE @CompanyName1 VARCHAR (50)

DECLARE @CounterParty1 VARCHAR (50)

DECLARE @CounterParty2 VARCHAR (50)

DECLARE @CounterParty3 VARCHAR (50)

DECLARE @OD1\_Filter VARCHAR (50)

DECLARE @Trigger VARCHAR (1)

DECLARE @CID VARCHAR (20)

DECLARE @CID2 VARCHAR (100)

DECLARE @FDIC\_PID VARCHAR (20)

DECLARE @wholeSale VARCHAR (100)

DECLARE @Company\_Code VARCHAR (100)

DECLARE @FilterName VARCHAR (150)

DECLARE @FilterName2 VARCHAR (50)

DECLARE @CID\_VALUE1 VARCHAR (2)

DECLARE @CID\_VALUE2 VARCHAR (2)

DECLARE @CID\_VALUE3 VARCHAR (2)

DECLARE @PID\_OD10 VARCHAR(50)

DECLARE @wholeSale\_OD10 VARCHAR (550)

DECLARE @AL VARCHAR(5)

DECLARE @BR VARCHAR(5)

DECLARE @FILTER\_FDIC VARCHAR (20)

DECLARE @FDIC VARCHAR (20)

DECLARE @FDIC\_Threshhold DECIMAL (30,15)

DECLARE @PID\_ALL VARCHAR (20)

DECLARE @FilterName3 VARCHAR (50)

DECLARE @Filter\_Uninsured VARCHAR (100)

DECLARE @OD2\_OR\_OD3 VARCHAR (100)

DECLARE @Filter\_CID1 VARCHAR (100)

DECLARE @Filter\_CID2 VARCHAR (100)

DECLARE @Filter\_CID3 VARCHAR (100)

DECLARE @OD4\_OD5\_Ratio VARCHAR (100)

DECLARE @OD4\_OD5 VARCHAR (100)

DECLARE @FILTER\_INTEREST VARCHAR (100)

DECLARE @FILTER\_INVTYPE VARCHAR (100)

DECLARE @FILTER\_ACCTNGTYPE VARCHAR(100)

DECLARE @PRODUCT VARCHAR (100)

DECLARE @rehypothecated VARCHAR (100)

DECLARE @DESC1\_Exclude VARCHAR (100)

DECLARE @FDIC\_INSURED VARCHAR (50)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

-- Variables are used in all PIDs

SET @PID\_ALL = 'O.D.1-13'

SET @FDIC\_Threshhold = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID\_ALL

AND FilterName = 'FDIC\_Threshold')

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- --TRUNCATE TABLE app.Tgt\_depositsOutflow

---- SET VARIABLES

SET @PID = 'O.D.1'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @OD1\_Filter = 'OD1\_Check'

SET @trigger = 'N'

SET @CID = 'CID'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @Filter\_Uninsured = 'Uninsured'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail')

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterParty]

,(dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,comp.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT] -- Reasonableness Test

,fmd.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.deal\_balance AS orig\_maturityValue -- Reasonableness Test

,fmd.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM src.T\_FMDDATA fmd

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL (ref\_Internal\_cp.InternalCounterParty,'')<> ref\_en.reportingentityDescription

AND ref\_cp.CounterParty IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID

)

-- Retail or Small business

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_ds.OD1\_Check IN (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @OD1\_Filter

AND PID = @PID ) -- DESC1 values identified in app.US\_FR2052A\_REF\_DepositScope

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- --SET VARIABLES

SET @PID = 'O.D.2'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @DESC1\_Exclude = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC1') -- DESC1 = 4960

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @trigger = 'N'

SET @CID = 'CID'

SET @PID1 = 'OD1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @FDIC\_PID = 'O.D.1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @OD2\_OR\_OD3 = 'OD2\_OR\_OD3'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END AS [maturityValue] -- Calculate maturity bucket using maturity bucket function

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket] -- Calcuation performed with Foreign Currency Exchange function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.deal\_balance AS orig\_maturityValue -- Reasonableness Test

,fmd.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE fmd.Customer\_no IN

(

SELECT FMD2.CUSTOMER\_NO

FROM SRC.T\_FMDDATA FMD2

WHERE ISNULL(FMD2.DEAL\_LOCAL\_BAL, 0) <> 0 AND FMD2.ASOFDATE = @AsofDate AND FMD2.CUSTOMER\_NO IS NOT NULL

GROUP BY FMD2.CUSTOMER\_NO

HAVING COUNT(DISTINCT FMD2.CONTACT\_ID) > 1

)

AND ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <>0

AND ref\_cp.CounterParty IN (SELECT DISTINCT Filter -- Retail or Small business

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName

AND PID = @PID1)

AND ref\_ds.OD1\_Check NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @OD1\_Filter

AND PID = @PID2 ) -- Exclude Values from OD1

AND ((ref\_ds.OD1\_2\_Check = @OD2\_OR\_OD3 AND ref\_ds.DESC1 <> @DESC1\_Exclude) OR (FMD.DESC1 = @DESC1\_Exclude AND DEAL\_BALANCE > 0)) -- DESC1 values identified in app.US\_FR2052A\_REF\_DepositScope. Exclude DESC1 4960 if value is negative

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'O.D.3'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @DESC1\_Exclude = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC1') -- DESC1 = 4960

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @OD2\_OR\_OD3 = 'OD2\_OR\_OD3'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue] -- Calculate maturity bucket using maturity bucket function

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket] -- Calcuation performed with Foreign Currency Exchange function

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.deal\_balance AS orig\_maturityValue -- Reasonableness Test

,fmd.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE fmd.Customer\_no NOT IN

(

SELECT FMD2.CUSTOMER\_NO

FROM SRC.T\_FMDDATA FMD2

WHERE ISNULL(FMD2.DEAL\_LOCAL\_BAL, 0) <> 0 AND FMD2.ASOFDATE = @AsofDate AND FMD2.CUSTOMER\_NO IS NOT NULL

GROUP BY FMD2.CUSTOMER\_NO

HAVING COUNT(DISTINCT FMD2.CONTACT\_ID) > 1

)

AND ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2) -- Retail or Small business

AND ref\_ds.OD1\_Check NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @OD1\_Filter

AND PID = @PID2) -- Exclude Values from O

AND ((ref\_ds.OD1\_2\_Check = @OD2\_OR\_OD3 AND ref\_ds.DESC1 <> @DESC1\_Exclude) OR (FMD.DESC1 = @DESC1\_Exclude AND DEAL\_BALANCE > 0)) -- DESC1 values identified in app.US\_FR2052A\_REF\_DepositScope. Exclude DESC1 4960 if value is negative

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.4

-- \*\* OD4 and OD10 are combined as OD4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID\_OD10 = 'O.D.10'

SET @Product ='Operational Accounts'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @wholeSale\_OD10 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD10\_Filter'

AND PID = 'O.D.10')

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @PID = 'O.D.4'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @DESC1\_Exclude = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC1') -- DESC1 = 4960

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

---- FDIC Variables

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @OD4\_OD5\_Ratio = 'OD4\_OD5\_Ratio'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,ORIG\_PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

SrcID

,ReportScope

,X.Currency

,Converted

,@PID AS PID

, PID AS ORIG\_PID

,@Product

,CID

,[CounterParty]

,maturityValue \* ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO, 0) AS MaturityValue

,maturityBucket

,CollateralClass

,CollateralValue

,CollateralCurrency

,CASE

--Step 1

WHEN Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1)

AND (maturityValue \* ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO, 0)) <= @FDIC\_Threshhold

AND desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS INSURED

,[Trigger]

,@rehypothecated

,Internal

,InternalCounterParty

,Portfolio

,DESC1

,CompanyName

,ReportType

,X.ASOFDATE

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN X.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN X.currency IS NULL OR X.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN X.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,X.customerNumber AS customerNumber

,X.orig\_marketValue AS orig\_marketValue

,X.orig\_maturityValue AS orig\_maturityValue

,X.orig\_currency AS orig\_currency

,X.maturityDate AS maturityDate

,@CreationDate AS CreationDate

FROM (

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- OD10 in the first query of the Union

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID\_OD10 AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,FMD.SECTOR

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,fmd.DEAL\_BALANCE AS DEAL\_BALANCE

,FMD.CURRENCY AS CURRENCY\_DEAL\_BALANCE

,fmd.Deal\_Mat\_date AS DEAL\_MAT\_DATE

,FMD.COMPANY\_CODE AS COMPANY\_CODE

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,NULL AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNAL AS SRC\_INTERNAL

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterParty]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,FMD.CONTACT\_ID

,ref\_Internal\_cp.InternalCounterParty AS REF\_INTERNAL\_CP

,ref\_en.reportingentityDescription AS REF\_REPORTING\_ENTITY

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN src.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID\_OD10

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale\_OD10

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2)

AND FMD.ASOFDATE = @AsofDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,FMD.SECTOR

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,fmd.DEAL\_BALANCE AS DEAL\_BALANCE

,FMD.CURRENCY AS CURRENCY\_DEAL\_BALANCE

,fmd.Deal\_Mat\_date AS DEAL\_MAT\_DATE

,FMD.COMPANY\_CODE AS COMPANY\_CODE

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,NULL AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNAL AS SRC\_INTERNAL

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterParty]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,FMD.CONTACT\_ID

,ref\_Internal\_cp.InternalCounterParty AS REF\_INTERNAL\_CP

,ref\_en.reportingentityDescription AS REF\_REPORTING\_ENTITY

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN src.company comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND ((ref\_ds.OD4\_5 = @OD4\_OD5\_Ratio AND FMD.DESC1 <> @DESC1\_Exclude) OR (FMD.DESC1 = @DESC1\_Exclude AND DEAL\_BALANCE > 0))

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2)

AND FMD.ASOFDATE = @AsofDate

) X

LEFT JOIN app.US\_FR2052A\_REF\_OperationalDepositTag ODT

ON X.DESC1 = ODT.TB\_LINE

AND ODT.CONTACT\_ID = X.CONTACT\_ID

AND ODT.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.5

-- \*\* Purpose: O.D.5 is combined with O.D.10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID\_OD10 = 'O.D.10'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @PRODUCT ='Non-Operational Accounts'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @wholeSale\_OD10 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD10\_Filter'

AND PID = 'O.D.10')

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @PID = 'O.D.5'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @DESC1\_Exclude = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'DESC1') -- DESC1 = 4960

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @FilterName3 = 'Sector\_Null\_values'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @OD4\_OD5\_Ratio = 'OD4\_OD5\_Ratio'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Orig\_PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

SrcID

,ReportScope

,X.Currency

,Converted

,@PID AS PID

,PID AS Orig\_PID

,@PRODUCT

,CID

,CounterParty

,maturityValue \* (1-ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO, 0)) AS MaturityValue

,maturityBucket

,CollateralClass

,CollateralValue

,CollateralCurrency

,CASE

--Step 1

WHEN Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND (maturityValue \* (1-ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO, 0))) <= @FDIC\_Threshhold

AND desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS INSURED

,[Trigger]

,@rehypothecated

,Internal

,InternalCounterParty

,Portfolio

,DESC1

,CompanyName

,ReportType

,X.ASOFDATE

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN X.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN X.currency IS NULL OR X.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN X.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,X.customerNumber AS customerNumber

,X.orig\_marketValue AS orig\_marketValue

,X.orig\_maturityValue AS orig\_maturityValue

,X.orig\_currency AS orig\_currency

,X.maturityDate AS maturityDate

,@CreationDate AS CreationDate

FROM (

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.10

-- \*\* Purpose: First Union is OD10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, 'O.D.10' AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty],FMD.SECTOR

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,fmd.DEAL\_BALANCE AS DEAL\_BALANCE

,FMD.CURRENCY AS CURRENCY\_DEAL\_BALANCE

,fmd.Deal\_Mat\_date AS DEAL\_MAT\_DATE

,FMD.COMPANY\_CODE AS COMPANY\_CODE

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,NULL AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterParty]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,FMD.CONTACT\_ID

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN src.company comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale\_OD10

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'cid'

AND PID = 'o.d.1')

AND ref\_ds.OD1\_Check NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @OD1\_Filter

AND PID = @PID2 )

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,FMD.SECTOR

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day 60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,fmd.DEAL\_BALANCE AS DEAL\_BALANCE

,FMD.CURRENCY AS CURRENCY\_DEAL\_BALANCE

,fmd.Deal\_Mat\_date AS DEAL\_MAT\_DATE

,FMD.COMPANY\_CODE AS COMPANY\_CODE

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,NULL AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterParty]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,FMD.CONTACT\_ID

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN src.company comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND (((ref\_ds.OD4\_5 = @OD4\_OD5\_Ratio AND FMD.DESC1 <> @DESC1\_Exclude) OR (FMD.DESC1 = @DESC1\_Exclude AND DEAL\_BALANCE > 0))

OR FMD.DESC1 IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName3

AND PID = @CID2))

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND (ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2) OR FMD.SECTOR IS NULL)

AND FMD.ASOFDATE = @AsofDate

AND FMD.ASOFDATE = @AsofDate

) X

LEFT JOIN app.US\_FR2052A\_REF\_OperationalDepositTag ODT

ON X.DESC1 = ODT.TB\_LINE

AND ODT.CONTACT\_ID = X.CONTACT\_ID

AND ODT.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.D.6'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @trigger = 'N'

SET @wholeSale = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD6\_Filter'

AND PID = 'O.D.6')

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FILTER\_FDIC = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2)

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Deposits

\*\* PID: O.D.7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--SET VARIABLES

SET @PID = 'O.D.7'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @wholeSale = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD7\_Filter'

AND PID = @PID)

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FilterName = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FilterName AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FilterName AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'')<> ref\_en.reportingentityDescription

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2)

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'O.D.9'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @wholeSale = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD9\_Filter'

AND PID = 'O.D.9')

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FilterName = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FilterName AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1 AND FILTERNAME = @FilterName) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FilterName AND Module = @Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1 AND FILTERNAME = @FilterName)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale

AND (ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @CID

AND PID = @PID2) OR FMD.SECTOR IS NULL)

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.D.10'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @trigger = 'N'

SET @wholeSale = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD10\_Filter'

AND PID = 'O.D.10')

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @Filter\_CID1 = 'CID1'

SET @Filter\_CID2 = 'CID2'

SET @Filter\_CID3 = 'CID3'

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @PID1 = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FilterName = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @FDIC = 'FDIC\_1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CID\_VALUE1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CID\_VALUE2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CID\_VALUE3

ELSE ref\_cp.CID END AS [CID]

,CASE WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID1) THEN @CounterParty1

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID2) THEN @CounterParty2

WHEN FMD.DESC1 IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @CID2 AND FilterName = @Filter\_CID3) THEN @CounterParty3

ELSE ref\_cp.COUNTERPARTY END AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (fmd.deal\_balance)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID1 AND FilterName = @FDIC AND Module =@Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1) THEN @FDIC\_INSURED

--Step 2

WHEN FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID1 AND FilterName = @FDIC AND Module =@Company\_Code)

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@PID1 AND Module = @DESC1)

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(fmd.deal\_balance), @AsofDate)) <= @FDIC\_Threshhold)

AND fmd.desc1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @PID1 AND FilterName = @Filter\_Uninsured)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [INSURED]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.Currency IS NULL OR fmd.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,fmd.deal\_balance AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.T\_FMDDATA fmd

LEFT JOIN SRC.COMPANY comp

ON fmd.COMPANY\_CODE = comp.COMPANY\_CODE

AND FMD.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_DepositScope ref\_ds

ON fmd.DESC1 = ref\_ds.DESC1

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_ds.WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16 = @wholeSale

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName

AND PID = @CID2)

AND FMD.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.2 - INTEREST

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

------ SET VARIABLES

SET @PID = 'O.D.2'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName2 = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @OD2\_OR\_OD3 = 'O.D.2-3'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @FILTER\_FDIC = 'FDIC\_1'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

('Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (MM.TOT\_INTEREST\_AMT)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(MM.TOT\_INTEREST\_AMT), @AsofDate)) END AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN (FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module =@Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1))

THEN @FDIC\_INSURED

--Step 2

WHEN

FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- In company Code US0010001 OR US0010008

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1) -- 4360 doesn't fit in this group

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND FilterName = @Filter\_Uninsured) -- 4360 doesn't fit in this group

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(MM.TOT\_INTEREST\_AMT), @AsofDate)) <= @FDIC\_Threshhold)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterParty]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,MM.TOT\_INTEREST\_AMT AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM SRC.T\_FMDDATA fmd

LEFT JOIN src.Multi\_mm MM

ON fmd.contact\_ID = MM.contract\_no

AND FMD.ASOFDATE = MM.ASOFDATE

LEFT JOIN src.company COMP

ON fmd.company\_code = comp.[company\_code]

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = FMD.Customer\_No

WHERE fmd.Customer\_no IN --New method of identifying relationship accounts

(

SELECT FMD2.CUSTOMER\_NO

FROM SRC.T\_FMDDATA FMD2

WHERE ISNULL(FMD2.DEAL\_LOCAL\_BAL, 0) <> 0 AND FMD2.ASOFDATE = @AsofDate AND FMD2.CUSTOMER\_NO IS NOT NULL

GROUP BY FMD2.CUSTOMER\_NO

HAVING COUNT(DISTINCT FMD2.CONTACT\_ID) > 1

)

AND FMD.DESC1 IN (SELECT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @OD2\_OR\_OD3 AND FilterName = @DESC1)

AND ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.company\_code = MM.co\_code

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND ref\_cp.CounterParty IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName2

AND PID = @CID2)

AND FMD.ASOFDATE = @AsofDate

AND MM.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.3 - INTEREST

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.D.3'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @OD2\_OR\_OD3 = 'O.D.2-3'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @FILTER\_FDIC = 'FDIC\_1'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

('Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,CASE WHEN fmd.currency = @Currency THEN (MM.TOT\_INTEREST\_AMT)

ELSE (dbo.fn\_CurrencyExchange

(fmd.Currency,(MM.TOT\_INTEREST\_AMT), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN (FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module =@Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1))

THEN @FDIC\_INSURED

--Step 2

WHEN

FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- In company Code US0010001 OR US0010008

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1) -- 4360 doesn't fit in this group

AND FMD.DESC1 NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND FilterName = @Filter\_Uninsured) -- 4360 doesn't fit in this group

AND ((dbo.fn\_CurrencyExchange (fmd.Currency,(MM.TOT\_INTEREST\_AMT), @AsofDate)) <= @FDIC\_Threshhold)

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

--,fmd.COMPANY\_code

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,MM.TOT\_INTEREST\_AMT AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.Multi\_mm MM

LEFT JOIN SRC.T\_FMDDATA fmd --src.Multi\_mm MM

ON fmd.contact\_ID = MM.contract\_no

AND FMD.ASOFDATE = MM.ASOFDATE

LEFT JOIN src.COMPANY COMP

ON fmd.company\_code = comp.[company\_code]

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = FMD.Customer\_No

WHERE fmd.Customer\_no NOT IN --New method of identifying relationship accounts

(

SELECT FMD2.CUSTOMER\_NO

FROM SRC.T\_FMDDATA FMD2

WHERE ISNULL(FMD2.DEAL\_LOCAL\_BAL, 0) <> 0 AND FMD2.ASOFDATE = @AsofDate AND FMD2.CUSTOMER\_NO IS NOT NULL

GROUP BY FMD2.CUSTOMER\_NO

HAVING COUNT(DISTINCT FMD2.CONTACT\_ID) > 1

)

AND FMD.DESC1 IN (SELECT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @OD2\_OR\_OD3 AND FilterName = @DESC1)

AND ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName

AND PID = @CID2)

AND FMD.company\_code = MM.co\_code

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND FMD.ASOFDATE = @AsofDate

AND MM.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.4 - INTEREST

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'O.D.4'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @AsofDate = @asofdate

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @wholeSale = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD4\_Filter'

AND PID = @PID

AND Filter = 'N/A'

)

SET @CounterParty1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party1')

SET @CounterParty2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party2')

SET @CounterParty3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'Counter Party3')

SET @CID\_VALUE1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue1')

SET @CID\_VALUE2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue2')

SET @CID\_VALUE3 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'CIDValue3')

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @OD4\_OD5 = 'O.D.4-5'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @FILTER\_FDIC = 'FDIC\_1'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate)

SELECT

('Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterParty]

,CASE WHEN fmd.currency = @Currency THEN (MM.TOT\_INTEREST\_AMT \* ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO,0))

ELSE ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO,0) \* (dbo.fn\_CurrencyExchange

(fmd.Currency,(MM.TOT\_INTEREST\_AMT), @AsofDate)) END

AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN (FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID= @FDIC\_PID AND Module = @DESC1))

THEN @FDIC\_INSURED

--Step 2

WHEN

LTRIM(RTRIM(FMD.Company\_code)) IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module =@Company\_Code) -- In company Code US0010001 OR US0010008

AND LTRIM(RTRIM(FMD.DESC1)) NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1) -- 4360 IS NOT in this group

AND LTRIM(RTRIM(FMD.DESC1)) NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND FilterName = @Filter\_Uninsured) -- 4360 IS NOT in this group

AND (ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO,0) \*(dbo.fn\_CurrencyExchange (fmd.Currency,(LTRIM(RTRIM(MM.TOT\_INTEREST\_AMT))), @ASOFDATE))) <= @FDIC\_Threshhold

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,MM.TOT\_INTEREST\_AMT AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.Multi\_mm MM

LEFT JOIN SRC.T\_FMDDATA fmd

ON fmd.contact\_ID = MM.contract\_no

AND FMD.ASOFDATE = MM.ASOFDATE

LEFT JOIN src.COMPANY COMP

ON fmd.company\_code = comp.[company\_code]

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_OperationalDepositTag ODT

ON ODT.CONTACT\_ID = FMD.CONTACT\_ID

AND ODT.TB\_Line = fmd.DESC1

AND ODT.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND FMD.DESC1 IN (SELECT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @OD4\_OD5 AND FilterName = @DESC1)

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName

AND PID = @CID2)

AND FMD.company\_code = MM.co\_code

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND FMD.ASOFDATE = @AsofDate

AND MM.ASOFDATE = @AsofDate

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

---- \*\* Portfolio: Commercial and Retail Deposits

---- \*\* PID: O.D.5 - INTEREST

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.D.5'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @Contac\_ID = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'CONTACT\_ID'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @wholeSale = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'OD5\_Filter'

AND PID = @PID

AND Filter = 'N/A'

)

SET @trigger = 'N'

SET @OD1\_Filter = 'OD1\_Check'

SET @CID = 'CID'

SET @trigger = 'N'

SET @FDIC\_PID = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @DESC1 = 'DESC1'

SET @CID2 = 'OD1-13'

SET @FilterName = 'CID\_ALL'

SET @PID2 = 'O.D.1'

SET @Filter\_Uninsured = 'Uninsured'

SET @FDIC\_INSURED = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = 'OD1-13'

AND FilterName = 'FDIC\_INSURED')

SET @FILTER\_FDIC = 'FDIC\_1'

SET @OD4\_OD5 = 'O.D.4-5'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

('Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterPary]

,(dbo.fn\_CurrencyExchange (fmd.Currency,(MM.TOT\_INTEREST\_AMT \* (1-ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO,0))), @AsofDate))AS [maturityValue]

,CASE WHEN fmd.DEAL\_MAT\_DATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

--Step 1

WHEN (FMD.Company\_code IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FilterName AND Module = @Company\_Code) -- STEP 1

AND FMD.DESC1 IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID=@FDIC\_PID AND Module = @DESC1))

THEN @FDIC\_INSURED

--Step 2

WHEN

LTRIM(RTRIM(FMD.Company\_code)) IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @FILTER\_FDIC AND Module = @Company\_Code) -- In company Code US0010001 OR US0010008

AND LTRIM(RTRIM(FMD.DESC1)) NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND Module = @DESC1) -- 4360 IS NOT in this group

AND LTRIM(RTRIM(FMD.DESC1)) NOT IN (SELECT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @FDIC\_PID AND FilterName = @Filter\_Uninsured) -- 4360 IS NOT in this group

AND ((1-ISNULL(ODT.OPERATIONAL\_DEPOSIT\_RATIO,0))\* dbo.fn\_CurrencyExchange (fmd.Currency,(LTRIM(RTRIM(MM.TOT\_INTEREST\_AMT))), @ASOFDATE)) <= @FDIC\_Threshhold

THEN @FDIC\_INSURED

--Step 3

ELSE @Filter\_Uninsured

END AS [Insured]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,fmd.DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END aS reasonableness\_Sector

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency IS NULL OR fmd.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN fmd.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,NULL AS orig\_marketValue

,MM.TOT\_INTEREST\_AMT AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS maturityDate

,@CreationDate AS CreationDate

FROM src.Multi\_mm MM

LEFT JOIN SRC.T\_FMDDATA fmd

ON fmd.contact\_ID = MM.contract\_no

AND FMD.ASOFDATE = MM.ASOFDATE

LEFT JOIN src.COMPANY COMP

ON fmd.company\_code = comp.[company\_code]

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_OperationalDepositTag ODT

ON ODT.CONTACT\_ID = FMD.CONTACT\_ID

AND ODT.TB\_Line = fmd.DESC1

AND ODT.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = fmd.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = fmd.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = FMD.Customer\_No

WHERE ODT.CONTACT\_ID IS NULL

AND ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND FMD.DESC1 IN (SELECT FILTER FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @OD4\_OD5 AND FilterName = @DESC1)

AND DEAL\_LOCAL\_BAL IS NOT NULL

AND DEAL\_LOCAL\_BAL <> 0

AND ref\_cp.CounterParty NOT IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @FilterName

AND PID = @CID2)

AND FMD.company\_code = MM.co\_code

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND FMD.ASOFDATE = @AsofDate

AND MM.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.7 (Interest -STB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID\_ALL = 'O.D.1-13'

SET @FDIC\_Threshhold = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID\_ALL

AND FilterName = 'FDIC\_Threshold')

SET @PID = 'O.D.7'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @Trigger =(SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Trigger')

SET @FDIC =(SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'FDIC')

SET @AsofDate = @asofdate

SET @Filter\_Uninsured = 'UNINSURED'

SET @FILTER\_INTEREST = 'INTEREST'

SET @FILTER\_ACCTNGTYPE = 'ACCTNGTYPE'

SET @FILTER\_INVTYPE = 'INVTYPE'

SET @BR = 'BR'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100)))AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS [Product]

,ref\_cp.CID AS CID

,ref\_cp.CounterParty AS counterPartySector

,CASE WHEN TPOS.CCY = @Currency THEN (TPOS.PRINAMT)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,(TPOS.PRINAMT), @AsofDate)) END AS [maturityValue]

,CASE WHEN SECM.MDATE IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE)<=60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,@FDIC AS [Insured]

,CASE WHEN @Trigger = 'Y' THEN 'True'

Else 'False' End AS [TRIGGER]

,@rehypothecated AS Rehyothecated

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,ref\_en.BranchName AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN CUST.CTYPE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector]

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N' AS [reasonableness\_contract\_dealno]

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,CUST.CNO AS customerNumber

,NULL AS orig\_marketValue

,TPOS.PRINAMT AS orig\_maturityValue

,TPOS.CCY AS orig\_currency

,SECM.MDATE AS maturityDate

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_CUST CUST

ON CUST.CNO = SECM.ISSUER

AND CUST.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = TPOS.BR

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.CTYPE

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND TPOS.ASOFDATE = @AsofDate

AND SECM.MDATE > @AsofDate

AND TPOS.BR IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR)

AND TPOS.INVTYPE IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND Module = @FILTER\_INTEREST AND FilterName = @FILTER\_INVTYPE)

AND SECM.ACCTNGTYPE IN (SELECT DISTINCT Filter FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID AND Module = @FILTER\_INTEREST AND FilterName = @FILTER\_ACCTNGTYPE)

AND TPOS.SETTAVGCOST <> 0

AND TPOS.ASOFDATE = SECM.ASOFDATE

AND SECM.ASOFDATE = @AsofDate

AND CUST.ASOFDATE = SECM.ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.D.11

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.D.11'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @OD1\_Filter = 'OD1\_Check'

SET @trigger = 'N'

SET @CID = 'CID'

SET @PID1 = 'O.D.1-13'

SET @Company\_Code = 'COMPANY\_CODE'

SET @FilterName = 'FDIC\_1'

SET @DESC1 = 'DESC1'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @Filter\_Uninsured = 'UNINSURED'

SET @rehypothecated = (SELECT LTRIM(RTRIM(Filter))

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'rehypothecated'

AND PID = 'OD1-13'

AND Module = 'all')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_depositsOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,[counterPartySector]

,maturityValue

,maturityBucket

,collateralClass

,collateralValue

,CollateralCurrency

,insured

,[trigger]

,[Rehypothecated]

,internal

,internalCounterParty

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate)

SELECT

'CDARS|'+CDARS.AccountId+'|'+CDARS.OriginalPrincipalAmount+'|'+CAST(CDARS.Asofdate as varchar(100)) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN CDARS.currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS PID

, Product AS Product

,ref\_cp.CID AS [CID]

,ref\_cp.COUNTERPARTY AS [CounterParty]

,CASE WHEN CDARS.currency = @Currency THEN (CAST(CDARS.AnticipatedPayoutAtMaturity AS DECIMAL(30,16)))

ELSE (dbo.fn\_CurrencyExchange

(CDARS.Currency,(CAST(CDARS.AnticipatedPayoutAtMaturity AS DECIMAL(30,16))), @AsofDate)) END AS [maturityValue]

,CASE WHEN CDARS.maturityDate IS NULL THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CDARS.maturityDate, 'Open')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,CDARS.maturityDate)BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CDARS.maturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,CDARS.maturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CDARS.maturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,CDARS.maturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CDARS.maturityDate, '> 1')

END

END

END

END AS [maturityBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE

WHEN LTRIM(RTRIM(CDARS.Company\_code)) IN (SELECT DISTINCT LTRIM(RTRIM(FILTER)) FROM src.US\_FR2052A\_REF\_TableFilters WHERE PID = @PID1 AND FilterName = @FilterName AND Module = @Company\_Code)

THEN 'FDIC'

ELSE 'Uninsured'

END AS [INSURED]

,CASE WHEN @Trigger = 'y' THEN 'True' Else 'False' End AS [Trigger]

,@rehypothecated AS [Rehypothecated]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.[InternalCounterParty] AS [InternalCounterPary]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,comp.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [ASOFDATE]

,CASE WHEN CDARS.CustomerId IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN CDARS.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector]

,CASE WHEN CDARS.Company\_Code IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N' AS [reasonableness\_contract\_dealno]

,((SELECT CASE WHEN CDARS.maturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN CDARS.currency IS NULL OR CDARS.currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN CDARS.currency NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END)

)

AS [EVENT]

,CDARS.CustomerId AS customerNumber

,NULL AS orig\_marketValue

,CDARS.AnticipatedPayoutAtMaturity AS orig\_maturityValue

,CDARS.currency AS orig\_currency

,CDARS.maturityDate AS maturityDate

,@CreationDate AS CreationDate

FROM APP.US\_FR2052A\_REF\_CDARS CDARS

LEFT JOIN SRC.COMPANY comp

ON CDARS.COMPANY\_CODE = comp.COMPANY\_CODE

AND CDARS.ASOFDATE = comp.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = CDARS.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.Sector = CDARS.Sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.CUSTOMER\_ID = CDARS.CustomerId

WHERE

CDARS.Asofdate = @AsofDate

AND CDARS.MaturityDate > @AsofDate

AND CAST(CDARS.AnticipatedPayoutAtMaturity AS DECIMAL(30,16)) <> 0

AND CAST(CDARS.AnticipatedPayoutAtMaturity AS DECIMAL(30,16)) is not Null

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_CurrentReference] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_CurrentReference]

AS

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

DECLARE @ASOFDate DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_CDARS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('[app].[US\_FR2052A\_REF\_CDARS]', 'U') IS NOT NULL

DROP TABLE [app].[US\_FR2052A\_REF\_CDARS];

SELECT Orig.[ReportingEntity], Orig.[Company\_Code], Orig.[AccountId], [AccountTitle], [CustomerTaxId], [OriginalPrincipalAmount], [Currency], [Sector], [CounterParty], [InternalTransactionIdr], [CustomerId], [BookBalance], [AnticipatedPayoutAtMaturity]

, [EffectiveDate], [MaturityDate], [TermType], [AnnaulInterestRate], [RelationshipAccountId], [RelationshipCustomerId], [CompanyID], [CompanyName], [SalesContactName], [InterestPaymentFrequency], [InterestDisbursementType]

, [MaturityDisbursementType], [PlacementType], [SourceOfFunds], [OneWayIdr], [AccruedInterestToDate], [AccruedInterestMTD], [InterestPaidYTD], [InterestPaidLastYear], [AccountOpeningFee], [OneWayRevenue]

, @ASOFDATE AS [Asofdate], Orig.[Asofdate] AS Orig\_AsOfDate

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_CDARS]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CDARS] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, A.Company\_Code, A.AccountId, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CDARS] A

WHERE A.Asofdate <= @ASOFDATE

GROUP BY A.ReportingEntity, A.Company\_Code, A.AccountId

) New

ON Orig.ReportingEntity = New.ReportingEntity AND Orig.Company\_Code = New.Company\_Code

AND Orig.AccountId = New.AccountId AND Orig.Asofdate = New.MaxDate;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_COLLATERAL \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('[app].[US\_FR2052A\_REF\_COLLATERAL]', 'U') IS NOT NULL DROP TABLE [app].[US\_FR2052A\_REF\_COLLATERAL];

SELECT Orig.[CollateralType], Orig.[Description], Orig.[CollateralClassHQLA], Orig.[CollateralClassFR2052A], Orig.[Comments], Orig.[Inputter], Orig.[Authoriser], Orig.[Reportingentity]

, @ASOFDATE AS [Asofdate], Orig.[Asofdate] AS Orig\_AsOfDate

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_COLLATERAL]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_COLLATERAL] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, A.[CollateralType], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_COLLATERAL] A

WHERE A.Asofdate <= DATEADD(Day, 3, @ASOFDATE)

GROUP BY A.ReportingEntity, A.[CollateralType]

) New

ON Orig.ReportingEntity = New.ReportingEntity AND Orig.[CollateralType] = New.[CollateralType]

AND Orig.Asofdate = New.MaxDate;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_CounterParty \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('[app].[US\_FR2052A\_REF\_CounterParty]', 'U') IS NOT NULL DROP TABLE [app].[US\_FR2052A\_REF\_CounterParty];

SELECT Orig.[Sector], [Description], [CID], [CounterParty], [ExceptionLogicNeeded], [Comments], [Inputter], [Authoriser], Orig.[ReportingEntity]

, @ASOFDATE AS [Asofdate], [Asofdate] AS Orig\_AsOfDate

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_CounterParty]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CounterParty] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, A.[Sector], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CounterParty] A

WHERE A.Asofdate <= DATEADD(Day, 3, @ASOFDATE)

GROUP BY A.ReportingEntity, A.[Sector]

) New

ON Orig.ReportingEntity = New.ReportingEntity AND Orig.[Sector] = New.[Sector]

AND Orig.Asofdate = New.MaxDate;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_CounterPartySTB \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('[app].[US\_FR2052A\_REF\_CounterPartySTB]', 'U') IS NOT NULL DROP TABLE [app].[US\_FR2052A\_REF\_CounterPartySTB];

SELECT Orig.[Ctype], [CustomerType], [CID], [CounterParty], [Comments], [Inputter], [Authoriser], Orig.[ReportingEntity]

, @ASOFDATE AS [Asofdate], Orig.[Asofdate] AS Orig\_AsOfDate

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_CounterPartySTB]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CounterPartySTB] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, A.[Ctype], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CounterPartySTB] A

WHERE A.Asofdate <= DATEADD(Day, 3, @ASOFDATE)

GROUP BY A.ReportingEntity, A.[Ctype]

) New

ON Orig.ReportingEntity = New.ReportingEntity AND Orig.[Ctype] = New.[Ctype]

AND Orig.Asofdate = New.MaxDate;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_DepositScope \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('[app].[US\_FR2052A\_REF\_DepositScope]', 'U') IS NOT NULL DROP TABLE [app].[US\_FR2052A\_REF\_DepositScope];

SELECT [ReportScopeID], Orig.[DESC1], [DepositDescription], [Step2], [OD1\_Check], [OD1\_2\_Check], [WholesaleLogic\_OD6\_8\_7\_9\_10\_11\_OW16], [OD4\_5], [Interestpayable], [Comments], [Inputter], [Authoriser], Orig.[ReportingEntity]

, @ASOFDATE AS [Asofdate], Orig.[Asofdate] AS Orig\_AsOfDate

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_DepositScope]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_DepositScope] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, A.[DESC1], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_DepositScope] A

WHERE A.Asofdate <= @ASOFDATE

GROUP BY A.ReportingEntity, A.[DESC1]

) New

ON Orig.ReportingEntity = New.ReportingEntity AND Orig.[DESC1] = New.[DESC1]

AND Orig.Asofdate = New.MaxDate;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_Entity \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_Entity', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].APP.US\_FR2052A\_REF\_Entity;

SELECT orig.Company\_code, orig.Company\_name, orig.ReportingEntityDescription, orig.Comments, orig.Inputter, orig.Authoriser, @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_Entity]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_Entity] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.ReportingEntity, Replace(A.Company\_Code,'BOCA','') AS Company\_code, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_Entity] A

-- To accommodate non-branch reportdates

WHERE A.Asofdate < = DATEADD(DAY, 3, @ASOFDate)

GROUP BY A.ReportingEntity, Replace(A.Company\_Code,'BOCA','')

) New

ON Orig.ReportingEntity = New.ReportingEntity

AND Orig.Company\_Code = New.Company\_code

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_Entity\_STB \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_EntitySTB', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].APP.US\_FR2052A\_REF\_EntitySTB;

SELECT orig.BranchID, orig.BranchName, orig.ReportingEntityDescription, orig.Inputter, orig.Authoriser, @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_EntitySTB]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_EntitySTB] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.BranchID, a.reportingentity, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_EntitySTB] A

WHERE A.Asofdate < = @asofdate

GROUP BY A.BranchID, a.reportingentity

) New

ON Orig.BranchID = New.BranchID

AND Orig.reportingentity = New.reportingentity

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_InternalCounterparty \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_InternalCounterParty', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_InternalCounterParty;

SELECT orig.Customer\_ID, orig.Short\_name, orig.Sector,orig.[Description],orig.Internal,orig.InternalCounterparty,orig.ExceptionLogicNeeded,orig.Comments, orig.Inputter, orig.Authoriser, @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_InternalCounterParty]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_InternalCounterParty] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.Customer\_ID, a.reportingentity, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_InternalCounterParty] A

WHERE A.Asofdate < = @asofdate

GROUP BY A.Customer\_ID, a.reportingentity

) New

ON Orig.Customer\_ID = New.Customer\_ID

AND Orig.reportingentity = New.reportingentity

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_InternalCounterparty\_STB\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_InternalCounterPartySTB', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_InternalCounterPartySTB;

SELECT orig.cno, orig.sn, orig.ccode,orig.ACCTINGTYPE,orig.Internal,orig.InternalCounterparty,orig.ExceptionLogicNeeded,orig.Comments, orig.Inputter, orig.Authoriser, @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_InternalCounterPartySTB]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_InternalCounterPartySTB] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.cno, a.reportingentity, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_InternalCounterPartySTB] A

WHERE A.Asofdate < = @asofdate

GROUP BY A.cno, a.reportingentity

) New

ON Orig.cno = New.cno

AND Orig.reportingentity = New.reportingentity

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_OperationalDepositTag\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_OperationalDepositTag', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_OperationalDepositTag;

SELECT orig.TB\_LINE, orig.CONTACT\_ID,orig.[CURRENCY],orig.[DEAL\_BALANCE],orig.[OPERATIONAL\_DEPOSIT\_RATIO],orig.[OPERATIONAL\_DEPOSIT\_BAL],orig.[NON\_OPERATIONAL\_DEPOSIT\_BAL],orig.[INPUTTER],orig.[AUTHORIZER], @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_OperationalDepositTag]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_OperationalDepositTag] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.TB\_LINE, A.CONTACT\_ID, a.reportingentity, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_OperationalDepositTag] A

WHERE A.Asofdate < = @asofdate

GROUP BY A.TB\_LINE, A.CONTACT\_ID, a.reportingentity

) New

ON Orig.TB\_LINE = New.TB\_LINE

AND Orig.CONTACT\_ID = New.CONTACT\_ID

AND Orig.reportingentity = New.reportingentity

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* REF\_OperationalDepositTag\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_OperationalDepositTag', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_OperationalDepositTag;

SELECT orig.TB\_LINE, orig.CONTACT\_ID,orig.[CURRENCY],orig.[DEAL\_BALANCE],orig.[OPERATIONAL\_DEPOSIT\_RATIO],orig.[OPERATIONAL\_DEPOSIT\_BAL],orig.[NON\_OPERATIONAL\_DEPOSIT\_BAL],orig.[INPUTTER],orig.[AUTHORIZER], @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_OperationalDepositTag]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_OperationalDepositTag] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.TB\_LINE, A.CONTACT\_ID, a.reportingentity, MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_OperationalDepositTag] A

WHERE A.Asofdate < = @asofdate

GROUP BY A.TB\_LINE, A.CONTACT\_ID, a.reportingentity

) New

ON Orig.TB\_LINE = New.TB\_LINE

AND Orig.CONTACT\_ID = New.CONTACT\_ID

AND Orig.reportingentity = New.reportingentity

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* US\_FR2052A\_REF\_CID\_Exception\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_CID\_Exception', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_CID\_Exception;

SELECT orig.[Sector], orig.[Industry], orig.[CustomerID], orig.[CID], orig.[Counterparty]

,orig.[INPUTTER],orig.[Authoriser], @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_CID\_Exception]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CID\_Exception] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.[Sector], A.[Industry], a.[CustomerID], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_CID\_Exception] A

WHERE A.Asofdate < = @ASOFDate

GROUP BY A.[Sector], A.[Industry], a.[CustomerID]

) New

ON ISNULL(Orig.[Sector], '') = ISNULL(New.[Sector], '')

AND ISNULL(Orig.[Industry], '') = ISNULL(New.[Industry], '')

AND ISNULL(Orig.[CustomerID], '') = ISNULL(New.[CustomerID], '')

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* US\_FR2052A\_REF\_Collateral\_MISDW\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_Collateral\_MISDW', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_Collateral\_MISDW;

SELECT orig.[PID],orig.[Portfolio],orig.[DESCR],orig.[CollateralClass],orig.[INPUTTER],orig.[Authoriser], @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_Collateral\_MISDW]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_Collateral\_MISDW] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.[PID], A.[Portfolio], a.[DESCR], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_Collateral\_MISDW] A

WHERE A.Asofdate < = @ASOFDate

GROUP BY A.[PID], A.[Portfolio], a.[DESCR]

) New

ON Orig.[PID] = New.[PID]

AND Orig.[Portfolio] = New.[Portfolio]

AND Orig.[DESCR] = New.[DESCR]

AND Orig.Asofdate = New.MaxDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* US\_FR2052A\_REF\_ReportType\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('APP.US\_FR2052A\_REF\_ReportType', 'U') IS NOT NULL

DROP TABLE [REPORT\_FR2052A].app.US\_FR2052A\_REF\_ReportType;

SELECT orig.[ReportType], orig.[PID],orig.[Product],orig.[Product\_Old], orig.[SID],orig.[SubProduct]

,orig.[INPUTTER],orig.[AUTHORISER], @ASOFDate as asofdate, orig.Asofdate as orig\_asofdate, orig.ReportingEntity

INTO [REPORT\_FR2052A].[app].[US\_FR2052A\_REF\_ReportType]

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_ReportType] Orig INNER JOIN

(-- Select latest date (Max Date) from original table

SELECT A.[ReportType], A.[PID], MAX(A.Asofdate) MaxDate

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_ReportType] A

WHERE A.Asofdate < = @ASOFDate

GROUP BY A.[ReportType], A.[PID]

) New

ON Orig.[ReportType] = New.[ReportType]

AND Orig.[PID] = New.[PID]

AND Orig.Asofdate = New.MaxDate

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_wholesale\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_wholesale\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_wholesale\_outflow]

\*\* Report Type: Wholesale Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[creationDate]

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, Tgt\_WO.AsofDate)) = sum(Tgt\_WO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_WO.Tgt\_wholesaleOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_WO.portfolio AS [Portfolio]

,Tgt\_WO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, Tgt\_WO.AsofDate))

--AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_WO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_WO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_WO.Tgt\_wholesaleOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationDate AS [creationDate]

FROM SRC.[T\_FMDDATA] fmd

INNER JOIN app.Tgt\_wholesaleOutflow Tgt\_WO

ON Tgt\_WO.SrcID = 'T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR)

AND TGT\_WO.ASOFDATE = FMD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_WO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND REF\_TBL.Portfolio = TGT\_WO.Portfolio

GROUP BY ref\_en.reportingentityDescription

,Tgt\_WO.reportScope

,Tgt\_WO.PID

,Tgt\_WO.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_WO.portfolio

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.9,O.W.10

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange

(SCHD.CCY,ABS(SCHD.PrinAmt), Tgt\_WO.AsofDate)) = sum(abs(Tgt\_WO.maturityValue))

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_WO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_WO.portfolio AS [Portfolio]

,Tgt\_WO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange

(SCHD.CCY,ABS(SCHD.PrinAmt), Tgt\_WO.AsofDate))

AS [AggregationFieldValue]

,COUNT(distinct schd.DEALNO+schd.BR+CAST(schd.INTENDDTE AS VARCHAR(50))+schd.SCHDSEQ )

AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_WO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(abs(Tgt\_WO.maturityValue)) AS [TargetFieldValue]

,COUNT(distinct Tgt\_WO.TGT\_wholesaleoutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,SCHD.ASOFDATE AS [AsOfDate]

,creationDate AS [creationDate]

FROM

SRC.US\_OPICS\_SCHD SCHD

INNER JOIN app.Tgt\_wholesaleOutflow Tgt\_WO

ON Tgt\_WO.SrcID = 'SCHD | '+schd.DEALNO+' | '+schd.BR+' | '+CAST(schd.INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ

AND Tgt\_WO.PID in ('O.W.9','O.W.10')

AND TGT\_WO.AsofDate = SCHD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_WO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND REF\_TBL.Portfolio = TGT\_WO.Portfolio

GROUP BY ref\_en.reportingentityDescription

,Tgt\_WO.reportScope

,Tgt\_WO.PID

,Tgt\_WO.Product

,SCHD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_WO.portfolio

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: O.W.11

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(STB.[maturityValue]) = sum(Tgt\_WO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_WO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_WO.portfolio AS [Portfolio]

,Tgt\_WO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(STB.[maturityValue]) AS [AggregationFieldValue]

,COUNT(distinct STB.TPOS\_ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_WO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_WO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_WO.TGT\_wholesaleoutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,STB.ASOFDATE AS [AsOfDate]

,creationDate AS [creationDate]

-- select Tgt\_WO.reportScope, Tgt\_WO.marketValue AS marketvalue, Tgt\_WO.PID

--linked server access

FROM --dbo.US\_OPICS\_DLDT DLDT

( SELECT

'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(B.BR)) + ' | ' + LTRIM(RTRIM(B.SECID)) + ' | ' + CAST(LTRIM(RTRIM(B.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(B.PORT)) + ' | ' + LTRIM(RTRIM(B.INVTYPE)) + ' | ' + CAST(B.COST AS VARCHAR (50)))) AS TPOS\_ID

, B.br

, B.ccy'CCY'

, B.secid

,D.CNO

,D.CTYPE

, A.mdate

,B.PrinAmt

,DBO.fn\_CurrencyExchange

(B.CCY,(B.PrinAmt), A.ASOFDATE) AS [maturityValue]

, A.ASOFDATE

FROM SRC.US\_OPICS\_secm A,

SRC.US\_OPICS\_tpos B,

SRC.US\_OPICS\_cust D

WHERE B.invtype = 'I' -- Issued investment type

AND A.issuer = D.cno

AND A.acctngtype = 'ISSUE'

AND A.secid = B.secid

AND B.br in ( '01','02')

AND (B.prinamt <> 0

OR B.unamortamt <> 0

OR B.settavgcost <> 0

OR B.mktval <> 0)

and B.invtype = 'I'

AND B.settqty <> 0

AND D.CMNE IN ('BOCNY' ) --Customer name is BOCNY or BOCNYIBF to identify BOC issued BONDs

--Above D.CMNE condition is temporary and should be changed to new criteria using A.DESCR.

--A.DESCR field is currently being created by OSD (Liu Li) in Production environment

AND A.ASOFDATE = B.ASOFDATE

AND B.ASOFDATE = D.ASOFDATE

) STB

INNER JOIN app.Tgt\_wholesaleOutflow Tgt\_WO

ON Tgt\_WO.SrcID = STB.TPOS\_ID

AND Tgt\_WO.PID = 'O.W.11'

AND Tgt\_WO.AsofDate = STB.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_WO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND TGT\_WO.Portfolio = REF\_TBL.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_WO.reportScope

,Tgt\_WO.PID

,Tgt\_WO.Product

,STB.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_WO.portfolio

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Deposits

\*\* PID: O.W.16

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(CASE WHEN TPOS.CCY = 'USD' THEN ABS(TPOS.PRINAMT)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,ABS(TPOS.PrinAmt), Tgt\_WO.ASOFDATE)) END ) = sum(abs(Tgt\_WO.maturityValue))

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_WO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_WO.portfolio AS [Portfolio]

,Tgt\_WO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(CASE WHEN TPOS.CCY = 'USD' THEN ABS(TPOS.PrinAmt)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,ABS(TPOS.PrinAmt), Tgt\_WO.ASOFDATE)) END )

AS [AggregationFieldValue]

,COUNT(distinct 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'

+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))

+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'

+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_WO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(abs(Tgt\_WO.maturityValue)) AS [TargetFieldValue]

,COUNT(distinct Tgt\_WO.Tgt\_wholesaleOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TPOS.ASOFDATE AS [AsOfDate]

,creationDate AS [creationDate]

FROM SRC.US\_OPICS\_TPOS TPOS

INNER JOIN app.Tgt\_wholesaleOutflow Tgt\_WO

ON Tgt\_WO.SrcID = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))

AND Tgt\_WO.PID = 'O.W.16'

AND Tgt\_WO.ASOFDATE = TPOS.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_WO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND ref\_tbl.Portfolio = TGT\_WO.Portfolio

-- AND ref\_tbl.Portfolio = Tgt\_WO.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_WO.reportScope

,Tgt\_WO.PID

,Tgt\_WO.Product

,TPOS.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_WO.portfolio

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* NON BRANCH

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other payables and margin collaterals & Wholesale Funding Transactions

-- \*\* PID: O.W.18 , O.W.19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[creationDate]

)

SELECT DISTINCT

(CASE

WHEN

SUM(dbo.fn\_CurrencyExchange

(NB\_WO.OriginalCurrencyOfMaturityAmount,(NB\_WO.MaturityAmountOriginalCurrency), TGT\_WO.ASOFDATE)) = sum(Tgt\_WO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_WO.Tgt\_OtherInflowID)

--Run a check to see if source = target

THEN 'Pass'

ELSE 'Fail'

END ) AS [Validation\_Check]

,Tgt\_WO.portfolio AS [Portfolio]

,Tgt\_WO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange

(NB\_WO.OriginalCurrencyOfMaturityAmount,(NB\_WO.MaturityAmountOriginalCurrency), TGT\_WO.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct NB\_WO.MaturityAmountOriginalCurrency) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_WO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_WO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_WO.Tgt\_wholesaleOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_WO.ASOFDATE AS [AsOfDate]

,creationDate AS [creationDate]

FROM SRC.US\_FR2052A\_NB\_OW NB\_WO

INNER JOIN app.Tgt\_wholesaleOutflow Tgt\_WO

ON Tgt\_WO.SrcID = 'NB\_OW | '+CAST(NB\_WO.ID AS VARCHAR) + ' | O.W.'+ CAST(NB\_WO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_WO.ReportingEntity +' | '+cast(NB\_WO.asofdate as VARCHAR)

and TGT\_WO.ASOFDATE = NB\_WO.ASOFDATE

----get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_WO.reportScope

AND NB\_WO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

-- ---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND REF\_tbl.portfolio = Tgt\_WO.Portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_WO.reportScope

,Tgt\_WO.PID

,Tgt\_WO.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,creationDate

,Tgt\_WO.portfolio

PRINT 'OW Ctrl complete :'+cast(getdate() as varchar)

END TRY

BEGIN CATCH

-- Execute ISSUE handling stroed procedure to get details about the ISSUEs

EXEC Proc\_InsertISSUEDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Unsecured\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Unsecured\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [sp\_Ctrl\_Unsecured\_Inflow]

\*\* Report Type: Unsecured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

-- SET XACT\_ABORT ON

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.Tgt\_assetInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

ON app.Tgt\_assetInflow(TGT\_assetInflowID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

--DROP INDEX IX\_Tgt\_UnsecuredInflow ON APP.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON APP.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_securedInflow') AND NAME ='IX\_Tgt\_securedInflow')

-- DROP INDEX IX\_Tgt\_securedInflow ON app.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_securedInflow

ON app.Tgt\_securedInflow(TGT\_securedInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_T24\_MULTI\_LD\_BALANCE') AND NAME ='IX\_bal')

-- DROP INDEX IX\_bal ON SRC.US\_T24\_MULTI\_LD\_BALANCE;

CREATE NONCLUSTERED INDEX IX\_bal

ON src.US\_T24\_MULTI\_LD\_BALANCE(id, date);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.TMP\_MG\_Mortgage\_Schedule') AND NAME ='IX\_TMP')

-- DROP INDEX IX\_TMP ON SRC.TMP\_MG\_Mortgage\_Schedule;

CREATE NONCLUSTERED INDEX IX\_TMP

ON src.TMP\_MG\_Mortgage\_Schedule(CONTRACT\_NO, DUE\_DATE);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

-- DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_FR2052A\_REF\_TableDetails') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON src.US\_FR2052A\_REF\_TableDetails;

CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

ON src.US\_FR2052A\_REF\_TableDetails(PID);

----insert data into target table

INSERT INTO APP.[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.3 (Multi-Account)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE WHEN (SUM(dbo.fn\_CurrencyExchange(MA.CURRENCY,(MA.EXPECT\_BALANCE), MA.ASOFDATE))) = sum(Tgt\_UI.[MaturityValue])

THEN 'Pass'

ELSE ('Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(MA.Currency,(MA.EXPECT\_BALANCE), MA.ASOFDATE)) AS [AggregationFieldValue]

,COUNT(distinct MA.ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum((Tgt\_UI.maturityValue)) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,MA.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM SRC.MULTI\_ACCOUNT MA

INNER JOIN APP.Tgt\_unsecuredInflow tgt\_UI

ON Tgt\_UI.SrcID = 'Multi-Account | '+cast(MA.ID as varchar)

AND tgt\_UI.PID IN ('I.U.3')

AND TGT\_UI.AsofDate = MA.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_UI.reportscope

AND ref\_en.Company\_name = Tgt\_UI.CompanyName

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_UI.portfolio

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,[MA].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.3 T\_FMDDATA

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION SELECT

CASE WHEN (SUM(dbo.fn\_CurrencyExchange(FMD.CURRENCY,(FMD.DEAL\_BALANCE), FMD.ASOFDATE))) = sum(Tgt\_UI.[MaturityValue])

THEN 'Pass'

ELSE ('Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(FMD.Currency,(FMD.DEAL\_BALANCE), FMD.ASOFDATE)) AS [AggregationFieldValue]

,COUNT(distinct FMD.N\_ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum((Tgt\_UI.maturityValue)) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,FMD.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM SRC.T\_FMDDATA FMD

INNER JOIN APP.Tgt\_unsecuredInflow tgt\_UI

ON Tgt\_UI.SrcID ='T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR)

AND tgt\_UI.PID IN ('I.U.3')

AND TGT\_UI.AsofDate = FMD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_UI.reportscope

AND ref\_en.Company\_name = Tgt\_UI.CompanyName

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_UI.portfolio

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,FMD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.U.4 T\_FMDDATA

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT

CASE WHEN abs((SUM((-1)\*(dbo.fn\_CurrencyExchange(fmd.CURRENCY, fmd.deal\_balance, FMD.ASOFDATE))-(dbo.fn\_CurrencyExchange(MA.CURRENCY, MA.EXPECT\_BALANCE, MA.ASOFDATE))))) = abs(sum(Tgt\_UI.[MaturityValue]))

THEN 'Pass'

ELSE ('Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM((-1)\*(dbo.fn\_CurrencyExchange(fmd.CURRENCY, fmd.deal\_balance, FMD.ASOFDATE))-(dbo.fn\_CurrencyExchange(MA.CURRENCY, MA.EXPECT\_BALANCE, MA.ASOFDATE))) AS [AggregationFieldValue]

,COUNT(distinct MA.ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum((Tgt\_UI.maturityValue)) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,MA.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.MULTI\_ACCOUNT MA

ON MA.CUSTOMER = FMD.CUSTOMER\_NO

AND MA.CATEGORY = FMD.CATEGORY

AND MA.ID = FMD.CONTACT\_ID

AND FMD.ASOFDATE = MA.ASOFDATE

INNER JOIN APP.Tgt\_unsecuredInflow tgt\_UI

ON (Tgt\_UI.SrcID = 'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR)

OR Tgt\_UI.SrcID = 'FMD | '+CAST(fmd.n\_ID AS VARCHAR)

OR Tgt\_UI.SrcID = 'MULTI-Account | '+CAST(fmd.n\_ID AS VARCHAR) )

AND tgt\_UI.PID IN ('I.U.4')

AND TGT\_UI.AsofDate = MA.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_UI.reportscope

AND ref\_en.Company\_name = Tgt\_UI.CompanyName

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_UI.portfolio

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,[MA].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: I.U.1 , IU2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(CASE WHEN SCHD.CCY = 'USD' THEN (SCHD.PrinAmt)

ELSE (DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), SCHD.ASOFDATE)) END ) = sum(Tgt\_UI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_UI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(CASE WHEN SCHD.CCY = 'USD' THEN (SCHD.PrinAmt)

ELSE (DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), SCHD.ASOFDATE)) END )

AS [AggregationFieldValue]

,COUNT(distinct SCHD.Dealno) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_UI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,SCHD.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

src.US\_OPICS\_SCHD SCHD

INNER JOIN APP.Tgt\_unsecuredInflow tgt\_UI

ON Tgt\_UI.SrcID = 'SCHD | '+ schd.DEALNO+' | '+schd.BR+' | '+CAST(INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ

and tgt\_ui.AsofDate = schd.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_UI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

--AND ref\_tbl.pid = fmd.PID

AND ref\_tbl.Portfolio = tgt\_ui.portfolio

GROUP BY ref\_en.reportingentityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,SCHD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: IU2 - FMD

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange

(FMD.CURRENCY,(FMD.DEAL\_BALANCE), FMD.ASOFDATE)) = sum(Tgt\_UI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_UI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange

(FMD.CURRENCY,(FMD.DEAL\_BALANCE), FMD.ASOFDATE)) AS [AggregationFieldValue]

,COUNT(distinct 'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR)) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_UI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,FMD.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

src.T\_FMDDATA FMD

INNER JOIN APP.Tgt\_unsecuredInflow tgt\_UI

ON Tgt\_UI.SrcID = 'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR)

and tgt\_ui.AsofDate = FMD.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_UI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

--AND ref\_tbl.pid = fmd.PID

AND ref\_tbl.Portfolio = tgt\_ui.portfolio

GROUP BY ref\_en.reportingentityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,FMD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.U.5 Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.PRINCIPAL ), bal.ASOFDATE)) = sum(Tgt\_UI.[MaturityValue])

--,COUNT(distinct (bal.ID + ' | '+bal.[DATE]) = sum(Tgt\_UI.[MaturityValue])

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.PRINCIPAL ), bal.ASOFDATE)) AS [AggregationFieldValue]

,COUNT(distinct (bal.ID + ' | '+bal.[DATE]))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_UI.[MaturityValue]) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,bal.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE bal

INNER JOIN

APP.Tgt\_unsecuredInflow tgt\_UI

ON ((Tgt\_UI.SrcID = 'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ BAL.ID+' | '+BAL.DATE )

)

and tgt\_UI.AsofDate = bal.ASOFDATE

and tgt\_ui.PID in ('I.U.5', 'I.U.6')

and Convert(Varchar(20),CAST(BAL.[DATE] as DATE),101) > tgt\_ui.asofdate

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

and fmd.DESC1 in ('0170', '0240', '0270', '0310', '0380', '0405', '0390', '0400')

LEFT JOIN SRC.MULTI\_LD LD

ON LD.ID = bal.ID

AND LD.ASOFDATE = bal.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_UI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_ui.portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,bal.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,ld.currency

,creationdate

--,bal.PRINCIPAL

--,bal.interest

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.U.5 Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.Interest ), bal.ASOFDATE)) = sum(Tgt\_UI.[MaturityValue])

--,COUNT(distinct (bal.ID + ' | '+bal.[DATE]) = sum(Tgt\_UI.[MaturityValue])

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.Interest ), bal.ASOFDATE)) AS [AggregationFieldValue]

,COUNT(distinct (bal.ID + ' | '+bal.[DATE]))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_UI.[MaturityValue]) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.TGT\_unsecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,bal.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE bal

INNER JOIN

APP.Tgt\_unsecuredInflow tgt\_UI

ON ((Tgt\_UI.SrcID = 'US\_T24\_MULTI\_LD\_BALANCE | Interest | '+ BAL.ID+' | '+BAL.DATE )

)

and tgt\_UI.AsofDate = bal.ASOFDATE

and tgt\_ui.PID in ('I.U.5', 'I.U.6')

and Convert(Varchar(20),CAST(BAL.[DATE] as DATE),101) > tgt\_ui.asofdate

INNER JOIN src.T\_FMDDATA FMD

ON FMD.CONTACT\_ID = BAL.ID

AND FMD.ASOFDATE = BAL.ASOFDATE

AND BAL.CO\_CODE = FMD.COMPANY\_CODE

and fmd.DESC1 in ('0170', '0240', '0270', '0310', '0380', '0405', '0390', '0400')

LEFT JOIN SRC.MULTI\_LD LD

ON LD.ID = bal.ID

AND LD.ASOFDATE = bal.ASOFDATE

AND LD.CO\_CODE = BAL.CO\_CODE

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_UI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_ui.portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_UI.reportScope

,Tgt\_UI.PID

,Tgt\_UI.Product

,bal.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,ld.currency

,creationdate

--,bal.PRINCIPAL

--,bal.interest

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Non - Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: CASH, Other Receivable, Wholesale Funding

-- \*\* PID: I.U.3, i.U.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange

(NB\_IU.OriginalCurrencyOfMaturityAmount,(NB\_IU.MaturityAmountOriginalCurrency), TGT\_UI.ASOFDATE))= sum(Tgt\_UI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_UI.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_UI.portfolio AS [Portfolio]

,Tgt\_UI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange

(NB\_IU.OriginalCurrencyOfMaturityAmount,(NB\_IU.MaturityAmountOriginalCurrency), TGT\_UI.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct NB\_IU.MaturityAmountOriginalCurrency)AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_UI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_UI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_UI.Tgt\_unsecuredInflowID)AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_UI.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM src.US\_FR2052A\_NB\_IU NB\_IU

INNER JOIN APP.Tgt\_unsecuredInflow Tgt\_UI

ON Tgt\_UI.SrcID = 'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+cast(NB\_IU.asofdate as VARCHAR)

AND NB\_IU.ASOFDATE = Tgt\_UI.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_UI.reportScope

AND NB\_IU.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND REF\_tbl.portfolio = Tgt\_UI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_UI.reportScope

,NB\_IU.currency

,Tgt\_UI.PID

,Tgt\_UI.Product

,Tgt\_UI.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_UI.portfolio

,creationdate

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_tgt\_agg\_logic\_check] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_tgt\_agg\_logic\_check]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Check the control tables for validation checks that have passed

and raise an error if not

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [sp\_Ctrl\_tgt\_agg\_logic\_check]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

--Check the control table for validation\_check = Fail

INSERT INTO app.ISSUETracer

SELECT '','','','',CTRL.TGT\_PID,

'Validation Check Fail in Ctrl\_Src\_tgt',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Ctrl\_Tgt\_Agg CTRL

where CTRL.validationcheck = 'FAIL'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Tgt\_Agg\_FR2052a] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Tgt\_Agg\_FR2052a]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: This Stored procedure will insert records into control table for all products

in report type FR2052a

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: [dbo].[sp\_Ctrl\_app.tgt\_Agg\_FR2052a]

\*\* Report Type: All Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.app.tgt\_unsecuredInflow') AND NAME ='IX\_app.tgt\_UnsecuredInflow')

-- DROP INDEX IX\_app.tgt\_UnsecuredInflow ON dbo.app.tgt\_unsecuredInflow;

--CREATE NONCLUSTERED INDEX IX\_app.tgt\_UnsecuredInflow

-- ON dbo.app.tgt\_unsecuredInflow(app.tgt\_unsecuredInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.app.tgt\_assetInflow') AND NAME ='IX\_app.tgt\_AssetInflow')

-- DROP INDEX IX\_app.tgt\_AssetInflow ON dbo.app.tgt\_assetInflow;

-- CREATE NONCLUSTERED INDEX IX\_app.tgt\_AssetInflow

-- ON dbo.app.tgt\_assetInflow(app.tgt\_assetInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.AGG\_FR2052a') AND NAME ='IX\_AGG\_2052a')

-- DROP INDEX IX\_app.tgt\_AssetInflow ON dbo.AGG\_FR2052a;

-- CREATE NONCLUSTERED INDEX IX\_AGG\_2052a

-- ON dbo.AGG\_FR2052a(AGG\_FR2052aID);

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.Ref\_table\_details') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON dbo.Ref\_table\_details;

-- CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

-- ON dbo.Ref\_table\_details(PID);

--Get Total rows from soure to ensure we have received all the data and get failure/success counts

BEGIN TRY

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--insert data into target table

INSERT INTO app.Ctrl\_tgt\_Agg

(

ValidationCheck

,app.tgt\_reportScope

,app.tgt\_PID

,app.tgt\_product

,app.tgt\_maturityValue

,app.tgt\_marketValue

--,app.tgt\_NumRec

,Agg\_reportScope

,Agg\_PID

,Agg\_product

,Agg\_maturityValue

,Agg\_marketValue

--,Agg\_NumRec

)

SELECT CASE WHEN ABS(ISNULL(Tgt.MaturityValue,0)) = ISNULL(AGG.MaturityValue,0)

and ABS(ISNULL(Tgt.MarketVALUE,0)) = ISNULL(AGG.MarketVALUE,0) THEN 'Pass' Else 'Fail' END AS ValidationCheck

,TGT.reportScope

,Tgt.PID

,Tgt.product

,ISNULL(Tgt.MaturityValue,0) AS [Target MaturityValue Value]

,ISNULL(Tgt.MarketVALUE,0) AS [Target MarketValue Value]

--,Tgt.NumRec

,AGG.ReportScope

,AGG.PID

,AGG.product

,ISNULL(AGG.MaturityValue,0) AS [Aggregation MaturityValue Value]

,ISNULL(AGG.MarketVALUE,0) AS [Aggregation MarketValue Value]

--,AGG.NumRec

FROM

(

SELECT reportScope, PID, product,

0 AS MaturityValue,

CASE WHEN PID = 'I.A.3' THEN

SUM( CASE WHEN DESC1 = '0060' THEN (ABS(TGT.marketValue))

WHEN DESC1 = '9003' THEN (-1)\*(ABS(TGT.marketValuE))

END )

ELSE SUM(TGT.MARKETVALUE) END AS MarketVALUE

FROM app.tgt\_assetinflow tgt WHERE MARKETVALUE <> 0 group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product,

CASE WHEN PID = 'I.U.4'

THEN

SUM(CASE WHEN TGT.SrcID like 'FMD\_ALL | %' THEN (-1)\*TGT.maturityValue

WHEN TGT.SrcID like 'MULTI%' THEN (-1)\* TGT.maturityValue

WHEN TGT.SrcID like 'T\_FMDDATA | %' THEN (-1)\* abs(TGT.maturityValue)

END )

ELSE SUM(ABS(MATURITYvALUE)) END AS MaturityValue,

0 AS MarketVALUE

FROM app.tgt\_unsecuredinflow TGT group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityAmountCurrency1 + MaturityAmountCurrency2) AS MaturityValue,SUM(0) AS MarketVALUE

FROM app.tgt\_foreignexchange group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_securedinflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_wholesaleOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product,

CASE WHEN PID = 'O.O.4'

THEN

SUM(CASE WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_TOTAL |%' THEN ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN (-1)\* ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_OS |%' THEN ABS(TGT.maturityValue)

END )

ELSE CASE WHEN PID = 'O.O.5'

THEN

SUM(CASE WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_OS\_4027 |%' THEN ABS( TGT.maturityValue)

END )

ELSE SUM(MATURITYvALUE) END END

AS [MaturityValue],

0 AS MarketVALUE

FROM app.tgt\_otherOUTFLOW TGT group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, 0 AS MaturityValue,SUM(marketvalue) AS MarketVALUE

FROM app.tgt\_informationalsupplemental group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_securedOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_depositsOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_otherINflow group by reportScope, PID, product

) TGT

FULL OUTER JOIN

(

SELECT reportScope, PID, product, isnull(sum(agg.maturityValue),0) AS MATURITYvALUE , ISNULL(SUM(agg.marketValue),0) AS MarketVALUE--COUNT(\*) AS NumRec

FROM app.AGG\_FR2052a agg

--WHERE PID = 5

GROUP BY reportScope, PID, product

) AGG

ON TGT.reportScope = AGG.ReportScope

AND SUBSTRING( tgt.PID,5,100)= agg.PID

AND tgt.product = agg.Product

where tgt.PID not in( 'S.FX.2', 'i.a.1','i.o.8')

ORDER BY TGT.PID

-- Error handling

END TRY

BEGIN CATCH

EXEC [dbo].[Proc\_InsertissueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_tgt\_agg\_BOC\_logic\_check] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_tgt\_agg\_BOC\_logic\_check]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Check the control tables for validation checks that have passed

and raise an error if not

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [sp\_Ctrl\_tgt\_agg\_BOC\_logic\_check]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

--Check the control table for validation\_check = Fail

INSERT INTO app.ISSUETracer

SELECT '','','','',CTRL.TGT\_PID,

'Validation Check Fail in Ctrl\_Src\_tgt',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Ctrl\_Tgt\_Agg\_boc CTRL

where CTRL.validationcheck = 'FAIL'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertissueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Tgt\_Agg\_BOC] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Tgt\_Agg\_BOC]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: This Stored procedure will insert records into control table for all products

in report type BOC

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: [dbo].[sp\_Ctrl\_app.tgt\_app.AGG\_BOC]

\*\* Report Type: All Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.app.tgt\_unsecuredInflow') AND NAME ='IX\_app.tgt\_UnsecuredInflow')

-- DROP INDEX IX\_app.tgt\_UnsecuredInflow ON dbo.app.tgt\_unsecuredInflow;

--CREATE NONCLUSTERED INDEX IX\_app.tgt\_UnsecuredInflow

-- ON dbo.app.tgt\_unsecuredInflow(app.tgt\_unsecuredInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.app.tgt\_assetInflow') AND NAME ='IX\_app.tgt\_AssetInflow')

-- DROP INDEX IX\_app.tgt\_AssetInflow ON dbo.app.tgt\_assetInflow;

-- CREATE NONCLUSTERED INDEX IX\_app.tgt\_AssetInflow

-- ON dbo.app.tgt\_assetInflow(app.tgt\_assetInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.app.AGG\_BOC') AND NAME ='IX\_AGG\_2052a')

-- DROP INDEX IX\_app.tgt\_AssetInflow ON dbo.app.AGG\_BOC;

-- CREATE NONCLUSTERED INDEX IX\_AGG\_2052a

-- ON dbo.app.AGG\_BOC(app.AGG\_BOCID);

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.Ref\_table\_details') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON dbo.Ref\_table\_details;

-- CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

-- ON dbo.Ref\_table\_details(PID);

--Get Total rows from soure to ensure we have received all the data and get failure/success counts

BEGIN TRY

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--insert data into target table

INSERT INTO app.Ctrl\_tgt\_Agg\_BOC

(

ValidationCheck

,app.tgt\_reportScope

,app.tgt\_PID

,app.tgt\_product

,app.tgt\_maturityValue

,app.tgt\_marketValue

--,app.tgt\_NumRec

,Agg\_reportScope

,Agg\_PID

,Agg\_product

,Agg\_maturityValue

,Agg\_marketValue

--,Agg\_NumRec

)

SELECT CASE WHEN ABS(ISNULL(Tgt.MaturityValue,0)) = ISNULL(AGG.MaturityValue,0)

and ABS(ISNULL(Tgt.MarketVALUE,0)) = ISNULL(AGG.MarketVALUE,0) THEN 'Pass' Else 'Fail' END AS ValidationCheck

,TGT.reportScope

,Tgt.PID

,Tgt.product

,ISNULL(Tgt.MaturityValue,0) AS [Target MaturityValue Value]

,ISNULL(Tgt.MarketVALUE,0) AS [Target MarketValue Value]

--,Tgt.NumRec

,AGG.ReportScope

,AGG.PID

,AGG.product

,ISNULL(AGG.MaturityValue,0) AS [Aggregation MaturityValue Value]

,ISNULL(AGG.MarketVALUE,0) AS [Aggregation MarketValue Value]

--,AGG.NumRec

FROM

(

SELECT reportScope, PID, product,

0 AS MaturityValue,

CASE WHEN PID = 'I.A.3' THEN

SUM( CASE WHEN DESC1 = '0060' THEN (ABS(TGT.marketValue))

WHEN DESC1 = '9003' THEN (-1)\*(ABS(TGT.marketValuE))

END )

ELSE SUM(TGT.MARKETVALUE) END AS MarketVALUE

FROM app.tgt\_assetinflow tgt WHERE MARKETVALUE <> 0 group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product,

CASE WHEN PID = 'I.U.4'

THEN

SUM(CASE WHEN TGT.SrcID like 'FMD\_ALL | %' THEN (-1)\*TGT.maturityValue

WHEN TGT.SrcID like 'MULTI%' THEN (-1)\* TGT.maturityValue

WHEN TGT.SrcID like 'T\_FMDDATA | %' THEN (-1)\* abs(TGT.maturityValue)

END )

ELSE SUM(ABS(MATURITYvALUE)) END AS MaturityValue,

0 AS MarketVALUE

FROM app.tgt\_unsecuredinflow TGT group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityAmountCurrency1 + MaturityAmountCurrency2) AS MaturityValue,SUM(0) AS MarketVALUE

FROM app.tgt\_foreignexchange group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_securedinflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_wholesaleOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product,

CASE WHEN PID = 'O.O.4'

THEN

SUM(CASE WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_TOTAL |%' THEN ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN (-1)\* ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_OS |%' THEN ABS(TGT.maturityValue)

END )

ELSE CASE WHEN PID = 'O.O.5'

THEN

SUM(CASE WHEN TGT.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN ABS(TGT.maturityValue)

WHEN TGT.SrcID like 'FBNK\_LIMIT\_OS\_4027 |%' THEN ABS( TGT.maturityValue)

END )

ELSE SUM(MATURITYvALUE) END END

AS [MaturityValue],

0 AS MarketVALUE

FROM app.tgt\_otherOUTFLOW TGT group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, 0 AS MaturityValue,SUM(marketvalue) AS MarketVALUE

FROM app.tgt\_informationalsupplemental group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_securedOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_depositsOutflow group by reportScope, PID, product

UNION ALL

SELECT reportScope, PID, product, SUM(maturityValue) AS MaturityValue,0 AS MarketVALUE

FROM app.tgt\_otherINflow group by reportScope, PID, product

) TGT

FULL OUTER JOIN

(

SELECT reportScope, PID, product, isnull(sum(agg.maturityValue),0) AS MATURITYvALUE , ISNULL(SUM(agg.marketValue),0) AS MarketVALUE--COUNT(\*) AS NumRec

FROM app.AGG\_BOC agg

--WHERE PID = 5

GROUP BY reportScope, PID, product

) AGG

ON TGT.reportScope = AGG.ReportScope

AND SUBSTRING( tgt.PID,5,100)= agg.PID

AND tgt.product = agg.Product

where tgt.PID not in( 'S.FX.2', 'i.a.1','i.o.8')

ORDER BY TGT.PID

-- issue handling

END TRY

BEGIN CATCH

EXEC [dbo].[Proc\_InsertissueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_src\_tgt\_logic\_check] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_src\_tgt\_logic\_check]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Check the control tables for validation checks that have passed

and raise an error if not

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [sp\_Ctrl\_src\_tgt\_logic\_check]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

INSERT INTO app.ISSUETracer

SELECT '','','','',CTRL.PRODUCT,

'Validation Check Fail in Ctrl\_Src\_tgt',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Ctrl\_Src\_Tgt CTRL

where CTRL.validationcheck = 'FAIL'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Secured\_Outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Secured\_Outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_Secured\_Outflow]

\*\* Report Type: Secured Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, FMD.ASOFDATE)) = sum(Tgt\_SO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SO.Tgt\_securedoutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SO.portfolio AS [Portfolio]

,Tgt\_SO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, FMD.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SO.Tgt\_SecuredOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

-- select Tgt\_SO.reportScope, Tgt\_SO.marketValue AS marketvalue, Tgt\_SO.PID

--linked server access

FROM SRC.[T\_FMDDATA] fmd

INNER JOIN app.Tgt\_securedoutflow Tgt\_SO

ON Tgt\_SO.SrcID = CAST(fmd.N\_ID AS VARCHAR)

AND TGT\_SO.AsofDate = FMD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND REF\_TBL.Portfolio = Tgt\_SO.PORTFOLIO

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_SO.reportScope

,Tgt\_SO.PID

,Tgt\_SO.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SO.portfolio

,CreationDate

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange(RPDT.CCY,RPDT.MatProcAmt, RPDT.ASOFDATE)) = sum(Tgt\_SO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SO.portfolio AS [Portfolio]

,Tgt\_SO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'MatProcAmt' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(RPDT.CCY,RPDT.MatProcAmt, RPDT.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct RPDT.DEALNO+ RPDT.BR + CAST(RPDT.DEALTIME AS VARCHAR(50))+CAST(RPDT.YIELD\_8 AS VARCHAR(100))+CAST(RPDT.CLPRICE\_8 AS VARCHAR(100))+CAST(RPDT.DRPRICE\_8 AS VARCHAR(100)) + CAST(RPDT.ASOFDATE AS VARCHAR(100))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SO.Tgt\_SecuredOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,RPDT.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

-- select Tgt\_SO.reportScope, Tgt\_SO.marketValue AS marketvalue, Tgt\_SO.PID

--linked server access

FROM SRC.US\_OPICS\_RPDT RPDT

INNER JOIN app.Tgt\_securedoutflow Tgt\_SO

ON Tgt\_SO.SrcID = 'RPDT | '+RPDT.DEALNO+ ' | '+RPDT.BR + ' | '+CAST(RPDT.DEALTIME AS VARCHAR(50))+' | '+CAST(RPDT.YIELD\_8 AS VARCHAR(100))+' | '+CAST(RPDT.CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(RPDT.DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100))

AND TGT\_SO.AsofDate = RPDT.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND REF\_TBL.Portfolio = TGT\_SO.PORTFOLIO

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_SO.reportScope

,Tgt\_SO.PID

,Tgt\_SO.Product

,[RPDT].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SO.portfolio

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Treasury Investment

-- \*\* PID: O.S.8

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange(SPSH.CCY,SPSH.SETTAMT, SPSH.ASOFDATE)) = sum(Tgt\_SO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SO.portfolio AS [Portfolio]

,Tgt\_SO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'MatProcAmt' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(SPSH.CCY,SPSH.SETTAMT, SPSH.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct SPSH.BR+' - '+SPSH.DEALNO) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SO.Tgt\_SecuredOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_SO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_OPICS\_SPSH SPSH

INNER JOIN app.Tgt\_securedoutflow Tgt\_SO

ON SPSH.BR+' - '+SPSH.DEALNO = Tgt\_SO.SrcID

AND SPSH.SETTDATE > TGT\_SO.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND REF\_TBL.Portfolio = TGT\_SO.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_SO.reportScope

,Tgt\_SO.PID

,Tgt\_SO.Product

,Tgt\_SO.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SO.portfolio

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--Non Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: wholesale Funding Transactions

\*\* PID: O.S.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN ISNULL((

CASE WHEN NB\_OS.currency = 'USD' THEN SUM((NB\_OS.MaturityAmountOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_OS.Currency,(NB\_OS.MaturityAmountOriginalCurrency), NB\_OS.ASOFDATE))) END),0) = ISNULL(sum(Tgt\_OS.maturityValue),0)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OS.Tgt\_OtherInflowID)

--Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OS.portfolio AS [Portfolio]

,Tgt\_OS.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,CASE WHEN NB\_OS.Currency = 'USD' THEN SUM((NB\_OS.MaturityAmountOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_OS.Currency,(NB\_OS.MaturityAmountOriginalCurrency), NB\_OS.ASOFDATE))) END

AS [AggregationFieldValue]

,COUNT(distinct NB\_OS.MaturityAmountOriginalCurrency) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OS.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OS.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OS.Tgt\_securedOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,NB\_OS.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_OS NB\_OS

INNER JOIN app.Tgt\_securedoutflow Tgt\_OS

ON Tgt\_OS.SrcID = 'NB\_OS | '+CAST(NB\_OS.ID AS VARCHAR) + ' | O.S.'+ CAST(NB\_OS.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OS.ReportingEntity +' | '+cast(NB\_OS.asofdate as VARCHAR)

-- To be uncommented after date issue is fixed

AND NB\_OS.ASOFDATE = TGT\_OS.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OS.reportScope

AND NB\_OS.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OS.PID

--To be uncommented after portfolio field is populated in source tables

AND REF\_tbl.portfolio = Tgt\_OS.Portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_OS.reportScope

,NB\_OS.currency

,Tgt\_OS.PID

,Tgt\_OS.Product

,NB\_OS.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OS.portfolio

,CreationDate

END TRY

BEGIN CATCH

-- Execute ISSUE handling stroed procedure to get details about the ISSUEs

EXEC Proc\_InsertISSUEDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Secured\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Secured\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_Secured\_Inflow]

\*\* Report Type: Secured Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.Tgt\_assetInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

ON app.Tgt\_assetInflow(TGT\_assetInflowID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

--DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.Tgt\_securedInflow') AND NAME ='IX\_Tgt\_securedInflow')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_securedInflow

ON APP.Tgt\_securedInflow(TGT\_securedInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_T24\_MULTI\_LD\_BALANCE') AND NAME ='IX\_bal')

-- DROP INDEX IX\_bal ON SRC.US\_T24\_MULTI\_LD\_BALANCE;

CREATE NONCLUSTERED INDEX IX\_bal

ON src.US\_T24\_MULTI\_LD\_BALANCE(id, date);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.TMP\_MG\_Mortgage\_Schedule') AND NAME ='IX\_TMP')

-- DROP INDEX IX\_TMP ON SRC.TMP\_MG\_Mortgage\_Schedule;

CREATE NONCLUSTERED INDEX IX\_TMP

ON src.TMP\_MG\_Mortgage\_Schedule(CONTRACT\_NO, DUE\_DATE);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

-- DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_FR2052A\_REF\_TableDetails') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON src.US\_FR2052A\_REF\_TableDetails;

CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

ON src.US\_FR2052A\_REF\_TableDetails(PID);

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, FMD.ASOFDATE)) = sum(Tgt\_SI.maturityValue)

AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'RPDT - DealNo+BR+DEALTIME+YIELD\_8+SECL-CSG\_PRICE'

AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(fmd.Currency,fmd.Deal\_Balance, FMD.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.Tgt\_SecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

-- select Tgt\_SI.reportScope, Tgt\_SI.marketValue AS marketvalue, Tgt\_SI.PID

--linked server access

FROM SRC.[T\_FMDDATA] fmd

INNER JOIN APP.Tgt\_securedInflow tgt\_SI

ON Tgt\_SI.SrcID = CAST(fmd.N\_ID AS VARCHAR)

AND tgt\_SI.AsofDate = fmd.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

and ref\_tbl.portfolio =tgt\_si.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,Tgt\_SI.maturityValue

,creationdate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: I.S.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange(RPDT.CCY,RPDT.MatProcAmt, RPDT.ASOFDATE)) = sum(Tgt\_SI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(RPDT.CCY,RPDT.MatProcAmt, RPDT.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct DEALNO+ rpdt.BR + CAST(DEALTIME AS VARCHAR(50))+CAST(YIELD\_8 AS VARCHAR(100))+CAST(CLPRICE\_8 AS VARCHAR(100))+CAST(DRPRICE\_8 AS VARCHAR(100)) + CAST(RPDT.ASOFDATE AS VARCHAR(100))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.Tgt\_SecuredInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,RPDT.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

-- select Tgt\_SI.reportScope, Tgt\_SI.marketValue AS marketvalue, Tgt\_SI.PID

--linked server access

FROM SRC.US\_OPICS\_RPDT RPDT

INNER JOIN APP.Tgt\_securedInflow tgt\_SI

ON Tgt\_SI.SrcID = 'RPDT | '+DEALNO+' | '+ rpdt.BR + ' | '+CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100))

and Tgt\_SI.asofdate = rpdt.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

and ref\_tbl.portfolio =tgt\_si.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,[RPDT].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,Tgt\_SI.maturityValue

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.S.7 Loan Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.principal), bal.ASOFDATE)) = sum(Tgt\_SI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.principal), bal.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct (bal.ID + ' | '+bal.[DATE]))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.TGT\_securedInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,bal.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE bal

INNER JOIN SRC.MULTI\_LD LD

ON LD.ID = bal.ID

AND LD.ASOFDATE = bal.ASOFDATE

--and bal.SCH\_TYPE = 'P'

--AND bal.AMOUNT <>' 0'

--AND bal.AMOUNT IS NOT NULL

INNER JOIN

APP.Tgt\_securedInflow tgt\_SI

ON tgt\_SI.SrcID = 'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ bal.ID+' | '+BAL.[DATE]

and tgt\_si.AsofDate = bal.asofdate

and Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) > tgt\_si.asofdate

and tgt\_si.PID = 'I.S.7'

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

AND ref\_tbl.Portfolio = Tgt\_SI.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,bal.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,ld.currency

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.S.7 Loan Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.Interest), bal.ASOFDATE)) = sum(Tgt\_SI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(LD.CURRENCY,(bal.Interest), bal.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct (bal.ID + ' | '+bal.[DATE]))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.TGT\_securedInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,bal.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE bal

INNER JOIN SRC.MULTI\_LD LD

ON LD.ID = bal.ID

AND LD.ASOFDATE = bal.ASOFDATE

--and bal.SCH\_TYPE = 'P'

--AND bal.AMOUNT <>' 0'

--AND bal.AMOUNT IS NOT NULL

INNER JOIN

APP.Tgt\_securedInflow tgt\_SI

ON tgt\_SI.SrcID = 'US\_T24\_MULTI\_LD\_BALANCE | Interest | '+ bal.ID+' | '+BAL.[DATE]

and tgt\_si.AsofDate = bal.asofdate

and Convert(Varchar(20),CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE),101) > tgt\_si.asofdate

and tgt\_si.PID = 'I.S.7'

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

AND ref\_tbl.Portfolio = Tgt\_SI.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,bal.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,ld.currency

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.S.7 Mortgage Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(TMP.CURRENCY,(TMP.PRINCIPAL\_amount), TMP.ASOFDATE) )= sum(Tgt\_SI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(TMP.CURRENCY,(TMP.PRINCIPAL\_amount), TMP.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct (TMP.CONTRACT\_NO+'\_'+TMP.DUE\_DATE))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.TGT\_securedInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TMP.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

FROM src.TMP\_MG\_Mortgage\_Schedule TMP

INNER JOIN src.Multi\_MG\_MORTGAGE mg

ON mg.id = TMP.CONTRACT\_NO

AND mg.ASOFDATE = TMP.ASOFDATE

INNER JOIN APP.Tgt\_securedInflow tgt\_SI

ON 'TMP\_MG\_Mortgage\_Schedule | Principal | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE= tgt\_SI.[SrcID]

and tmp.DUE\_DATE > tgt\_si.AsofDate

AND tgt\_SI.AsofDate = TMP.ASOFDATE

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

AND ref\_tbl.Portfolio = Tgt\_SI.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,TMP.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.S.7 Mortgage Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

sum(dbo.fn\_CurrencyExchange

(TMP.CURRENCY,(TMP.INTEREST\_AMOUNT), TMP.ASOFDATE) )= sum(Tgt\_SI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_SI.portfolio AS [Portfolio]

,Tgt\_SI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange

(TMP.CURRENCY,(TMP.INTEREST\_AMOUNT), TMP.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct (TMP.CONTRACT\_NO+'\_'+TMP.DUE\_DATE))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_SI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_SI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_SI.TGT\_securedInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TMP.ASOFDATE AS [AsOfDate]

,creationdate AS [CreationDate]

FROM src.TMP\_MG\_Mortgage\_Schedule TMP

INNER JOIN src.Multi\_MG\_MORTGAGE mg

ON mg.id = TMP.CONTRACT\_NO

AND mg.ASOFDATE = TMP.ASOFDATE

INNER JOIN APP.Tgt\_securedInflow tgt\_SI

ON 'TMP\_MG\_Mortgage\_Schedule | Interest | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE= tgt\_SI.[SrcID]

and tmp.DUE\_DATE > tgt\_si.AsofDate

AND tgt\_SI.AsofDate = TMP.ASOFDATE

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_SI.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

AND ref\_tbl.Portfolio = Tgt\_SI.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_SI.reportScope

,Tgt\_SI.PID

,Tgt\_SI.Product

,TMP.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_SI.portfolio

,creationdate

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Other\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Other\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_other\_outflow]

\*\* Report Type: other Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange(SCHD.CCY,SCHD.IntAmt, SCHD.ASOFDATE)) = sum(Tgt\_OO.maturityValue)

AND COUNT(distinct SCHD.DEALNO+SCHD.BR+CAST(SCHD.INTENDDTE AS VARCHAR)) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(SCHD.CCY,SCHD.IntAmt, SCHD.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct SCHD.DEALNO+SCHD.BR+CAST(SCHD.INTENDDTE AS VARCHAR(50))+SCHD.SCHDSEQ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,SCHD.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

-- select Tgt\_OO.reportScope, Tgt\_OO.marketValue AS marketvalue, Tgt\_OO.PID

--linked server access

FROM SRC.[US\_OPICS\_SCHD] SCHD

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = 'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ

AND TGT\_OO.AsofDate = SCHD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.Portfolio

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,SCHD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Portfolio: Wholesale Funding Transactions

PID: O.O.19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE

WHEN (

ISNULL(SUM(DBO.fn\_CurrencyExchange

(MM.Currency,(mm.Tot\_interest\_amt), MM.ASOFDATE)),0) = ISNULL(sum(Tgt\_OO.maturityValue),0)

-- AND COUNT(distinct LIM.ID) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange

(MM.Currency,(mm.Tot\_interest\_amt), MM.ASOFDATE))

AS [AggregationFieldValue]

,COUNT('T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR) ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,MM.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--,tgt\_OO.PID

-- select Tgt\_OO.reportScope, Tgt\_OO.marketValue AS marketvalue, Tgt\_OO.PID

--linked server access

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.[Multi\_MM] mm

ON fmd.CONTACT\_ID = mm.CONTRACT\_NO

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = 'T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR)

AND TGT\_OO.AsofDate = FMD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_OO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.Portfolio

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,MM.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Portfolio: Wholesale Funding Transactions

PID: O.O.19 BOND

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE

WHEN (

SUM((DBO.fn\_CurrencyExchange

(TPOS.CCY,SECS.INTPAYAMT\_8, SECS.ASOFDATE))\*TPOS.QTY) = sum(Tgt\_OO.maturityValue)

-- AND COUNT(distinct SCHD.DEALNO+SCHD.BR+CAST(SCHD.INTENDDTE AS VARCHAR)) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM((DBO.fn\_CurrencyExchange

(TPOS.CCY,SECS.INTPAYAMT\_8, tpos.ASOFDATE))\*TPOS.QTY)

AS [AggregationFieldValue]

,COUNT(distinct LTRIM(RTRIM('TPOS | '+LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TPOS.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--select SUM(maturityvalue), SUM((DBO.fn\_CurrencyExchange

--(TPOS.CCY,SECS.INTPAYAMT\_8, tpos.ASOFDATE))\*TPOS.QTY)

FROM SRC.[US\_OPICS\_TPOS] TPOS

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = LTRIM(RTRIM('TPOS | '+LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50))))

AND TGT\_OO.AsofDate = TPOS.ASOFDATE

INNER JOIN SRC.US\_OPICS\_SECS SECS

ON SECS.SECID = TPOS.SECID

AND TPOS.ASOFDATE = SECS.ASOFDATE

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OO.reportScope

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.Portfolio

WHERE tgt\_oo.PID = 'O.O.19'

AND TGT\_OO.Portfolio = 'WHOLESALE Funding Transactions'

and DATEDIFF (DAY, SECS.ASOFDATE, SECS.IPAYDATE) BETWEEN 0 AND 30

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,TPOS.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Portfolio: Credit and Liquidity Facility

PID: O.O.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, LIMIT.ASOFDATE))=sum(Tgt\_OO.maturityValue)

-- AND COUNT(distinct LIM.ID) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, LIMIT.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(LIMIT.CREDIT\_LINE) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,LIMIT.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--,tgt\_OO.PID

-- select Tgt\_OO.reportScope, Tgt\_OO.marketValue AS marketvalue, Tgt\_OO.PID

--linked server access

FROM SRC.FBNK\_LIMIT LIMIT

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON (Tgt\_OO.SrcID = 'FBNK\_LIMIT\_AVAIL\_TOTAL | '+LIMIT.[CREDIT\_LINE]

or (Tgt\_OO.SrcID = 'FBNK\_LIMIT\_AVAIL\_4027 | '+LIMIT.[CREDIT\_LINE\_NO]))

AND TGT\_OO.AsofDate = LIMIT.ASOFDATE

AND Tgt\_OO.PID in ( 'O.O.4','o.o.5')

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_OO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.Portfolio

WHERE LIMIT.RECORD\_PARENT IS NULL

AND LIMIT.COMMITTED\_YN = 'Y'

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,limIT.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives

\*\* PID: O.O.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange(M\_LD.Currency, BAL.INTEREST,tgt\_oo.asofdate)) = ISNULL(sum(Tgt\_OO.maturityValue),0)

-- AND COUNT(distinct LIM.ID) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(M\_LD.Currency, BAL.INTEREST,tgt\_oo.asofdate)) AS [AggregationFieldValue]

,COUNT(distinct M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_OO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--,tgt\_OO.PID

FROM SRC.MULTI\_LD M\_LD

LEFT JOIN SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

ON M\_LD.ID = BAL.ID

AND M\_LD.CO\_CODE = BAL.CO\_CODE

AND M\_LD.ASOFDATE = BAL.ASOFDATE

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID = Tgt\_OO.SrcID

and m\_ld.ASOFDATE = tgt\_oo.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OO.reportScope

and ref\_en.Company\_code = m\_ld.co\_Code

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.PORTFOLIO

WHERE TGT\_OO.PID = 'O.O.1'

AND BAL.DATE > BAL.ASOFDATE

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,Tgt\_OO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Trade Finance

\*\* PID: O.O.7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), fmd.asofdate)) = ISNULL(sum(Tgt\_OO.maturityValue),0)

-- AND COUNT(distinct LIM.ID) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), fmd.asofdate))

AS [AggregationFieldValue]

,COUNT('T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR) ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--,tgt\_OO.PID

-- select Tgt\_OO.reportScope, Tgt\_OO.marketValue AS marketvalue, Tgt\_OO.PID

--linked server access

FROM SRC.T\_FMDDATA FMD

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = cast(fmd.[N\_ID] as varchar)

AND TGT\_OO.AsofDate = FMD.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = Tgt\_OO.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_OO.PID

AND REF\_TBL.Portfolio = TGT\_OO.Portfolio

GROUP BY ref\_en.ReportingEntityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,fmd.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--Non Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Other Payables and Margin Collateral

\*\* PID: O.O.2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

CASE

WHEN (

CASE WHEN NB\_OO.currency = 'USD' THEN SUM((NB\_OO.MaturityAmountOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_OO.Currency,(NB\_OO.MaturityAmountOriginalCurrency), TGT\_OO.ASOFDATE))) END = sum(Tgt\_OO.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OO.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,CASE WHEN NB\_OO.Currency = 'USD' THEN SUM((NB\_OO.MaturityAmountOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_OO.Currency,(NB\_OO.MaturityAmountOriginalCurrency), TGT\_OO.ASOFDATE))) END

AS [AggregationFieldValue]

,COUNT(distinct NB\_OO.MaturityAmountOriginalCurrency) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,NB\_OO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_OO NB\_OO

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = 'NB\_OO | '+CAST(NB\_OO.ID AS VARCHAR) + ' | O.O.'+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OO.ReportingEntity +' | '+cast(NB\_OO.asofdate as VARCHAR)

and NB\_OO.ASOFDATE = Tgt\_OO.asofdate

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_OO.Reportingentity

AND NB\_OO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND REF\_tbl.portfolio = Tgt\_OO.Portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_OO.reportScope

,NB\_OO.currency

,Tgt\_OO.PID

,Tgt\_OO.Product

,NB\_OO.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,CreationDate

UNION

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Operational Deposits

\*\* PID: O.O.19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE WHEN X.AggregationFieldGroupByValue = X.TargetFieldGroupByValue THEN 'Pass' ELSE 'Fail' END AS [Validation\_Check]

, Portfolio

,Product

,src\_Table\_Name

,AggregationFieldGroupBy

,AggregationFieldGroupByValue

,AggregationField

,[AggregationFieldValue]

,#ofRecordsFromSrc

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetField]

,[TargetFieldValue]

,[#ofRecordsFromTgt]

,[TargetTable]

,[AsOfDate]

,[CreationDate]

FROM(

SELECT

CASE

WHEN (

CAST(ISNULL(SUM(dbo.fn\_CurrencyExchange(TPOS.CCY, CAST(SECS.INTPAYAMT\_8 AS DECIMAL(30,15)) , TPOS.ASOFDATE) \* CAST(TPOS.QTY AS DECIMAL(30,15))),0) AS DECIMAL(30,15)) = CAST(ISNULL(sum(CAST(Tgt\_OO.maturityValue AS DECIMAL (30,15))),0) AS DECIMAL (30,15))

-- AND COUNT(distinct LIM.ID) = COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OO.portfolio AS [Portfolio]

,Tgt\_OO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,CAST((SUM(dbo.fn\_CurrencyExchange(TPOS.CCY, CAST(SECS.INTPAYAMT\_8 AS DECIMAL(30,15)), TPOS.ASOFDATE)\* CAST(TPOS.QTY AS DECIMAL(30,15)))) AS DECIMAL (30,15)) AS [AggregationFieldValue]

,COUNT(distinct 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))

+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))

+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))

+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OO.Tgt\_otherOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_OO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--,tgt\_OO.PID

FROM SRC.US\_OPICS\_TPOS TPOS

INNER JOIN app.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))

INNER JOIN SRC.US\_OPICS\_SECS SECS

ON TPOS.SECID = SECS.SECID

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_OO.reportScope

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_OO.PID

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_oo.Portfolio))

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OO.reportScope

,Tgt\_OO.PID

,Tgt\_OO.Product

,Tgt\_OO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OO.portfolio

,Tgt\_OO.maturityValue

,CreationDate )X

END TRY

BEGIN CATCH

-- Execute ISSUE handling stroed procedure to get details about the ISSUEs

EXEC Proc\_InsertISSUEDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Other\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Other\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_Other\_Inflow]

\*\* Report Type: Other Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO app.[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[creationdate]

)

SELECT DISTINCT

CASE

WHEN (

CASE WHEN fmd.currency = 'USD' THEN CAST(SUM(fmd.deal\_balance) AS DECIMAL(30,15))

ELSE SUM((DBO.fn\_CurrencyExchange

(fmd.Currency,CAST((fmd.deal\_balance) AS DECIMAL(30,15)), fmd.ASOFDATE))) END = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,CASE WHEN fmd.currency = 'USD' THEN CAST(SUM(fmd.deal\_balance) AS DECIMAL(30,15))

ELSE SUM((DBO.fn\_CurrencyExchange

(fmd.Currency,CAST((fmd.deal\_balance) AS DECIMAL(30,15)), fmd.ASOFDATE))) END

AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

-- select Tgt\_OI.reportScope, Tgt\_OI.marketValue AS marketvalue, Tgt\_OI.PID

--linked server access

FROM src.[T\_FMDDATA] fmd

INNER JOIN app.Tgt\_otherInflow tgt\_OI

ON Tgt\_OI.SrcID = CAST(fmd.N\_ID AS VARCHAR(10))

AND FMD.ASOFDATE = TGT\_oi.ASOFDATE

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = tgt\_OI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OI.reportScope

,fmd.currency

,Tgt\_OI.PID

,Tgt\_OI.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Wholesale Funding Transactions

\*\* PID: TSD - I.O.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(CASE WHEN SCHD.CCY = 'USD' THEN (SCHD.IntAmt)

ELSE (DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.IntAmt), SCHD.ASOFDATE)) END ) = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(CASE WHEN SCHD.CCY = 'USD' THEN (SCHD.IntAmt)

ELSE (DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.PrinAmt), SCHD.ASOFDATE)) END )

AS [AggregationFieldValue]

,COUNT(distinct DEALNO+BR+CAST(INTENDDTE AS VARCHAR(50))+SCHDSEQ ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.TGT\_otherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,SCHD.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

FROM

src.US\_OPICS\_SCHD SCHD

INNER JOIN app.Tgt\_otherInflow tgt\_OI

ON tgt\_OI.SrcID = 'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ

AND tgt\_OI.AsofDate = SCHD.ASOFDATE

and tgt\_OI.PID ='i.o.6'

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.reportingentityDescription = tgt\_OI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OI.PID

AND ref\_tbl.Portfolio = tgt\_OI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OI.reportScope

,Tgt\_OI.PID

,Tgt\_OI.Product

,SCHD.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investments

\*\* PID: I.O.6 MIS\_DW

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UNION

SELECT DISTINCT

CASE

WHEN (

SUM((DBO.fn\_CurrencyExchange (TPOS.CCY,(SECS.INTPAYAMT\_8 \* TPOS.QTY), TPOS.ASOFDATE))) = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'INTPAYAMT\_8' AS [AggregationField]

,SUM((DBO.fn\_CurrencyExchange (TPOS.CCY,(SECS.INTPAYAMT\_8 \* TPOS.QTY), TPOS.ASOFDATE))) AS [AggregationFieldValue]

,COUNT(distinct LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(TPOS.SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50)))) ) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.TGT\_otherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TPOS.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

FROM

src.US\_OPICS\_TPOS TPOS

INNER JOIN app.Tgt\_otherInflow tgt\_OI

ON CAST(tgt\_OI.SrcID AS VARCHAR (MAX)) = LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(TPOS.SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50))))

AND tgt\_OI.AsofDate = TPOS.ASOFDATE

AND tgt\_OI.PID ='I.O.6'

AND Tgt\_OI.Portfolio = 'Treasury Investment'

INNER JOIN src.US\_OPICS\_SECS SECS

ON SECS.SECID = TPOS.SECID

AND SECS.ASOFDATE = Tgt\_OI.AsofDate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON Ref\_en.ReportingEntityDescription = tgt\_OI.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OI.PID

AND ref\_tbl.Portfolio = tgt\_OI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.ReportingEntitY

,Tgt\_OI.reportScope

,Tgt\_OI.PID

,Tgt\_OI.Product

,TPOS.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

--Execute the next procedures to migrate data to Target table for the enxt Report Type - Unsecured Inflow

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives Receivables, Treasury Investment

\*\* PID: I.O.1, I.o.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM(DBO.fn\_CurrencyExchange

(M\_LD.Currency,(MLSDD.INTEREST), tgt\_oi.ASOFDATE )) = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE 'Fail'

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange

(M\_LD.Currency,(MLSDD.INTEREST), tgt\_oi.ASOFDATE ))

AS [AggregationFieldValue]

,COUNT(distinct M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID)AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,tgt\_oi.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

FROM src.[MULTI\_LD] M\_LD

LEFT JOIN [SRC].[US\_T24\_MULTI\_LD\_BALANCE] MLSDD

ON M\_LD.ID = MLSDD.ID

AND M\_LD.ASOFDATE = MLSDD.ASOFDATE

LEFT JOIN app.Tgt\_otherInflow tgt\_OI

ON Tgt\_OI.SrcID = CAST(M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS VARCHAR(100))

AND Tgt\_OI.AsofDate = M\_LD.asofdate

--get Reporting Entity

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OI.reportScope

and ref\_en.company\_code = m\_ld.co\_code

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = tgt\_OI.Portfolio

--AND ref\_tbl.pid = fmd.PID

where mlsdd.DATE > MLSDD.asofdate

and tgt\_oi.pid in ('I.O.1','I.O.6')

GROUP BY ref\_en.reportingentityDescription

,Tgt\_OI.reportScope

--,M\_LD.currency

,Tgt\_OI.PID

--,Tgt\_OI.Product

,tgt\_oi.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Commercial and Retail Loans

\*\* PID: I.O.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID(N'tempdb..#Temp') IS NOT NULL

BEGIN

DROP TABLE #Temp

END

IF OBJECT\_ID(N'tempdb..#Temp2') IS NOT NULL

BEGIN

DROP TABLE #Temp2

END

CREATE TABLE #TEMP

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,REVIEW\_FREQUENCY VARCHAR(100)

,STARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

CREATE TABLE #TEMP2

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,STARTDATE DATE

,EFFECTIVESTARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

INSERT INTO #TEMP

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,REVIEW\_FREQUENCY

,STARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,LIMIT\_CURRENCY

--,REVIEW\_FREQUENCY

,(CAST(AVAIL\_AMT AS DECIMAL(30,15)) \* (CAST(SUBSTRING(CUSIP\_NO,1,CHARINDEX('%',CUSIP\_NO)-1) AS DECIMAL(30,15))/100))/(12/CAST(LEFT(SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) AS DECIMAL )) AS AMOUNT

,REVIEW\_FREQUENCY

,LEFT(REVIEW\_FREQUENCY, 8) AS STARTDATE

,EXPIRY\_DATE

,CAST(round((DATEDIFF(MONTH,cast(LEFT(REVIEW\_FREQUENCY,8) as DATE),CAST(EXPIRY\_DATE AS DATE)))/cast(LEFT(SUBSTRING (REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) as decimal)+0.4,0) AS INTEGER) AS FREQUENCY

,LIMIT.CO\_CODE

,ASOFDATE

FROM SRC.FBNK\_LIMIT LIMIT

WHERE AVAIL\_AMT NOT LIKE '%]%'

AND CUSIP\_NO IS NOT NULL

AND CUSIP\_NO <> '0'

AND AVAIL\_AMT <> '0'

AND AVAIL\_AMT IS NOT NULL

INSERT INTO #TEMP2

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,DATEADD(MONTH, (B.RN-1)\* LEFT((SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY))),2),STARTDATE) AS EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

FROM #TEMP A

CROSS JOIN (SELECT TOP (SELECT ISNULL(MAX(FREQUENCY)+1,0) FROM #TEMP) rn = ROW\_NUMBER()

OVER (ORDER BY [object\_id])

FROM sys.all\_columns

ORDER BY [object\_id])B

WHERE B.RN < = A.FREQUENCY

ORDER BY CREDIT\_LINE\_NO

INSERT INTO app.[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[creationdate]

)

SELECT DISTINCT

CASE

WHEN (

SUM(CASE WHEN LIM.CURRENCY = 'USD' THEN (LIM.amount)

ELSE (dbo.fn\_CurrencyExchange

(LIM.CURRENCY,(LIM.amount), LIM.ASOFDATE)) END ) = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'PrinAmt' AS [AggregationField]

,SUM(CASE WHEN LIM.CURRENCY = 'USD' THEN (LIM.amount)

ELSE (dbo.fn\_CurrencyExchange

(LIM.CURRENCY,(LIM.amount), LIM.ASOFDATE)) END )

AS [AggregationFieldValue]

,COUNT(distinct lim.CREDIT\_LINE\_NO + CAST(EFFECTIVESTARTDATE AS VARCHAR))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.Tgt\_otherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,LIM.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

FROM #TEMP2 LIM

INNER JOIN

app.Tgt\_otherInflow Tgt\_OI

ON Tgt\_OI.SrcID = lim.CREDIT\_LINE\_NO + CAST(EFFECTIVESTARTDATE AS VARCHAR)

AND Tgt\_OI.AsofDate = LIM.ASOFDATE

and tgt\_OI.PID = 'I.O.9'

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_OI.reportScope

-- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.portfolio

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_OI.reportScope

,Tgt\_OI.PID

,Tgt\_OI.Product

,LIM.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

,creationdate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* NON BRANCH

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other payables and margin collaterals

-- \*\* PID: I.O.2, I.O.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[creationdate]

)

SELECT DISTINCT

CASE

WHEN (

SUM((dbo.fn\_CurrencyExchange

(NB\_IO.OriginalCurrencyOfMaturityAmount,(NB\_IO.MaturityAmountOriginalCurrency), TGT\_OI.ASOFDATE))) = sum(Tgt\_OI.maturityValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_OI.portfolio AS [Portfolio]

,Tgt\_OI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,Ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange

(NB\_IO.OriginalCurrencyOfMaturityAmount,(NB\_IO.MaturityAmountOriginalCurrency), TGT\_OI.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct NB\_IO.MaturityAmountOriginalCurrency)AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_OI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_OI.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_OI.Tgt\_OtherInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,NB\_IO.ASOFDATE AS [AsOfDate]

,creationdate AS [creationdate]

FROM src.US\_FR2052A\_NB\_IO NB\_IO

INNER JOIN app.Tgt\_otherInflow Tgt\_OI

ON Tgt\_OI.SrcID = 'NB\_IO | '+CAST(NB\_IO.ID AS VARCHAR) + ' | I.O.'+ CAST(NB\_IO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IO.ReportingEntity +' | '+cast(NB\_IO.asofdate as VARCHAR)

and NB\_IO.ASOFDATE = Tgt\_OI.asofdate

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IO.Reportingentity

AND NB\_IO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

---- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND REF\_tbl.portfolio = Tgt\_OI.Portfolio

GROUP BY

Ref\_en.reportingentityDescription

,Tgt\_OI.reportScope

,Tgt\_OI.PID

,Tgt\_OI.Product

,NB\_IO.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_OI.portfolio

,creationdate

END TRY

BEGIN CATCH

-- Execute ISSUE handling stroed procedure to get details about the ISSUEs

EXEC Proc\_InsertISSUEDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Informational\_Supplemental] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Informational\_Supplemental]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_deposits\_outflow]

\*\* Report Type: other Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

----insert data into target table

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

CASE WHEN NB\_SI.CURRENCY= 'USD' THEN SUM((NB\_SI.MarketValueOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_SI.Currency,(NB\_SI.MarketValueOriginalCurrency), Tgt\_IS.ASOFDATE))) END = sum(Tgt\_IS.marketValue)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_IS.Tgt\_OtherInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_IS.portfolio AS [Portfolio]

,Tgt\_IS.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,CASE WHEN NB\_SI.Currency = 'USD' THEN SUM((NB\_SI.MarketValueOriginalCurrency))

ELSE SUM((dbo.fn\_CurrencyExchange

(NB\_SI.Currency,(NB\_SI.MarketValueOriginalCurrency), Tgt\_IS.ASOFDATE))) END

AS [AggregationFieldValue]

,COUNT(distinct NB\_SI.MarketValueOriginalCurrency) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_IS.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_IS.marketValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_IS.Tgt\_informationalSupplementalID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,NB\_SI.ASOFDATE AS [AsOfDate]

,Creationdate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN APP.Tgt\_informationalSupplemental Tgt\_IS

ON Tgt\_IS.SrcID = 'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.'+ CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+cast(NB\_SI.asofdate as VARCHAR)

and NB\_SI.ASOFDATE = Tgt\_IS.asofdate

--get Reporting Entity

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_IS.PID

AND REF\_tbl.portfolio = Tgt\_IS.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_IS.reportScope

,NB\_SI.currency

,Tgt\_IS.PID

,Tgt\_IS.Product

,NB\_SI.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_IS.portfolio

,Creationdate

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_ForeignExchange] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_ForeignExchange]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_ForeignExchange]

\*\* Report Type: Secured Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

SUM(dbo.fn\_CurrencyExchange(F\_REVAL.CCY,F\_REVAL.CCYAMT, F\_REVAL.[ASOFDATE] )) = sum(Tgt\_Fx.MaturityAmountCurrency1)

-- AND COUNT(distinct fmd.n\_id) = COUNT(distinct Tgt\_SO.Tgt\_securedInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_Fx.portfolio AS [Portfolio]

,Tgt\_Fx.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'MatProcAmt' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(F\_REVAL.CCY,F\_REVAL.CCYAMT, F\_REVAL.[ASOFDATE] ))

AS [AggregationFieldValue]

,COUNT(distinct F\_REVAL.BR+F\_REVAL.DEALNO) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_Fx.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_Fx.MaturityAmountCurrency1) AS [TargetFieldValue]

,COUNT(distinct Tgt\_Fx.SrcID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,F\_REVAL.[ASOFDATE] AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM src.[US\_OPICS\_FXDH\_REVAL] F\_REVAL

INNER JOIN APP.Tgt\_ForeignExchange Tgt\_Fx -- requires update

ON Tgt\_Fx.SrcID = F\_REVAL.BR+F\_REVAL.DEALNO

and tgt\_fx.AsofDate = F\_REVAL.ASOFDATE

--get Reporting Entity

left JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = Tgt\_Fx.reportScope

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_Fx.PID

and ref\_tbl.Portfolio = tgt\_fx.portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,Tgt\_Fx.reportScope

,Tgt\_Fx.PID

,Tgt\_Fx.Product

,F\_REVAL.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_Fx.portfolio

,CreationDate

--,Tgt\_Fx.MaturityAmountCurrency1

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_deposits\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_deposits\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_deposits\_outflow]

\*\* Report Type: other Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

----insert data into target table

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--\*\* Report Type: OD2,3,4,5and 11 (Interest)

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

ISNULL(SUM(dbo.fn\_CurrencyExchange(fmd.Currency,MM.TOT\_INTEREST\_AMT, fmd.ASOFDATE)), 0) = ISNULL(SUM(Tgt\_DO.maturityValue),0)

-- AND COUNT(distinct 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ('Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(fmd.Currency,MM.TOT\_INTEREST\_AMT, MM.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct MM.ID+'|'+MM.CONTRACT\_NO+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100)))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.[Multi\_MM] MM

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))

INNER JOIN SRC.T\_FMDDATA FMD --get Reporting Entity

ON fmd.contact\_ID = MM.contract\_no

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND FMD.company\_code = MM.co\_code

AND FMD.ASOFDATE = MM.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

AND REF\_en.Company\_name = tgt\_do.CompanyName

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'Multi\_mm'

AND LTRIM(RTRIM(REF\_TBL.Portfolio)) = LTRIM(RTRIM(TGT\_DO.Portfolio))

WHERE Tgt\_DO.PID IN ('O.D.2','O.D.3') AND ref\_tbl.SourceTableName = 'Multi\_mm'

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

--,Tgt\_DO.maturityValue

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--\*\* Report Type: OD4,5 (Interest)

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

ISNULL(SUM(MM.TOT\_INTEREST\_AMT), 0) = ISNULL(SUM(Tgt\_DO.orig\_maturityValue),0)

-- AND COUNT(distinct 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ('Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(MM.TOT\_INTEREST\_AMT)

AS [AggregationFieldValue]

,COUNT(distinct 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100)))AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.orig\_maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.[Multi\_MM] MM

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))

INNER JOIN SRC.T\_FMDDATA FMD --get Reporting Entity

ON fmd.contact\_ID = MM.contract\_no

AND FMD.Customer\_no = MM.CUSTOMER\_ID

AND FMD.company\_code = MM.co\_code

AND FMD.ASOFDATE = MM.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

AND ref\_en.Company\_name = Tgt\_do.CompanyName

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'Multi\_mm'

AND LTRIM(RTRIM(REF\_TBL.Portfolio)) = LTRIM(RTRIM(TGT\_DO.Portfolio))

WHERE Tgt\_DO.PID IN ( 'O.D.4', 'O.D.5')

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

--,Tgt\_DO.maturityValue

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: O.D.7: Non-Reciprocal Brokered Accounts

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT

CASE

WHEN (

ISNULL(SUM(DBO.fn\_CurrencyExchange(TPOS.CCY,TPOS.PRINAMT, TPOS.ASOFDATE)), 0) = ISNULL(sum(Tgt\_DO.maturityValue),0)

-- AND COUNT(distinct MM.MULTI\_CONCATE\_KEY) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(DBO.fn\_CurrencyExchange(TPOS.CCY,TPOS.PRINAMT, TPOS.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'

+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))

+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'

+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_DO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_OPICS\_TPOS TPOS

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100))) AND Tgt\_DO.AsofDate = TPOS.ASOFDATE

AND Tgt\_DO.AsofDate = TPOS.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

AND ref\_en.BranchName = Tgt\_do.CompanyName

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND LTRIM(RTRIM(REF\_TBL.Portfolio)) = LTRIM(RTRIM(TGT\_DO.Portfolio))

WHERE Tgt\_DO.PID IN ('O.D.7')

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,Tgt\_DO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: O.D.1-13 Principal Amount

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

ISNULL(SUM(DBO.fn\_CurrencyExchange(FMD.Currency,fmd.deal\_balance, CONVERT(VARCHAR(10),FMD.ASOFDATE,101))), 0) = ISNULL(sum(Tgt\_DO.maturityValue),0)

-- AND COUNT(distinct MM.MULTI\_CONCATE\_KEY) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(FMD.Currency,fmd.deal\_balance, CONVERT(VARCHAR(10),FMD.ASOFDATE,101)))

AS [AggregationFieldValue]

,COUNT(distinct FMD.N\_ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_DO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.T\_FMDDATA FMD

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = CAST(FMD.N\_ID AS VARCHAR (MAX))

AND Tgt\_DO.AsofDate = FMD.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

AND ref\_en.Company\_name = Tgt\_do.CompanyName

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'T\_FMDDATA'

AND LTRIM(RTRIM(REF\_TBL.Portfolio)) = LTRIM(RTRIM(TGT\_DO.Portfolio))

WHERE Tgt\_DO.PID IN ('O.D.1'

,'O.D.10'

,'O.D.11'

,'O.D.12'

,'O.D.13'

,'O.D.2'

,'O.D.3'

,'O.D.6'

,'O.D.7'

,'O.D.8'

,'O.D.9'

)

AND SourceTableName = 'T\_FMDDATA'

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,Tgt\_DO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: O.D.1-13 Principal Amount

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

ISNULL(SUM(fmd.deal\_balance), 0) = ISNULL(sum(Tgt\_DO.orig\_maturityValue),0)

-- AND COUNT(distinct MM.MULTI\_CONCATE\_KEY) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(FMD.Currency,fmd.deal\_balance, CONVERT(VARCHAR(10),FMD.ASOFDATE,101)))

AS [AggregationFieldValue]

,COUNT(distinct FMD.N\_ID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.orig\_maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_DO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

FROM SRC.T\_FMDDATA FMD

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = CAST(FMD.N\_ID AS VARCHAR (MAX))

AND Tgt\_DO.AsofDate = FMD.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'T\_FMDDATA'

AND REF\_TBL.Portfolio = TGT\_DO.Portfolio

WHERE Tgt\_DO.PID IN ('O.D.4','O.D.5')

AND SourceTableName = 'T\_FMDDATA'

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,Tgt\_DO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: O.D.11 Principal Amount

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [app].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,[CreationDate]

)

SELECT DISTINCT

CASE

WHEN (

ISNULL(SUM(CAST(CDARS.AnticipatedPayoutAtMaturity AS DECIMAL(30,15))), 0) = ISNULL(sum(Tgt\_DO.maturityValue),0)

-- AND COUNT(distinct MM.MULTI\_CONCATE\_KEY) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,Tgt\_DO.portfolio AS [Portfolio]

,Tgt\_DO.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM(dbo.fn\_CurrencyExchange(CDARS.Currency,CDARS.AnticipatedPayoutAtMaturity, CONVERT(VARCHAR(10),CDARS.ASOFDATE,101)))

AS [AggregationFieldValue]

,COUNT(distinct 'CDARS|'+CDARS.AccountId+'|'+CDARS.OriginalPrincipalAmount+'|'+CAST(CDARS.Asofdate as varchar(100))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(Tgt\_DO.maturityValue) AS [TargetFieldValue]

,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,Tgt\_DO.ASOFDATE AS [AsOfDate]

,CreationDate AS [CreationDate]

--SELECT CDARS.\*

FROM APP.US\_FR2052A\_REF\_CDARS CDARS

INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'CDARS|'+CDARS.AccountId+'|'+CDARS.OriginalPrincipalAmount+'|'+CAST(CDARS.Asofdate as varchar(100))

AND Tgt\_DO.AsofDate = CDARS.ASOFDATE

INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.PID = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'US\_FR2052A\_REF\_CDARS'

AND REF\_TBL.Portfolio = TGT\_DO.Portfolio

WHERE Tgt\_DO.PID IN ('O.D.11')

AND SourceTableName = 'US\_FR2052A\_REF\_CDARS'

GROUP BY

ref\_en.reportingentityDescription

,Tgt\_DO.reportScope

,Tgt\_DO.PID

,Tgt\_DO.Product

,Tgt\_DO.ASOFDATE

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,Tgt\_DO.portfolio

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: All OD PIDs that source from T\_FMDDDATA

\*\* OD1 - OD 11 Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--SELECT

-- CASE

-- WHEN (

-- ISNULL(SUM(dbo.fn\_CurrencyExchange(TPOS.CCY,TPOS.PRINAMT, TPOS.ASOFDATE)), 0) = ISNULL(sum(Tgt\_DO.maturityValue),0)

-- -- AND COUNT(distinct MM.MULTI\_CONCATE\_KEY) = COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

-- ) --Run a check to see if source = target

-- THEN 'Pass'

-- ELSE ( 'Fail')

-- END AS [Validation\_Check]

-- ,Tgt\_DO.portfolio AS [Portfolio]

-- ,Tgt\_DO.PID AS [Product]

-- ,ref\_tbl.SourceTableName AS [src\_Table\_Name]

-- ,'Report Scope' AS [AggregationFieldGroupBy]

-- ,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

-- ,'Deal\_Balance' AS [AggregationField]

-- ,SUM(dbo.fn\_CurrencyExchange(TPOS.CCY,TPOS.PRINAMT, TPOS.ASOFDATE))

-- AS [AggregationFieldValue]

-- ,COUNT(distinct 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'

-- +LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))

-- +'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'

-- +LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))) AS [#ofRecordsFromSrc]

-- ,'Report Scope' AS [TargetFieldGroupBy]

-- ,Tgt\_DO.reportScope AS [TargetFieldGroupByValue]

-- ,'Market Value' AS [TargetField]

-- ,sum(Tgt\_DO.maturityValue) AS [TargetFieldValue]

-- ,COUNT(distinct Tgt\_DO.Tgt\_depositsOutflowID)

-- AS [#ofRecordsFromTgt]

-- ,ref\_tbl.TargetTableName AS [TargetTable]

-- ,Tgt\_DO.ASOFDATE AS [AsOfDate]

-- ,CreationDate AS [CreationDate]

--FROM SRC.US\_OPICS\_TPOS TPOS

-- INNER JOIN app.Tgt\_depositsOutflow Tgt\_DO

-- ON Tgt\_DO.SrcID = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))

-- INNER JOIN app.US\_FR2052A\_REF\_Entity ref\_en -- get table details from reference table

-- ON Ref\_en.reportingentityDescription = Tgt\_DO.reportScope

-- INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

-- ON ref\_tbl.PID = Tgt\_DO.PID

-- GROUP BY

-- ref\_en.reportingentityDescription

-- ,Tgt\_DO.reportScope

-- ,Tgt\_DO.PID

-- ,Tgt\_DO.Product

-- ,Tgt\_DO.ASOFDATE

-- ,ref\_tbl.TargetTableName

-- ,ref\_tbl.SourceTableName

-- ,Tgt\_DO.portfolio

-- ,Tgt\_DO.maturityValue

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Ctrl\_Asset\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Ctrl\_Asset\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Compare data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Ctrl\_Asset\_Inflow]

\*\* Report Type: Asset Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

SET NOCOUNT ON

SET XACT\_ABORT ON

--insert data into target table

INSERT INTO [APP].[Ctrl\_Src\_Tgt]

(

[ValidationCheck]

,[Portfolio]

,[Product]

,[SourceTableName]

,[SourceFieldGroupBy]

,[SourceFieldGroupByValue]

,[SourceAggregatedField]

,[SourceAggregatedValue]

,[#ofRecordsFromSrc]

,[TargetFieldGroupBy]

,[TargetFieldGroupByValue]

,[TargetAggregatedField]

,[TargetAggregatedValue]

,[#ofRecordsFromTgt]

,[TargetTableName]

,[AsOfDate]

,CreationDate

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.A.3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--Code for cash - not aggregating monetary values from ref tables

SELECT

CASE WHEN sum(dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE)) = SUM(tgt\_AI.marketValue)

THEN 'Pass'

ELSE 'Fail' END

AS [Validation\_Check]

,tgt\_AI.portfolio AS [Portfolio]

,tgt\_AI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,tgt\_AI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

, sum((tgt\_AI.marketValue)) AS [TargetFieldValue]

,COUNT(distinct tgt\_AI.Tgt\_assetInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM src.[T\_FMDDATA] fmd

INNER JOIN APP.[TGT\_assetInflow] tgt\_AI

ON tgt\_AI.SrcID = 'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR)

AND tgt\_AI.PID = 'I.A.3'

AND TGT\_AI.AsofDate = FMD.ASOFDATE

--get Reporting Entity

left JOIN src.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = tgt\_AI.reportScope

and tgt\_ai.CompanyName = ref\_en.Company\_name

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND REF\_TBL.Portfolio = TGT\_AI.Portfolio

where (( FMD.DESC1 = '0060' AND FMD.SECTOR = '3200')

OR

(FMD.DESC1 = ' 9003' AND CATEGORY = '19063'))

GROUP BY ref\_en.ReportingEntityDescription

,tgt\_AI.reportScope

,tgt\_AI.PID

,tgt\_AI.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,tgt\_AI.portfolio

,fmd.Currency

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.A.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT

CASE WHEN sum(dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE)) = SUM(tgt\_AI.marketValue)

THEN 'Pass'

ELSE 'Fail' END

AS [Validation\_Check]

,tgt\_AI.portfolio AS [Portfolio]

,tgt\_AI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.ReportingEntityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,sum(dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE))

AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,tgt\_AI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum((tgt\_AI.marketValue)) AS [TargetFieldValue]

,COUNT(distinct tgt\_AI.Tgt\_assetInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM src.[T\_FMDDATA] fmd

INNER JOIN APP.[TGT\_assetInflow] tgt\_AI

ON tgt\_AI.SrcID = 'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR)

AND tgt\_AI.PID = 'I.A.4'

AND TGT\_AI.AsofDate = FMD.ASOFDATE

--get Reporting Entity

left JOIN src.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.ReportingEntityDescription = tgt\_AI.reportScope

and tgt\_ai.CompanyName = ref\_en.Company\_name

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND REF\_TBL.Portfolio = TGT\_AI.Portfolio

GROUP BY ref\_en.ReportingEntityDescription

,tgt\_AI.reportScope

,tgt\_AI.PID

,tgt\_AI.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,tgt\_AI.portfolio

,fmd.Currency

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: BOLI

\*\* PID: I.A.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

sum(isnull(dbo.fn\_CurrencyExchange(fmd.currency,(fmd.Deal\_Balance), fmd.asofdate),0)) = isnull(SUM((tgt\_AI.marketValue)),0)

--AND COUNT(fmd.n\_id) = COUNT(tgt\_AI.Tgt\_assetInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,tgt\_AI.portfolio AS [Portfolio]

,tgt\_AI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,sum(isnull(dbo.fn\_CurrencyExchange(fmd.currency,(fmd.Deal\_Balance), fmd.asofdate),0)) AS [AggregationFieldValue]

,COUNT(distinct fmd.n\_id) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,tgt\_AI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,SUM(tgt\_AI.marketValue) AS [TargetFieldValue]

,COUNT(distinct tgt\_AI.Tgt\_assetInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,fmd.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

FROM src.[T\_FMDDATA] fmd

INNER JOIN APP.[TGT\_assetInflow] tgt\_AI

ON tgt\_AI.SrcID = cast(fmd.N\_ID AS VARCHAR(MAX))

AND tgt\_AI.PID = 'I.A.1'

AND TGT\_AI.Portfolio = 'BOLI'

AND tgt\_AI.AsofDate = fmd.asofdate

left JOIN src.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = tgt\_AI.reportScope

---- get table details from reference table

left JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_AI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

-- ,Ref\_El.Value

,tgt\_AI.reportScope

,tgt\_AI.PID

,tgt\_AI.Product

,[FMD].[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,tgt\_AI.portfolio

,fmd.Currency

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investment

\*\* PID: I.A.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

CASE

WHEN (

SUM((TPOS.MKTVAL)) = SUM((tgt\_AI.lendableValue))

--AND COUNT(fmd.n\_id) = COUNT(tgt\_AI.Tgt\_assetInflowID)

) --Run a check to see if source = target

THEN 'Pass'

ELSE ( 'Fail')

END AS [Validation\_Check]

,tgt\_AI.portfolio AS [Portfolio]

,tgt\_AI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'Deal\_Balance' AS [AggregationField]

,SUM((tgt\_AI.lendableValue)) AS [AggregationFieldValue]

,COUNT(distinct LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50))))) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,tgt\_AI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,(tgt\_AI.marketValue) AS [TargetFieldValue]

,COUNT(distinct tgt\_AI.Tgt\_assetInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,TPOS.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

-- select tgt\_AI.reportScope, tgt\_AI.marketValue AS marketvalue, tgt\_AI.PID

--linked server access

FROM SRC.US\_OPICS\_TPOS TPOS

INNER JOIN APP.[TGT\_assetInflow] tgt\_AI

ON tgt\_AI.SrcID = LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50))))

AND tgt\_AI.PID = 'I.A.1'

AND tgt\_AI.Portfolio = 'Treasury Investment'

--get Reporting Entity

INNER JOIN src.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = tgt\_AI.reportScope

and tgt\_ai.CompanyName = ref\_en.Company\_name

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_AI.Portfolio

--AND ref\_tbl.pid = fmd.PID

GROUP BY ref\_en.reportingentityDescription

,tgt\_AI.reportScope

,tgt\_AI.PID

,tgt\_AI.Product

,TPOS.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,tgt\_AI.portfolio

,tgt\_AI.marketValue

,TPOS.CCY

,creationDate

UNION

--Code for other PIDs

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investment

\*\* PID: I.A.5 & I.A.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

CASE WHEN SPSH.CCY = 'USD'

THEN

(

CASE WHEN SUM (SPSH.SETTAMT) = SUM(tgt\_AI.marketValue)

-- AND COUNT(fmd.n\_id) = COUNT(tgt\_AI.Tgt\_assetInflowID)

THEN 'Pass'

ELSE 'Fail' END

)

ELSE

(

CASE WHEN SUM(DBO.fn\_CurrencyExchange

(SPSH.CCY,SPSH.SETTAMT, tgt\_AI.AsofDate)) = SUM(tgt\_AI.marketValue)

THEN 'Pass'

ELSE 'Fail' END

)

END AS [Validation\_Check]

,tgt\_AI.portfolio AS [Portfolio]

,tgt\_AI.PID AS [Product]

,ref\_tbl.SourceTableName AS [src\_Table\_Name]

,'Report Scope' AS [AggregationFieldGroupBy]

,ref\_en.reportingentityDescription AS [AggregationFieldGroupByValue]

,'SPSH.SETTAMT' AS [AggregationField]

,CASE WHEN SPSH.CCY = 'USD'

THEN SUM(SPSH.SETTAMT)

ELSE

SUM(DBO.fn\_CurrencyExchange

(SPSH.CCY,SPSH.SETTAMT, tgt\_AI.AsofDate))

END AS [AggregationFieldValue]

,COUNT(distinct SPSH.SECID) AS [#ofRecordsFromSrc]

,'Report Scope' AS [TargetFieldGroupBy]

,tgt\_AI.reportScope AS [TargetFieldGroupByValue]

,'Market Value' AS [TargetField]

,sum(tgt\_AI.marketValue) AS [TargetFieldValue]

,COUNT(distinct tgt\_AI.Tgt\_assetInflowID) AS [#ofRecordsFromTgt]

,ref\_tbl.TargetTableName AS [TargetTable]

,tgt\_AI.ASOFDATE AS [AsOfDate]

,creationDate AS [CreationDate]

-- select tgt\_AI.reportScope, tgt\_AI.marketValue AS marketvalue, tgt\_AI.PID

--linked server access

FROM SRC.US\_OPICS\_SPSH SPSH

INNER JOIN

APP.[TGT\_assetInflow] tgt\_AI

ON tgt\_AI.SrcID = (SPSH.BR+' - '+SPSH.DEALNO)

AND

tgt\_AI.PID IN ('I.A.5', 'I.A.6')

AND tgt\_AI.Portfolio = 'Treasury Investment'

--get Reporting Entity

LEFT JOIN src.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.reportingentityDescription = tgt\_AI.reportScope

and tgt\_ai.CompanyName = ref\_en.Company\_name

-- get table details from reference table

LEFT JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

-- BOLI does not have a sector code

GROUP BY

ref\_en.reportingentityDescription

,tgt\_AI.reportScope

,tgt\_AI.PID

,tgt\_AI.Product

,tgt\_AI.[ASOFDATE]

,ref\_tbl.TargetTableName

,ref\_tbl.SourceTableName

,tgt\_AI.portfolio

,SPSH.CCY

,creationDate

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Asset\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Asset\_Inflow]

-- user parameter

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Asset\_Inflow]

\*\* Report Type: Asset Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

--Call the stored procedure to check if key tables exist and are populated

--Declare to enter PID dynamically

DECLARE @DESC11 VARCHAR(10)

DECLARE @DESC1 VARCHAR(10)

DECLARE @DESC12 VARCHAR(10)

DECLARE @Sector1 VARCHAR(10)

DECLARE @Sector2 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @Portfolio VARCHAR(100)

DECLARE @Currency VARCHAR(3)

DECLARE @SubProduct VARCHAR(100)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR(1)

DECLARE @SID INT

DECLARE @Contac\_ID VARCHAR(5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate1 DATE

DECLARE @LendableValue DECIMAL(30,15)

DECLARE @BR VARCHAR(3)

DECLARE @BR1 VARCHAR(3)

DECLARE @PS VARCHAR(3)

DECLARE @DELAYDELIVIND VARCHAR(25)

DECLARE @ACCOUNTNO VARCHAR(50)

DECLARE @ACCOUNTNO1 VARCHAR(50)

DECLARE @INVTYPE VARCHAR(10)

DECLARE @CATEGORY VARCHAR(30)

DECLARE @SETTAVGCOST INT

DECLARE @TDYPOOLQTY INT

DECLARE @CollateralClass VARCHAR (30)

DECLARE @ReportEntity VARCHAR(50)

DECLARE @Restricted VARCHAR(10)

DECLARE @REVREASON VARCHAR(5)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: BOLI

\*\* PID: I.A.1

\*\* Comment: At this time BOLI is not included, but may be included in future report submissions

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*

SET @PID = 'I.A.1'

SET @Portfolio = 'BOLI'

SET @Currency = 'USD'

SET @MaturityBucket = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'MaturityBucket'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @CollateralClass = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CollateralClass'

)

SET @DESC1 = 'DESC1'

SET @ReportEntity = (SELECT filter

FROM SRC.[US\_FR2052A\_REF\_TableFilters]

WHERE FilterName = 'ReportEntity')

--SET @LendableValue = (SELECT DISTINCT Value

-- FROM dbo.REF\_Element

-- WHERE Portfolio = @Portfolio

-- AND PID = @PID)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

--Insert scripts for target table

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[marketValue]

,[lendableValue]

,[maturityBucket]

,[forwardStartValue]

,[forwardStartBucket]

,[collateralClass]

,[treasuryControl]

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,[deal\_balance]

,[deal\_balance\_currency]

,AsofDate

,[event]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,orig\_date

,creationDate

)

-- Select records from source table

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.reportingentityDescription AS [ReportScope]

,@Currency AS [Currency]

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

--Get the product name from the reference table based on the PID

,ref\_rep.Product AS [Product]

,NULL AS [SID]

,NULL AS [SubProduct]

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [MarketVal]

,@LendableValue AS [LendableValue]

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket]

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardStartBucket]

,@CollateralClass AS [CollateralClass]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,[DESC1] AS [DESC1]

--Call the foreign exchange function to convert currency to USD

,COMP.Company\_Name AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [ReportType]

,fmd.DEAL\_BALANCE AS [deal\_balance]

,fmd.CURRENCY AS [currency\_deal\_balance]

-- If Currency is already in USD, the converted field will say 'N' else, 'Y'

,@AsofDate AS [AsofDate]

,(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

AS [EVENT]

,fmd.CUSTOMER\_NO AS customerNumber

,fmd.deal\_balance AS orig\_marketValue

,NULL AS orig\_maturityValue

,fmd.CURRENCY AS orig\_currency

,fmd.Deal\_Mat\_date AS orig\_date

,@CreationDate AS CreationDate

FROM

-- Access the linked server

SRC.T\_FMDDATA fmd

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

-- Join company table to get the company name based on the id from the source table

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON Ref\_en.Company\_code = COMP.Company\_Code COLLATE Latin1\_General\_CI\_AS

--get Product

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

fmd.[ASOFDATE] = @asofdate

AND fmd.[deal\_balance] IS NOT NULL

AND fmd.Deal\_Balance <> 0

AND fmd.DESC1 IN (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investment

\*\* PID: I.A.1

\*\* Comment: There are two sections (1) SSAH and (2) TPOS which are inserted into

APP.[TGT\_assetInflow] separately. Calculation is performed at aggregation.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* COMMENTOUTSTART \*/

SET @PID = 'I.A.1'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @MaturityBucket = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'MaturityBucket')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @DESC1 = 'DESC1'

SET @BR = 'BR'

SET @ACCOUNTNO = 'AccountNO'

SET @AsofDate1 = @AsOfDate

SET @CollateralClass = 'CollateralClass'

SET @ReportEntity = 'ReportEntity'

SET @INVTYPE = 'INVTYPE'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Section 1: SSAH, Report all values from SSAH. This section is to populate the QTY Field in the

Target Table with the TDYPOOLQTY

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,QTY

,[Portfolio]

,AsofDate

,[PID]

,[ReportScope]

)

SELECT

LTRIM(RTRIM('SSAH-'+LTRIM(RTRIM(BR)) +'|' + LTRIM(RTRIM(ACCOUNTNO)) + '|' +LTRIM(RTRIM(SECID))+ '|' + (CAST(LTRIM(RTRIM(SSAH.ASOFDATE)) AS VARCHAR(100)))))AS [SrcID] -- Create Unique ID

,SSAH.TDYPOOLQTY AS QTY

,@Portfolio AS [Portfolio]

,@AsofDate1 AS [AsofDate]

,@PID AS [PID]

,ref\_stb.ReportingEntityDescription AS [ReportScope]

FROM SRC.US\_OPICS\_SSAH SSAH

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON ref\_stb.BranchID = SSAH.BR

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE SSAH.ASOFDATE = @AsofDate1

AND SSAH.TDYPOOLQTY <> 0

AND SSAH.ACCOUNTNO IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @ACCOUNTNO) -- TDYPOOLQTY = Today pool quantity

AND SSAH.BR IN (SELECT DISTINCT FILTER -- Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR) -- BR Values 01,02,07

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

--GET all TPOS MKTVAL and QTY in BR (1,2,7) and MKTVAL NOT ZERO)

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[marketValue]

,[QTY]

,[maturityBucket]

,[lendableValue]

,[collateralClass]

,[treasuryControl]

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,CreationDate

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Section 2: TPOS, Report all MarketValue and Quatity from TPOS.

This section is to populate the MarketValue and quantity field in the target table

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT

LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(TPOS.BR)) + '|' + LTRIM(RTRIM(TPOS.SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(TPOS.PORT)) + '|' + LTRIM(RTRIM(TPOS.INVTYPE)) + '|' + CAST(TPOS.COST AS VARCHAR (50)))) AS [SrcID] -- create Unique Source ID

,ref\_stb.ReportingEntityDescription AS [ReportScope]

,@Currency AS [Currency]

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS [Product]

,TPOS.MKTVAL AS [MarketVal]

,TPOS.QTY AS [QTY]

,dbo.fn\_MaturityDateCalculater

(@AsofDate1, @AsofDate1,'Open') AS [MaturityBucket]

,TPOS.MKTVAL AS LendableValue

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass] -- Collateral class from DESCR Table

,CASE WHEN (SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) IN (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @CollateralClass)

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,NULL AS [DESC1]

,ref\_stb.BranchName AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate1 AS [AsofDate]

,((SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )) AS [EVENT] -- Reasonableness Test

,CASE WHEN SECM.ISSUER IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,TPOS.MKTVAL AS orig\_marketValue -- Reasonableness Test

,NULL AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,NULL AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON TPOS.BR = ref\_stb.BranchID

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_stb.ReportingEntityDescription

AND TPOS.ASOFDATE = @AsofDate

AND SECM.ASOFDATE = @AsofDate

AND TPOS.INVTYPE<> (SELECT filter -- INVETYPE <> I

FROM SRC.[US\_FR2052A\_REF\_TableFilters]

WHERE FilterName = @INVTYPE

AND PID = @PID) --TDYPOOLQTY = Today pool quantity

AND TPOS.SETTAVGCOST <> 0

AND TPOS.SETTQTY<> 0

AND SECM.MDATE > @AsofDate

AND TPOS.MKTVAL <> 0

AND TPOS.MKTVAL IS NOT NULL

AND TPOS.BR IN (SELECT DISTINCT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.A.3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.A.3'

SET @Portfolio = 'CASH'

SET @Sector = 'Sector'

SET @DESC1 = 'DESC1'

SET @Category = 'Category'

SET @Currency = 'USD'

SET @Restricted = 'I.A.4'

set @CATEGORY = 'Category'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[marketValue]

,[lendableValue]

,[maturityBucket]

,[forwardStartBucket]

,[forwardStartValue]

,[collateralClass]

,[treasuryControl]

,[Portfolio]

,[DESC1]

,[CompanyName]

,[Sector]

,[ReportType]

,[AsofDate]

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,CreationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ref\_rep.[Sid] AS [SID]

, ref\_rep.Subproduct AS [Sub-Product]

,dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE) AS [MarketValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, FMD.ASOFDATE) AS [LendableValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] --Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,'A-0-Q' AS [CollateralClass]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,@Portfolio AS Portfolio

, DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,FMD.Sector

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )) AS [EVENT] -- Reasonableness Test

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,fmd.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,fmd.deal\_balance AS orig\_marketValue -- Reasonableness Test

,NULL AS orig\_maturityValue -- Reasonableness Test

,fmd.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND COMP.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = COMP.company\_code

WHERE

FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND ((DESC1 = (SELECT Filter --DESC1 field = 60

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND SECTOR = (SELECT Filter --Sector = 3200 (Bank - Central)

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @SECTOR

)

)

OR

(DESC1 = (SELECT Filter --DESC1 field = 9003

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @Restricted

AND Portfolio = @Portfolio

AND FilterName = @DESC1

)

AND CATEGORY = (SELECT Filter --Category = 19063

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @Restricted

AND Portfolio = @Portfolio

AND FilterName = @CATEGORY

)

)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: CASH

\*\* PID: I.A.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.A.4'

SET @Portfolio = 'CASH'

SET @Sector = 'Sector'

SET @DESC1 = 'DESC1'

SET @Category = 'Category'

SET @Currency = 'USD'

SET @TreasuryControl = (SELECT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[marketValue]

,[lendableValue]

,[maturityBucket]

,[forwardStartBucket]

,[forwardStartValue]

,[collateralClass]

,[treasuryControl]

,[Portfolio]

,[DESC1]

,[CompanyName]

,[Sector]

,[ReportType]

,[AsofDate]

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR) AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ref\_rep.[SID] AS [SID]

, ref\_rep.Subproduct AS [Sub-Product]

,dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, @ASOFDATE)

AS [MarketValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_CurrencyExchange(FMD.CURRENCY, FMD.DEAL\_BALANCE, @ASOFDATE)

AS [LendableValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Open')

AS [MaturityBucket] --Calculate maturity bucket using maturity bucket function

, NULL AS [ForwardStartAmount]

, NULL AS [ForwardStartBucket]

,'A-0-Q' AS [CollateralClass]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,@Portfolio AS Portfolio

, DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,FMD.Sector

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN fmd.COMPOUND\_ID IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN fmd.CONTACT\_ID IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,fmd.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,fmd.deal\_balance AS orig\_marketValue -- Reasonableness Test

,NULL AS orig\_maturityValue -- Reasonableness Test

,fmd.CURRENCY AS orig\_currency -- Reasonableness Test

,fmd.Deal\_Mat\_date AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.T\_FMDDATA FMD

INNER JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

AND COMP.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_code = COMP.company\_code

WHERE

FMD.ASOFDATE = @ASOFDATE

AND fmd.[deal\_local\_bal] IS NOT NULL

AND fmd.[deal\_local\_bal] <> 0

AND DESC1 = (SELECT Filter --DESC1 field = 9003

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'desc1')

AND CATEGORY = (SELECT Filter --Category = 19063

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'category')

/\* ENDCOMMENTOUT \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investment

\*\* PID: I.A.5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.A.5'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @DESC1 = 'DESC1'

SET @BR = 'BR'

SET @PS = 'PS'

SET @DELAYDELIVIND = 'DELAYDELIVIND'

SET @CollateralClass = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Collateral Class')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @REVREASON = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'REVREASON')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

----Insert scripts for target table

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[marketValue]

,[maturityBucket]

,[lendableValue]

,forwardStartValue

,forwardStartBucket

,[collateralClass]

,[treasuryControl]

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,AsofDate

,[event]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

SPSH.BR+' - '+SPSH.DEALNO AS [SrcID] --Create unique source ID

,ref\_stb.ReportingEntityDescription AS [ReportScope]

,@Currency AS [Currency]

,CASE WHEN SPSH.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS [Product]

, NULL AS [SID]

, NULL AS [Sub-Product]

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) AS [MarketVal] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, '> 5')

END

END

END AS [MaturityBucket] --Calculate maturity bucket using maturity bucket function

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) \* (1-(SECM.SYMBOLID/100)) AS lendableValue --Calcuation performed with Foreign Currency Exchange function

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) AS [ForwardStartValue]

,CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, '> 5')

END

END

END AS [ForwardStartBucket] --Calculate maturity bucket using maturity bucket function

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass] --Collateral Class To be sourced from SECM.DESR for values described in the DAST

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,NULL AS [DESC1]

,ref\_stb.BranchName AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE OR SETTDATE < @AsofDate

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY IS NULL OR SPSH.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Customer -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN SPSH.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Company -- Reasonableness Test

,CASE WHEN SPSH.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,SPSH.SETTAMT AS orig\_marketValue -- Reasonableness Test

,NULL AS orig\_maturityValue -- Reasonableness Test

,SPSH.CCY AS orig\_currency -- Reasonableness Test

,SECM.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_SPSH SPSH

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON SECM.SECID = SPSH.SECID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON SPSH.BR = ref\_stb.BranchID

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

SPSH.[DELAYDELIVIND] = (SELECT FILTER --Delayed Delivery Indicator SET TO N

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @DELAYDELIVIND)

AND SPSH.PS = (SELECT FILTER --Purchase/Sale Indicator SET TO P

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @PS)

AND SPSH.BR IN(SELECT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @BR)

AND SPSH.SETTDATE > @AsOfDate

AND SPSH.REVREASON = @REVREASON

AND SPSH.ASOFDATE = @AsOfDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investment

\*\* PID: I.A.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.A.6'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @DESC1 = 'DESC1'

SET @BR = 'BR'

SET @PS = 'PS'

SET @DELAYDELIVIND = 'DELAYDELIVIND'

SET @CollateralClass = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Collateral Class')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @REVREASON = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'REVREASON')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

--Insert scripts for target table

INSERT INTO APP.[TGT\_assetInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[marketValue]

,[maturityBucket]

,[lendableValue]

,forwardStartValue

,forwardStartBucket

,[collateralClass]

,[treasuryControl]

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,AsofDate

,[event]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

SPSH.BR+' - '+SPSH.DEALNO AS [SrcID]

,ref\_stb.ReportingEntityDescription AS [ReportScope]

,@Currency AS [Currency]

,CASE WHEN SPSH.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS [Product]

, NULL AS [SID]

, NULL AS [Sub-Product]

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) AS [MarketVal] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SECM.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SECM.MDATE, '> 5')

END

END

END AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) \* (1-(SECM.SYMBOLID/100)) AS lendableValue -- Calcuation performed with Foreign Currency Exchange function

,DBO.fn\_CurrencyExchange

(SPSH.CCY,(SPSH.SETTAMT), @AsOfDate) AS [ForwardStartValue]

,CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsOfDate,SPSH.SETTDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsOfDate,SPSH.SETTDATE, '> 5')

END

END

END AS [ForwardStartBucket] -- Calculate maturity bucket using maturity bucket function

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass]

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,NULL AS [DESC1]

,ref\_stb.BranchName AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate1 AS [AsofDate]

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE OR SETTDATE < @AsofDate

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY IS NULL OR SPSH.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SPSH.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )) AS [EVENT] -- Reasonableness Test

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN SPSH.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN SPSH.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,SPSH.SETTAMT AS orig\_marketValue -- Reasonableness Test

,NULL AS orig\_maturityValue -- Reasonableness Test

,SPSH.CCY AS orig\_currency -- Reasonableness Test

,SECM.MDATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_SPSH SPSH

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON SECM.SECID = SPSH.SECID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON SPSH.BR = ref\_stb.BranchID

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

SPSH.[DELAYDELIVIND] IN (SELECT FILTER -- Delayed Delivery Indicator SET TO Y

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @DELAYDELIVIND)

AND SPSH.PS = (SELECT FILTER --Purchase/Sale Indicator SET TO P

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @PS)

AND SPSH.BR IN(SELECT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

and FilterName = @BR)

AND SPSH.SETTDATE > @AsOfDate -- Current Reporting date'12/30/2016'

AND SPSH.REVREASON = @REVREASON

AND SPSH.ASOFDATE = @AsOfDate

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

--If user parameter is entered as Null, throw a customized ISSUE

--Call procedure to handle ISSUEs

EXEC DBO.[Proc\_InsertissueDetails]

END CATCH

-- Execute ISSUE handling stroed procedure to get details about the ISSUEs

--SELECT \* FROM APP.[ISSUETracer]

--SELECT \* FROM APP.FRValidationTracer

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_AggFR2052a] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_AggFR2052a]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from target tables to aggregate tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_AggFR2052a]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.TGT\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON APP.TGT\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON APP.TGT\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.TGT\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

DROP INDEX IX\_Tgt\_AssetInflow ON APP.TGT\_assetInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

ON APP.TGT\_assetInflow(TGT\_assetInflowID);

DECLARE @MaturityStart VARCHAR(20)

DECLARE @MaturityEnd\_calc varchar(20)

DECLARE @MaturityEnd VARCHAR(20)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.1 (BOLI), I.A.5 & I.A.6 (Treasury Investment)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[lendableValue]

,[MarketValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[forwardstartValue]

,[ForwardStartBucket]

,[Forwardstart]

,[Forwardend]

,[ReportType]

,[CollateralClass]

,[TreasuryControL]

,asofdate

,creationdate

)

SELECT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.product AS [Product]

,tgt\_AI.[SID] AS [SID]

,tgt\_AI.SubProduct AS [SubProduct]

,sum(ABS(tgt\_AI.lendableValue)) AS [lendableValue]

,sum(ABS(tgt\_AI.marketValue)) AS [MarketValue]

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,tgt\_ai.forwardStartBucket AS [ForwardStartBucket]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

APP.[TGT\_assetInflow] tgt\_AI

WHERE ((PID = 'I.A.1'

AND Portfolio = 'BOLI')

OR

(PID IN ('I.A.4')

AND Portfolio = 'CASH')

OR

(PID IN ('I.A.5', 'I.A.6')

AND Portfolio = 'Treasury Investment')

)

GROUP BY tgt\_AI.reportScope

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.DESC1

,tgt\_AI.PID

,tgt\_AI.Product

,tgt\_AI.[SID]

,TGT\_AI.SubProduct

,tgt\_AI.maturityBucket

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,tgt\_AI.CompanyName

,Tgt\_AI.reportType

,[ForwardStartValue]

,forwardStartBucket

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.3 (CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[lendableValue]

,[MarketValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[forwardstartValue]

,[ForwardStartBucket]

,[Forwardstart]

,[Forwardend]

,[ReportType]

,[CollateralClass]

,[TreasuryControL]

,asofdate

,creationdate

)

SELECT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.product AS [Product]

,tgt\_AI.[SID] AS [SID]

,tgt\_AI.SubProduct AS [SubProduct]

,SUM(ABS(TGT\_AI.marketValue)) - (SELECT SUM(ABS(TGT\_AI.marketValue))

FROM APP.Tgt\_assetInflow TGT\_AI

WHERE DESC1 = '9003'

AND PID ='I.A.4') AS [lendableValue]

,SUM(ABS(TGT\_AI.marketValue)) - (SELECT SUM(ABS(TGT\_AI.marketValue))

FROM APP.Tgt\_assetInflow TGT\_AI

WHERE DESC1 = '9003'

AND PID ='I.A.4') AS [MarketValue]

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM((tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,tgt\_ai.forwardStartBucket AS [ForwardStartBucket]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

APP.[TGT\_assetInflow] tgt\_AI

WHERE (

(PID IN ('I.A.3')

AND Portfolio = 'CASH')

)

and marketValue is not null

and lendableValue is not null

and marketValue <> 0

and lendableValue <> 0

and tgt\_ai.DESC1 = '0060'

GROUP BY tgt\_AI.reportScope

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.DESC1

,tgt\_AI.PID

,tgt\_AI.Product

,tgt\_AI.[SID]

,TGT\_AI.SubProduct

,tgt\_AI.maturityBucket

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,tgt\_AI.CompanyName

,Tgt\_AI.reportType

,[ForwardStartValue]

,forwardStartBucket

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.1 (Treasury Investment)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--=================================================================================================

--Create temp table containing copy of data from TGT\_assetInflow table

IF OBJECT\_ID('tempdb..#Temp')IS NOT NULL DROP TABLE #Temp

SELECT \*

INTO #Temp

FROM APP.[TGT\_assetInflow]

WHERE PID = 'I.A.1'

AND Portfolio = 'Treasury Investment'

--Test temp table

--SELECT \* FROM #tEMP WHERE SrcID LIKE '%US94974BGG95%' OR SrcID LIKE '%US94974BGS34%'

--Create fields holding Security ID and Branch (BR) information. This will be used to perform calculations at Security and Branch level

IF OBJECT\_ID('tempdb..#Temp\_A')IS NOT NULL DROP TABLE #Temp\_A

SELECT

CASE LEFT(A.SRCID, 4)

WHEN 'TPOS' THEN SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

ELSE SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

END AS SECID

, SUBSTRING(SrcID, 6, 2) AS BR

, A.\*

INTO #Temp\_A

FROM #Temp A

--==================================================================================================

--Create aggregate of TPOS and SSAH based on subqueries aggregating TPOS and SSAH separately

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[lendableValue]

,[MarketValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[forwardstartValue]

,[ForwardStartBucket]

,[Forwardstart]

,[Forwardend]

,[ReportType]

,[CollateralClass]

,[TreasuryControL]

,asofdate

,creationdate

)

SELECT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.PID AS [PID]

,tgt\_AI.product AS [Product]

,tgt\_AI.[SID] AS [SID]

,tgt\_AI.SubProduct AS [SubProduct]

,SUM(CAST(Tgt\_AI.LendableValue AS DECIMAL (30,15))) AS [LendableValue]

,SUM(CAST(tgt\_AI.marketValue AS DECIMAL (30,15))) AS [MarketValue]

,TGT\_AI.maturityBucket AS [MaturityBucket]

,Tgt\_AI.maturityStart AS [maturityStart]

,Tgt\_AI.maturityEnd AS [maturityEnd]

,Tgt\_ai.ForwardStartValue AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,tgt\_AI.ForwardStart AS [ForwardStart]

,tgt\_AI.ForwardEnd AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM (

SELECT

TPOS.SECID

,TPOS.BR

,TPOS.reportScope AS [ReportScope]

,TPOS.CompanyName AS [CompanyName]

,TPOS.Currency AS [Currency]

,TPOS.Converted AS [Converted]

,TPOS.DESC1 AS [DESC1]

,TPOS.PID AS [PID]

-- ,TPOS.Portfolio AS [Portfolio]

,TPOS.Product AS [Product]

,TPOS.[SID] AS [SID]

,TPOS.SubProduct AS [Subproduct]

,SUM(ABS(TPOS.LendableValue)) AS [TPOS\_LendableValue]

,SUM(ABS(TPOS.marketValue)) AS [TPOS\_MarketValue]

,SUM(TPOS.Original\_QTY) AS TPOS\_QTY

,SUM(SSAH.Original\_QTY) AS SSAH\_QTY

,((SUM(CAST(TPOS.Original\_QTY AS DECIMAL(30,15))) - SUM(isnull(CAST(SSAH.Original\_QTY AS DECIMAL(30,15)),0))) / SUM(CAST(TPOS.Original\_QTY AS DECIMAL(30,15)))) \* SUM(ABS(CAST(TPOS.marketValue AS DECIMAL(30,15)))) AS [MarketValue]

,(((SUM(CAST(TPOS.Original\_QTY AS DECIMAL(30,15))) - SUM(isnull(CAST(SSAH.Original\_QTY AS DECIMAL(30,15)),0))) / SUM(CAST(TPOS.Original\_QTY AS DECIMAL(30,15)))) \* SUM(ABS(CAST(TPOS.marketValue AS DECIMAL(30,15))))) \* (1-(CAST(SECM.SYMBOLID AS FLOAT)/100)) AS [LendableValue]

,TPOS.[MaturityBucket]

,TPOS.[maturityStart]

,TPOS.[maturityEnd]

,ABS(SUM(TPOS.forwardStartValue)) AS [ForwardStartValue]

,TPOS.[ForwardStart]

,TPOS.[ForwardEnd]

,TPOS.reportType AS [ReportType]

,TPOS.collateralClass AS [CollateralClass]

,TPOS.treasuryControl AS [TreasuryControl]

,TPOS.AsofDate AS [Asofdate]

,TPOS.creationDate AS [CreationDate]

FROM

(

---SUBQUERY to get Aggregate of TPOS at Security and BR levels

SELECT

tgt\_AI.SECID

, tgt\_AI.BR

, tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

--,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,SUM(Tgt\_AI.LendableValue) AS [LendableValue]

,SUM(tgt\_AI.marketValue) AS [MarketValue]

,SUM(tgt\_AI.QTY) AS Original\_QTY

--,(SUM(CASE WHEN LEFT(tgt\_AI.SRCID, 4) = 'TPOS' THEN tgt\_AI.QTY ELSE (-1 \* tgt\_AI.QTY) END)/SUM(CASE WHEN LEFT(tgt\_AI.SRCID, 4) = 'TPOS' THEN tgt\_AI.QTY ELSE 0 END)\*(SUM(CAST(ISNULL(tgt\_AI.[marketValue],1)AS DECIMAL(30,15))))) AS apportionedMarketValueScript

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(abs(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM #Temp\_A tgt\_AI

WHERE tgt\_AI.PID = 'I.A.1' AND LEFT(tgt\_AI.SRCID, 4) = 'TPOS' --AND tgt\_AI.SECID IN ('US94974BGG95', 'US94974BGS34')

GROUP BY

tgt\_AI.SECID, tgt\_AI.BR

, tgt\_AI.reportScope, tgt\_AI.Currency, tgt\_AI.Converted, tgt\_AI.DESC1, tgt\_AI.PID, tgt\_AI.Product, tgt\_AI.[SID], TGT\_AI.SubProduct, tgt\_AI.maturityBucket

, tgt\_AI.collateralClass, tgt\_AI.treasuryControl, tgt\_AI.CompanyName, Tgt\_AI.reportType, tgt\_AI.forwardStartValue, tgt\_AI.forwardStartBucket, Tgt\_AI.AsofDate, TGT\_AI.creationDate

) AS TPOS

LEFT JOIN

(

---SUBQUERY to get Aggregate of SSAH at Security and BR levels

SELECT

tgt\_AI.SECID

,tgt\_AI.BR

,tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

--,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,SUM(CAST(Tgt\_AI.LendableValue AS DECIMAL (30,15))) AS [LendableValue]

,SUM(CAST(tgt\_AI.marketValue AS DECIMAL (30,15))) AS [MarketValue]

,SUM(CAST(tgt\_AI.QTY AS DECIMAL (30,15))) AS Original\_QTY

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM #Temp\_A tgt\_AI

WHERE tgt\_AI.PID = 'I.A.1' AND LEFT(tgt\_AI.SRCID, 4) = 'SSAH' --AND tgt\_AI.SECID IN ('US94974BGG95', 'US94974BGS34') --AND LEFT(tgt\_AI\_SSAH.SRCID, 4) = 'SSAH'

GROUP BY

tgt\_AI.SECID, tgt\_AI.BR

, tgt\_AI.reportScope, tgt\_AI.Currency, tgt\_AI.Converted, tgt\_AI.DESC1, tgt\_AI.PID ,tgt\_AI.Product, tgt\_AI.[SID], TGT\_AI.SubProduct, tgt\_AI.maturityBucket

, tgt\_AI.collateralClass, tgt\_AI.treasuryControl, tgt\_AI.CompanyName, Tgt\_AI.reportType, tgt\_AI.forwardStartValue, tgt\_AI.forwardStartBucket, Tgt\_AI.AsofDate,creationDate

) AS SSAH

ON

TPOS.reportScope = SSAH.reportScope

--AND TPOS.Portfolio = SSAH.Portfolio

AND TPOS.PID = SSAH.PID

AND TPOS.BR = SSAH.BR

AND TPOS.SECID = SSAH.SECID

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.AsofDate = SECM.ASOFDATE

GROUP BY

TPOS.SECID, TPOS.BR

, TPOS.reportScope, TPOS.Currency, TPOS.Converted, TPOS.DESC1, TPOS.PID, TPOS.Product, TPOS.[SID], TPOS.SubProduct, TPOS.maturityBucket

, TPOS.maturityStart, TPOS.maturityEnd, TPOS.ForwardStart, TPOS.ForwardEnd, TPOS.collateralClass, TPOS.treasuryControl, TPOS.CompanyName, TPOS.reportType,SECM.SYMBOLID, TPOS.ASOFDATE, TPOS.CreationDate

) tgt\_AI

GROUP BY

tgt\_AI.reportScope

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.PID

,tgt\_AI.product

,tgt\_AI.[SID]

,tgt\_AI.SubProduct

,TGT\_AI.maturityBucket

,Tgt\_AI.maturityStart

,Tgt\_AI.maturityEnd

,Tgt\_ai.ForwardStartValue

,tgt\_AI.ForwardStart

,tgt\_AI.ForwardEnd

,Tgt\_AI.reportType

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,TGT\_AI.AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU1, IU2 (Wholesale Funding Transactions)

, IU3 NB (Other Receivable),

IU6 NB (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,asofdate

,creationdate

)

SELECT

Tgt\_UI.reportScope AS [ReportScope]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

,SUM(Tgt\_UI.maturityValue) AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_UI.forwardStartValue) AS [Forward Start]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

[APP].[TGT\_UnsecuredInflow] tgt\_UI

WHERE (

( PID IN ('I.U.1','I.U.2')

AND Portfolio = 'Wholesale Funding Transactions')

OR

( PID IN ('I.U.3')

AND Portfolio = 'Other Receivable')

OR

( PID IN ('I.U.6')

AND Portfolio = 'Wholesale Funding Transactions')

)

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU5, IU6, I.S.7 (Commercial and Retail loans),

IU6 (Trade Finance), IU3(CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,asofdate

,creationdate

)

SELECT

Tgt\_UI.reportScope AS [ReportScope]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

,SUM(ABS(Tgt\_UI.maturityValue)) AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_UI.forwardStartValue) AS [Forward Start]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

[APP].[TGT\_UnsecuredInflow] tgt\_UI

WHERE ((PID IN ('I.U.5','I.U.6')

AND Portfolio = 'Commercial and Retail Loans')

OR

(PID = 'I.U.6'

AND Portfolio = 'Trade Finance')

OR

( PID IN ('I.U.3')

AND Portfolio = 'Cash')

)

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU4 (CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,asofdate

,creationdate

)

SELECT

Tgt\_UI.reportScope AS [ReportScope]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

, SUM(CASE WHEN TGT\_UI.SrcID like 'FMD\_ALL | %' THEN (-1)\*TGT\_UI.maturityValue

WHEN TGT\_UI.SrcID like 'MULTI%' THEN (-1)\* TGT\_UI.maturityValue

WHEN TGT\_UI.SrcID like 'T\_FMDDATA | %' THEN (-1)\* abs(TGT\_UI.maturityValue)

END ) AS [MaturityValue]

--,ABS(CASE WHEN tgt\_ui.SrcID like 'FMD\_ALL | %' THEN SUM(TGT\_UI.maturityValue) -

-- (CASE WHEN tgt\_ui.SrcID like 'MULTI%' THEN SUM(TGT\_UI.maturityValue) -

--(CASE WHEN tgt\_ui.SrcID like 'FMD | %' THEN SUM(TGT\_UI.maturityValue) ELSE 0 END

-- )

-- ELSE 0 END )

--ELSE 0 END ) AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(ABS(Tgt\_UI.forwardStartValue)) AS [Forward Start]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

[APP].[TGT\_UnsecuredInflow] tgt\_UI

WHERE

PID IN ('I.U.4')

AND Portfolio = 'Cash'

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Inflows IS1 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SUbProduct]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[EffectiveMaturityBucket]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Unencumbered]

,treasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[Settlementmechanism]

,[CID]

,[counterPartySector]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_SI.reportScope AS [ReportScope]

,Tgt\_SI.Currency AS [Currency]

,Tgt\_SI.Converted AS [Converted]

,SUBSTRING(tgt\_SI.PID, 5, 100) AS [PID]

,Tgt\_SI.product AS [Product]

,tgt\_SI.[SID] AS [SID]

,tgt\_SI.[SubProduct] AS [SubProduct]

,SUM(Tgt\_SI.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_SI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_SI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SI.forwardStartBucket,0, CHARINDEX('-', Tgt\_SI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_SI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SI.forwardStartBucket,CHARINDEX('-', Tgt\_SI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_SI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SI.maturityBucket,0, CHARINDEX('-', Tgt\_SI.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_SI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SI.maturityBucket,CHARINDEX('-', Tgt\_SI.maturityBucket)+1,50)

END AS [maturityEnd]

,tgt\_SI.EffectiveMaturityBucket AS [EffectiveMaturityDate]

,SUM(Tgt\_SI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_SI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SI.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_SI.collateralClass AS [CollateralClass]

,SUM(Tgt\_SI.collateralValue) AS [CollateralValue]

,Tgt\_SI.CollateralCurrency AS [CollateralCurrency]

,tgt\_SI.Unencumbered AS [Unencumbered]

,Tgt\_SI.treasuryControl AS [TreasuryControl]

,Tgt\_SI.internal AS [Internal]

,Tgt\_SI.internalCounterParty AS [InternalCounterparty]

,Tgt\_SI.primeBrokerage AS [PrimeBrokerage]

,tgt\_SI.settlementMechanism AS [Settlement]

,tgt\_SI.CID AS [CID]

,tgt\_SI.counterPartySector AS [COunterParty]

,Tgt\_SI.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_securedInflow tgt\_SI

WHERE ((PID = 'I.S.1'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'I.S.7'

AND Portfolio = 'Commercial and Retail Loans'))

GROUP BY

Tgt\_SI.reportScope

,Tgt\_SI.Currency

,Tgt\_SI.Converted

,Tgt\_SI.PID

,Tgt\_SI.product

,tgt\_SI.[SID]

,tgt\_SI.[SubProduct]

,Tgt\_SI.maturityBucket

,tgt\_SI.EffectiveMaturityBucket

,reportType

,tgt\_SI.CID

,tgt\_SI.counterPartySector

,Tgt\_SI.collateralClass

,Tgt\_SI.CollateralCurrency

,tgt\_SI.Unencumbered

,Tgt\_SI.forwardStartBucket

,tgt\_SI.settlementMechanism

,Tgt\_SI.internal

,Tgt\_SI.treasuryControl

,Tgt\_SI.internalCounterParty

,Tgt\_SI.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - IO8 (Treasury Investments)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

----Create temp table containing copy of data from TGT\_assetInflow table

IF OBJECT\_ID('tempdb..#Temp\_IO8')IS NOT NULL DROP TABLE #Temp\_IO8

SELECT \*

INTO #Temp\_IO8

FROM [APP].[TGT\_otherInflow]

WHERE PID = 'I.O.8'

AND Portfolio = 'Treasury Investment'

--Test temp table

--SELECT \* FROM #tEMP WHERE SrcID LIKE '%US94974BGG95%' OR SrcID LIKE '%US94974BGS34%'

--Create fields holding Security ID and Branch (BR) information. This will be used to perform calculations at Security and Branch level

IF OBJECT\_ID('tempdb..#Temp\_A\_IO8')IS NOT NULL DROP TABLE #Temp\_A\_IO8

SELECT

CASE LEFT(A.SRCID, 4)

WHEN 'TPOS' THEN SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

ELSE SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

END AS SECID

, SUBSTRING(SrcID, 6, 2) AS BR

, A.\*

INTO #Temp\_A\_IO8

FROM #Temp\_IO8 A

--==================================================================================================

--Create aggregate of TPOS and SSAH based on subqueries aggregating TPOS and SSAH separately

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[maturityValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[forwardstartValue]

,[ForwardStartBucket]

,[Forwardstart]

,[Forwardend]

,[ReportType]

,[CollateralClass]

,[CollateralValue]

,[TreasuryControL]

,[Internal]

,InternalCounterParty

,PrimeBrokerage

,asofdate

,creationdate

)

SELECT

tgt\_IO\_outer.reportScope AS [ReportScope]

,tgt\_IO\_outer.Currency AS [Currency]

,tgt\_IO\_outer.Converted AS [Converted]

,tgt\_IO\_outer.PID AS [PID]

,tgt\_IO\_outer.product AS [Product]

,SUM(CAST(tgt\_IO\_outer.maturityValue AS DECIMAL (30,15))) AS [maturityValue]

,tgt\_IO\_outer.maturityBucket AS [MaturityBucket]

,tgt\_IO\_outer.maturityStart AS [maturityStart]

,tgt\_IO\_outer.maturityEnd AS [maturityEnd]

,tgt\_IO\_outer.ForwardStartValue AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,tgt\_IO\_outer.ForwardStart AS [ForwardStart]

,tgt\_IO\_outer.ForwardEnd AS [ForwardEnd]

,tgt\_IO\_outer.reportType AS [ReportType]

,tgt\_IO\_outer.collateralClass AS [CollateralClass]

,SUM(CAST(tgt\_IO\_outer.maturityValue AS DECIMAL (30,15))) AS [CollateralValue]

,tgt\_IO\_outer.treasuryControl AS [TreasuryControl]

,tgt\_IO\_outer.Internal AS Internal

,tgt\_IO\_outer.InternalCounterParty AS InternalCounterParty

,tgt\_IO\_outer.PrimeBrokage AS primeBrokage

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM (

SELECT

TPOS.SECID

,TPOS.BR

,TPOS.reportScope AS [ReportScope]

,TPOS.CompanyName AS [CompanyName]

,TPOS.Currency AS [Currency]

,TPOS.Converted AS [Converted]

,TPOS.DESC1 AS [DESC1]

,TPOS.PID AS [PID]

--,TPOS.Portfolio AS [Portfolio]

,TPOS.Product AS [Product]

--,TPOS.[SID] AS [SID]

--,TPOS.SubProduct AS [Subproduct]

--,SUM(ABS(TPOS.LendableValue)) AS [TPOS\_LendableValue]

,SUM(ABS(TPOS.maturityValue)) AS [TPOS\_maturityValue]

,SUM(TPOS.Original\_QTY) AS TPOS\_QTY

,SUM(SSAH.Original\_QTY) AS SSAH\_QTY

,((SUM(CAST(TPOS.Original\_QTY AS DECIMAL(30,15))) - SUM(isnull(CAST(SSAH.Original\_QTY AS DECIMAL(30,15)),0))) / SUM(CAST(TPOS.Original\_QTY AS DECIMAL (30,15)))) \* SUM(ABS(CAST(TPOS.maturityValue AS DECIMAL (30,15)))) AS MaturityValue

--,(((SUM(TPOS.Original\_QTY) - SUM(isnull(SSAH.Original\_QTY,0))) / SUM(TPOS.Original\_QTY)) \* SUM(ABS(TPOS.maturityValue))) \* (1-(ref\_el.Value/100)) AS [LendableValue]

,TPOS.[MaturityBucket]

,TPOS.[maturityStart]

,TPOS.[maturityEnd]

,ABS(SUM(CAST(TPOS.forwardStartValue AS DECIMAL (30,15)))) AS [ForwardStartValue]

,TPOS.[ForwardStart]

,TPOS.[ForwardEnd]

,TPOS.reportType AS [ReportType]

,TPOS.collateralClass AS [CollateralClass]

,TPOS.treasuryControl AS [TreasuryControl]

,TPOS.Internal AS [Internal]

,TPOS.InternalCounterParty AS [InternalCounterParty]

,TPOS.PrimeBrokage AS [PrimeBrokage]

,TPOS.AsofDate AS [Asofdate]

,TPOS.creationDate AS [CreationDate]

FROM

(

---SUBQUERY to get Aggregate of TPOS at Security and BR levels

SELECT

tgt\_IO.SECID

,tgt\_IO.BR

,tgt\_IO.reportScope AS [ReportScope]

,tgt\_IO.CompanyName AS [CompanyName]

,tgt\_IO.Currency AS [Currency]

,tgt\_IO.Converted AS [Converted]

,tgt\_IO.DESC1 AS [DESC1]

,SUBSTRING(tgt\_IO.PID, 5, 100) AS [PID]

--,tgt\_IO.Portfolio AS [Portfolio]

,tgt\_IO.Product AS [Product]

--,tgt\_IO.[SID] AS [SID]

--,tgt\_IO.SubProduct AS [Subproduct]

--,SUM(tgt\_IO.LendableValue) AS [LendableValue]

-- ,SUM(tgt\_IO.maturityValue) AS [maturityValue]

, tgt\_IO.[CollateralClass] AS [CollateralClass]

,SUM((DBO.fn\_CurrencyExchange (tgt\_IO.Currency,(tgt\_IO.maturityValue), Tgt\_IO.AsofDate))) AS [maturityValue]

,SUM(CAST(tgt\_IO.QTY AS DECIMAL (30,15))) AS Original\_QTY

,Tgt\_IO.reportType AS reportType

,tgt\_IO.treasuryControl AS TreasuryControl

,Tgt\_IO.internal AS Internal

,Tgt\_IO.internalCounterParty AS internalCounterParty

,Tgt\_IO.primeBrokerage AS primeBrokage

--,(SUM(CASE WHEN LEFT(tgt\_IO.SRCID, 4) = 'TPOS' THEN tgt\_IO.QTY ELSE (-1 \* tgt\_IO.QTY) END)/SUM(CASE WHEN LEFT(tgt\_IO.SRCID, 4) = 'TPOS' THEN tgt\_IO.QTY ELSE 0 END)\*(SUM(CAST(ISNULL(tgt\_IO.[maturityValue],1)AS DECIMAL(30,15))))) AS apportionedmaturityValueScript

,CASE WHEN tgt\_IO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_IO.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,0, CHARINDEX('-', tgt\_IO.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,CHARINDEX('-', tgt\_IO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(abs(tgt\_IO.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,0, CHARINDEX('-', tgt\_IO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,CHARINDEX('-', tgt\_IO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM #Temp\_A\_IO8 tgt\_IO

WHERE tgt\_IO.PID = 'I.O.8' AND LEFT(tgt\_IO.SRCID, 4) = 'TPOS' --AND tgt\_IO.SECID IN ('US94974BGG95', 'US94974BGS34')

GROUP BY

tgt\_IO.SECID, tgt\_IO.BR

, tgt\_IO.reportScope, tgt\_IO.Currency, tgt\_IO.Converted, tgt\_IO.DESC1, tgt\_IO.PID, tgt\_IO.Product, tgt\_IO.[CollateralClass], tgt\_IO.maturityBucket

, tgt\_IO.collateralClass, tgt\_IO.treasuryControl, tgt\_IO.CompanyName, tgt\_IO.reportType, tgt\_IO.forwardStartValue, tgt\_IO.forwardStartBucket

, tgt\_IO.internal, tgt\_IO.internalCounterParty, Tgt\_IO.primeBrokerage,AsofDate, CREATIONDATE

) AS TPOS

LEFT JOIN

(

---SUBQUERY to get Aggregate of SSAH at Security and BR levels

SELECT

tgt\_IO.SECID

,tgt\_IO.BR

,tgt\_IO.reportScope AS [ReportScope]

,tgt\_IO.CompanyName AS [CompanyName]

,tgt\_IO.Currency AS [Currency]

,tgt\_IO.Converted AS [Converted]

,tgt\_IO.DESC1 AS [DESC1]

,SUBSTRING(tgt\_IO.PID, 5, 100) AS [PID]

--,tgt\_IO.Portfolio AS [Portfolio]

,tgt\_IO.Product AS [Product]

,SUM(CAST(tgt\_IO.maturityValue AS DECIMAL(30,15))) AS [maturityValue]

,SUM(CAST(tgt\_IO.QTY AS DECIMAL (30,15))) AS Original\_QTY

,CASE WHEN tgt\_IO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_IO.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,0, CHARINDEX('-', tgt\_IO.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,CHARINDEX('-', tgt\_IO.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_IO.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,0, CHARINDEX('-', tgt\_IO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,CHARINDEX('-', tgt\_IO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,tgt\_IO.reportType AS [ReportType]

,tgt\_IO.collateralClass AS [CollateralClass]

,tgt\_IO.treasuryControl AS [TreasuryControl]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM #Temp\_A\_IO8 tgt\_IO

WHERE tgt\_IO.PID = 'I.O.8' AND LEFT(tgt\_IO.SRCID, 4) = 'SSAH' --AND tgt\_IO.SECID IN ('US94974BGG95', 'US94974BGS34') --AND LEFT(tgt\_IO\_SSAH.SRCID, 4) = 'SSAH'

GROUP BY

tgt\_IO.SECID, tgt\_IO.BR

, tgt\_IO.reportScope, tgt\_IO.Currency, tgt\_IO.Converted, tgt\_IO.DESC1, tgt\_IO.PID, tgt\_IO.Product, tgt\_IO.maturityBucket

, tgt\_IO.collateralClass, tgt\_IO.treasuryControl, tgt\_IO.CompanyName, tgt\_IO.reportType, tgt\_IO.forwardStartValue, tgt\_IO.forwardStartBucket, AsofDate, CREATIONDATE

) AS SSAH

ON

TPOS.reportScope = SSAH.reportScope

--AND TPOS.Portfolio = SSAH.Portfolio

AND TPOS.PID = SSAH.PID

AND TPOS.BR = SSAH.BR

AND TPOS.SECID = SSAH.SECID

GROUP BY

TPOS.SECID, TPOS.BR

, TPOS.reportScope, TPOS.Currency, TPOS.Converted, TPOS.DESC1, TPOS.PID, TPOS.Product, TPOS.maturityBucket

, TPOS.maturityStart, TPOS.maturityEnd, TPOS.ForwardStart, TPOS.ForwardEnd, TPOS.treasuryControl, TPOS.CompanyName, TPOS.reportType, TPOS.collateralClass, TPOS.Internal

, TPOS.internalCounterParty, TPOS.primeBrokage,TPOS.Asofdate, TPOS.CreationDate

) tgt\_IO\_outer

GROUP BY

tgt\_IO\_outer.reportScope

,tgt\_IO\_outer.Currency

,tgt\_IO\_outer.Converted

,tgt\_IO\_outer.PID

,tgt\_IO\_outer.product

--,tgt\_IO.[SID]

--,tgt\_IO.SubProduct

,tgt\_IO\_outer.maturityBucket

,tgt\_IO\_outer.maturityStart

,tgt\_IO\_outer.maturityEnd

,tgt\_IO\_outer.ForwardStartValue

,tgt\_IO\_outer.ForwardStart

,tgt\_IO\_outer.ForwardEnd

,tgt\_IO\_outer.reportType

,tgt\_IO\_outer.treasuryControl

,tgt\_IO\_outer.Internal

,tgt\_IO\_outer.InternalCounterParty

,tgt\_IO\_outer.PrimeBrokage

,tgt\_IO\_outer.collateralClass

,AsofDate

,creationDate

-- INSERT INTO [APP].[AGG\_FR2052a]

--(

-- [ReportScope]

-- ,[Currency]

-- ,[Converted]

-- ,[PID]

-- ,[Product]

-- ,[Maturityvalue]

-- ,[MaturityBucket]

-- ,[maturityStart]

-- ,[maturityEnd]

-- ,[ForwardStartValue]

-- ,[ForwardStartBucket]

-- ,collateralClass

-- ,collateralValue

-- ,CollateralCurrency

-- ,treasuryControl

-- ,[Internal]

-- ,[internalCounterParty]

-- ,[primeBrokerage]

-- ,[ReportType]

--)

-- SELECT

-- tgt\_OI.reportScope AS [ReportScope]

-- ,tgt\_OI.Currency AS [Currency]

-- ,tgt\_OI.Converted AS [Converted]

-- ,SUBSTRING(tgt\_OI.PID, 5, 100) AS [PID]

-- ,tgt\_OI.product AS [Product]

-- ,SUM(ABS(tgt\_OI.maturityValue)) AS [MaturityAmount]

-- ,CASE WHEN tgt\_OI.maturityBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.maturityBucket

-- END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE

-- SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket))

-- END AS [maturityStart]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_OI.maturityBucket,CHARINDEX('-', Tgt\_OI.maturityBucket)+1,50)

-- END AS [maturityEnd]

-- ,SUM(tgt\_OI.forwardStartValue) AS [ForwardStart]

-- ,CASE WHEN tgt\_OI.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.forwardStartBucket

-- END AS [ForwardBucket]

-- ,tgt\_OI.collateralClass AS [CollateralClass]

-- ,SUM(tgt\_OI.collateralValue) AS [CollateralValue]

-- ,tgt\_OI.CollateralCurrency AS [CollateralCurrency]

-- ,tgt\_OI.treasuryControl AS [TreasuryControl]

-- ,tgt\_OI.internal AS [Internal]

-- ,tgt\_OI.internalCounterParty AS [InternalCounterparty]

-- ,tgt\_OI.primeBrokerage AS [PrimeBrokerage]

-- ,Tgt\_OI.reportType AS [ReportType]

-- FROM APP.TGT\_otherInflow tgt\_OI

-- WHERE PID = 'I.O.8'

-- and Portfolio = 'Treasury Investments'

-- GROUP BY

-- Tgt\_OI.reportScope

-- ,tgt\_OI.CompanyName

-- ,Tgt\_OI.Currency

-- ,Tgt\_OI.Converted

-- ,Tgt\_OI.PID

-- ,Tgt\_OI.product

-- ,Tgt\_OI.maturityBucket

-- ,reportType

-- ,tgt\_OI.collateralValue

-- ,tgt\_OI.collateralClass

-- ,tgt\_OI.CollateralCurrency

-- ,tgt\_OI.forwardStartValue

-- ,tgt\_OI.forwardStartBucket

-- ,Tgt\_OI.internal

-- ,tgt\_OI.treasuryControl

-- ,Tgt\_OI.internalCounterParty

-- ,Tgt\_OI.primeBrokerage

-- ,Tgt\_OI.Portfolio

-- -- ,Tgt\_OI.DESC1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - IO9 (Trade Finance), IO6 (Wholesale Funding Transactions)

,IO9 (Commercial and Retail Loans), IO2 NB (Other Payables and margin collaterals)

,IO9 NB (Other Receivable), IO7 Derivatives, I01 Derivatives

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

tgt\_OI.reportScope AS [ReportScope]

,tgt\_OI.Currency AS [Currency]

,tgt\_OI.Converted AS [Converted]

,SUBSTRING(tgt\_OI.PID, 5, 100) AS [PID]

,tgt\_OI.product AS [Product]

,SUM(tgt\_OI.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_OI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OI.maturityBucket,CHARINDEX('-', Tgt\_OI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(tgt\_OI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_OI.collateralClass AS [CollateralClass]

,SUM(tgt\_OI.collateralValue) AS [CollateralValue]

,tgt\_OI.CollateralCurrency AS [CollateralCurrency]

,tgt\_OI.treasuryControl AS [TreasuryControl]

,tgt\_OI.internal AS [Internal]

,tgt\_OI.internalCounterParty AS [InternalCounterparty]

,tgt\_OI.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OI.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otherInflow tgt\_OI

WHERE ((PID = 'I.O.9'

AND Portfolio = 'Trade Finance')

OR

(PID = 'I.O.6'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'I.O.9'

AND Portfolio = 'Commercial and Retail Loans')

OR

(PID = 'I.O.2'

AND Portfolio = 'Other Payables and margin collaterals')

OR

(PID = 'I.O.9'

AND Portfolio = 'Other Receivable')

OR

(PID = 'I.O.6'

AND Portfolio = 'Treasury Investment')

OR

(PID = 'I.O.7'

AND Portfolio = 'Derivatives')

OR

(PID = 'I.O.1'

AND Portfolio = 'Derivatives')

)

GROUP BY

Tgt\_OI.reportScope

,Tgt\_OI.Currency

,Tgt\_OI.Converted

,Tgt\_OI.PID

,Tgt\_OI.product

,Tgt\_OI.maturityBucket

,reportType

,tgt\_OI.collateralValue

,tgt\_OI.collateralClass

,tgt\_OI.CollateralCurrency

,tgt\_OI.forwardStartValue

,tgt\_OI.forwardStartBucket

,Tgt\_OI.internal

,tgt\_OI.treasuryControl

,Tgt\_OI.internalCounterParty

,Tgt\_OI.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: secured Outflows OS1 (Wholesale Funding Transactions)

,OS9 (Wholesale Funding Transactions)

,OS8 Treasury Investment

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,TreasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,settlementMechanism

,Rehypothecated

,[CID]

,[counterPartySector]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_SO.reportScope AS [ReportScope]

,Tgt\_SO.Currency AS [Currency]

,Tgt\_SO.Converted AS [Converted]

,SUBSTRING(tgt\_SO.PID, 5, 100) AS [PID]

,Tgt\_SO.product AS [Product]

,Tgt\_SO.[SID] AS [SID]

,Tgt\_SO.[Subproduct] AS [SubProduct]

,SUM(Tgt\_SO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_SO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_SO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SO.forwardStartBucket,0, CHARINDEX('-', Tgt\_SO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_SO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SO.forwardStartBucket,CHARINDEX('-', Tgt\_SO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_SO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_SO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SO.maturityBucket,CHARINDEX('-', Tgt\_SO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_SO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_SO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_SO.collateralClass AS [CollateralClass]

,SUM(Tgt\_SO.collateralValue) AS [CollateralValue]

,Tgt\_SO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_SO.treasuryControl AS [TreasuryControl]

,Tgt\_SO.internal AS [Internal]

,Tgt\_SO.internalCounterParty AS [InternalCounterparty]

,Tgt\_SO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_SO.settlementMechanism AS [Settlement]

,Tgt\_SO.rehypothecated AS [Rehypothecated]

,Tgt\_SO.CID AS [CID]

,Tgt\_SO.counterPartySector AS [CounterParty]

,Tgt\_SO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_securedOutflow Tgt\_SO

WHERE ((PID = 'O.S.1'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.S.9'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.S.8'

AND Portfolio = 'Treasury Investment')

)

GROUP BY

Tgt\_SO.reportScope

,Tgt\_SO.Currency

,Tgt\_SO.Converted

,Tgt\_SO.PID

,Tgt\_SO.[SID]

,Tgt\_SO.Subproduct

,Tgt\_SO.product

,Tgt\_SO.maturityBucket

,reportType

,Tgt\_SO.collateralClass

,Tgt\_SO.CollateralCurrency

,Tgt\_SO.forwardStartBucket

,Tgt\_SO.internal

,Tgt\_SO.treasuryControl

,Tgt\_SO.internalCounterParty

,Tgt\_SO.primeBrokerage

,Tgt\_SO.settlementMechanism

,Tgt\_SO.rehypothecated

,Tgt\_SO.CID

,Tgt\_SO.counterPartySector

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows - OW9,OW11 (Wholesale Funding Transactions)

, OW19 (Trade Finance)

, OW18 (Other payables and margin collateralss)

, OW19 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[ForwardStartValue]

,[ForwardStartBucket]

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_WO.reportScope AS [ReportScope]

,Tgt\_WO.Currency AS [Currency]

,Tgt\_WO.Converted AS [Converted]

,SUBSTRING(tgt\_WO.PID, 5, 100) AS [PID]

,Tgt\_WO.product AS [Product]

,tgt\_WO.CID AS [CID]

,tgt\_WO.counterPartySector AS [Counterparty]

,SUM(Tgt\_WO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_WO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.forwardStartBucket,0, CHARINDEX('-', Tgt\_WO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.forwardStartBucket,CHARINDEX('-', Tgt\_WO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.maturityBucket,CHARINDEX('-', Tgt\_WO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_WO.collateralClass AS [CollateralClass]

,sum(Tgt\_WO.collateralValue) AS [CollateralValue]

,Tgt\_WO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_WO.forwardStartValue AS [ForwardStart]

,CASE WHEN tgt\_WO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_WO.internal AS [Internal]

,Tgt\_WO.internalCounterParty AS [InternalCounterparty]

,Tgt\_WO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_WO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_wholesaleoutflow tgt\_WO

WHERE ((PID = 'O.W.19'

AND Portfolio = 'Trade Finance'

)

OR

(PID IN ('O.W.9', 'O.W.11')

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID IN ('O.W.18')

AND Portfolio ='Other payables and margin collaterals')

OR

(PID IN ('O.W.19')

AND Portfolio = 'Wholesale Funding Transactions')

)

GROUP BY

Tgt\_WO.reportScope

,Tgt\_WO.Currency

,Tgt\_WO.Converted

,Tgt\_WO.PID

,Tgt\_WO.product

,Tgt\_WO.maturityBucket

,reportType

,Tgt\_WO.collateralValue

,Tgt\_WO.collateralClass

,Tgt\_WO.CollateralCurrency

,Tgt\_WO.forwardStartValue

,Tgt\_WO.forwardStartBucket

,Tgt\_WO.internal

,tgt\_WO.CID

,tgt\_WO.counterPartySector

,Tgt\_WO.internalCounterParty

,Tgt\_WO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows - OW10 & OW16 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[ForwardStartValue]

,[ForwardStartBucket]

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_WO.reportScope AS [ReportScope]

,Tgt\_WO.Currency AS [Currency]

,Tgt\_WO.Converted AS [Converted]

,SUBSTRING(tgt\_WO.PID, 5, 100) AS [PID]

,Tgt\_WO.product AS [Product]

,tgt\_WO.CID AS [CID]

,tgt\_WO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_WO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_WO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.forwardStartBucket,0, CHARINDEX('-', Tgt\_WO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.forwardStartBucket,CHARINDEX('-', Tgt\_WO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.maturityBucket,CHARINDEX('-', Tgt\_WO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_WO.collateralClass AS [CollateralClass]

,sum(Tgt\_WO.collateralValue) AS [CollateralValue]

,Tgt\_WO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_WO.forwardStartValue AS [ForwardStart]

,CASE WHEN tgt\_WO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_WO.internal AS [Internal]

,Tgt\_WO.internalCounterParty AS [InternalCounterparty]

,Tgt\_WO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_WO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_wholesaleoutflow tgt\_WO

WHERE (PID = 'O.W.10' AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.W.16' AND Portfolio = 'Commercial and Retail deposits')

GROUP BY

Tgt\_WO.reportScope

,Tgt\_WO.Currency

,Tgt\_WO.Converted

,Tgt\_WO.PID

,Tgt\_WO.product

,Tgt\_WO.maturityBucket

,reportType

,Tgt\_WO.collateralValue

,Tgt\_WO.collateralClass

,Tgt\_WO.CollateralCurrency

,Tgt\_WO.forwardStartValue

,Tgt\_WO.forwardStartBucket

,Tgt\_WO.internal

,tgt\_WO.CID

,tgt\_WO.counterPartySector

,Tgt\_WO.internalCounterParty

,Tgt\_WO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO2 (Other Payables and margin collaterals), 001

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(Tgt\_OO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE

(PID = 'O.O.2'

AND Portfolio = 'Other Payables and margin collaterals')

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO4 (Credit and Liquidity Facility)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

, SUM(CASE WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_TOTAL |%' THEN ABS(TGT\_OO.maturityValue)

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN (-1)\* ABS(TGT\_OO.maturityValue)

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_OS |%' THEN ABS(TGT\_OO.maturityValue)

END ) AS [MaturityValue]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE (

(PID = 'O.O.4'

AND Portfolio = 'Credit and Liquidity Facility')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO5 (Credit and Liquidity Facility)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

, SUM(CASE WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN ABS(TGT\_OO.maturityValue)

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_OS\_4027 |%' THEN ABS( TGT\_OO.maturityValue)

END ) AS [MaturityValue]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE (

(PID = 'O.O.5'

AND Portfolio = 'Credit and Liquidity Facility')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO7 (Trade Finance), OO19 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE ((PID = 'O.O.7'

AND Portfolio = 'Trade Finance'

)

OR

(PID = 'O.O.19'

AND Portfolio = 'Wholesale Funding Transactions')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO7 (Trade Finance), OO19 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE (PID = 'O.O.19' AND Portfolio = 'Commercial and Retail Deposits')

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO20 (Derivatives)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE PID = 'O.O.20' AND Portfolio = 'Derivatives'

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO1 (Derivatives)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_otheroutflow Tgt\_OO

WHERE PID = 'O.O.1' AND Portfolio = 'Derivatives'

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Operational Deposits

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[Insured]

,[Trigger]

,[Rehypothecated]

,[Internal]

,[internalCounterParty]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_DO.reportScope AS [ReportScope]

,Tgt\_DO.Currency AS [Currency]

,Tgt\_DO.Converted AS [Converted]

,SUBSTRING(Tgt\_DO.PID, 5, 100) AS [PID]

,Tgt\_DO.product AS [Product]

,Tgt\_DO.CID AS [CID]

,Tgt\_DO.counterPartySector AS [Counterparty]

,SUM(Tgt\_DO.maturityValue) AS [MaturityAmount]

,CASE WHEN Tgt\_DO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE Tgt\_DO.maturityBucket

END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

-- THEN 'P0D'

--ELSE

-- SUBSTRING (Tgt\_DO.forwardStartBucket,0, CHARINDEX('-', Tgt\_DO.forwardStartBucket))

--END AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

--THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_DO.forwardStartBucket,CHARINDEX('-', Tgt\_DO.forwardStartBucket)+1,50)

--END AS [ForwardEnd]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_DO.maturityBucket,0, CHARINDEX('-', Tgt\_DO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_DO.maturityBucket,CHARINDEX('-', Tgt\_DO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_DO.collateralClass AS [CollateralClass]

,sum(Tgt\_DO.collateralValue) AS [CollateralValue]

,Tgt\_DO.CollateralCurrency AS [CollateralCurrency]

,tgt\_do.insured AS [Insured]

,Tgt\_DO.[Trigger] AS [Trigger]

,Tgt\_do.Rehypothecated AS [Rehypothecated]

--,Tgt\_DO.forwardStartValue AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_DO.forwardStartBucket

-- END AS [ForwardBucket]

,Tgt\_DO.internal AS [Internal]

,Tgt\_DO.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_DO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_DO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_depositsOutflow Tgt\_DO

WHERE PID NOT IN ('O.D.11')

GROUP BY

Tgt\_DO.reportScope

,Tgt\_DO.Currency

,Tgt\_DO.Converted

,Tgt\_DO.PID

,Tgt\_DO.product

,Tgt\_DO.maturityBucket

,reportType

,Tgt\_DO.collateralValue

,Tgt\_DO.collateralClass

,Tgt\_DO.CollateralCurrency

-- ,Tgt\_DO.forwardStartValue

-- ,Tgt\_DO.forwardStartBucket

,Tgt\_do.Insured

,Tgt\_DO.[Trigger]

,Tgt\_do.Rehypothecated

,Tgt\_DO.internal

,Tgt\_DO.CID

,Tgt\_DO.counterPartySector

,Tgt\_DO.internalCounterParty

--,Tgt\_DO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Operational Deposits

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[Insured]

,[Trigger]

,[Rehypothecated]

,[Internal]

,[internalCounterParty]

,[ReportType]

,asofdate

,creationdate

)

SELECT

Tgt\_DO.reportScope AS [ReportScope]

,Tgt\_DO.Currency AS [Currency]

,Tgt\_DO.Converted AS [Converted]

,SUBSTRING(Tgt\_DO.PID, 5, 100) AS [PID]

,Tgt\_DO.product AS [Product]

,Tgt\_DO.CID AS [CID]

,Tgt\_DO.counterPartySector AS [Counterparty]

,SUM(Tgt\_DO.maturityValue) AS [MaturityAmount]

,CASE WHEN Tgt\_DO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE Tgt\_DO.maturityBucket

END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

-- THEN 'P0D'

--ELSE

-- SUBSTRING (Tgt\_DO.forwardStartBucket,0, CHARINDEX('-', Tgt\_DO.forwardStartBucket))

--END AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

--THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_DO.forwardStartBucket,CHARINDEX('-', Tgt\_DO.forwardStartBucket)+1,50)

--END AS [ForwardEnd]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_DO.maturityBucket,0, CHARINDEX('-', Tgt\_DO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_DO.maturityBucket,CHARINDEX('-', Tgt\_DO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_DO.collateralClass AS [CollateralClass]

,sum(Tgt\_DO.collateralValue) AS [CollateralValue]

,Tgt\_DO.CollateralCurrency AS [CollateralCurrency]

,tgt\_do.insured AS [Insured]

,Tgt\_DO.[Trigger] AS [Trigger]

,Tgt\_do.Rehypothecated AS [Rehypothecated]

--,Tgt\_DO.forwardStartValue AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_DO.forwardStartBucket

-- END AS [ForwardBucket]

,Tgt\_DO.internal AS [Internal]

,Tgt\_DO.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_DO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_DO.reportType AS [ReportType]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM APP.TGT\_depositsOutflow Tgt\_DO

WHERE PID IN ('O.D.11')

GROUP BY

Tgt\_DO.reportScope

,Tgt\_DO.Currency

,Tgt\_DO.Converted

,Tgt\_DO.PID

,Tgt\_DO.product

,Tgt\_DO.maturityBucket

,reportType

,Tgt\_DO.collateralValue

,Tgt\_DO.collateralClass

,Tgt\_DO.CollateralCurrency

-- ,Tgt\_DO.forwardStartValue

-- ,Tgt\_DO.forwardStartBucket

,Tgt\_do.Insured

,Tgt\_DO.[Trigger]

,Tgt\_do.Rehypothecated

,Tgt\_DO.internal

,Tgt\_DO.CID

,Tgt\_DO.counterPartySector

,Tgt\_DO.internalCounterParty

--,Tgt\_DO.primeBrokerage

,AsofDate

,creationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Foreign Exchange

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT INTO [APP].[AGG\_FR2052a]

--(

-- [ReportScope]

-- ,[Currency]

-- ,[Converted]

-- ,[PID]

-- ,[Product]

-- ,[Maturityvalue]

-- ,[MaturityBucket]

-- ,[maturityStart]

-- ,[maturityEnd]

-- ,[Internal]

-- ,[internalCounterParty]

-- ,[ReportType]

--)

-- SELECT

-- Tgt\_FX.reportScope AS [ReportScope]

-- ,Tgt\_FX.Currency\_1 AS [Currency]

-- ,Tgt\_FX.Currency\_1\_Converted AS [Converted]

-- ,SUBSTRING(Tgt\_FX.PID, 6, 100) AS [PID]

-- ,Tgt\_FX.product AS [Product]

-- ,SUM(ABS(Tgt\_FX.[MaturityAmountCurrency1])) AS [MaturityAmount]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_FX.maturityBucket

-- END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE

-- SUBSTRING (Tgt\_FX.maturityBucket,0, CHARINDEX('-', Tgt\_FX.maturityBucket))

-- END AS [maturityStart]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_FX.maturityBucket,CHARINDEX('-', Tgt\_FX.maturityBucket)+1,50)

-- END AS [maturityEnd]

-- ,Tgt\_FX.internal AS [Internal]

-- ,Tgt\_FX.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_FX.reportType AS [ReportType]

-- FROM APP.TGT\_ForeignExchange Tgt\_FX

-- GROUP BY

-- Tgt\_FX.reportScope

-- ,Tgt\_FX.Currency\_1

-- ,Tgt\_FX.Currency\_1\_Converted

-- ,Tgt\_FX.PID

-- ,Tgt\_FX.product

-- ,Tgt\_FX.maturityBucket

-- ,reportType

-- ,Tgt\_FX.internal

-- ,Tgt\_FX.internalCounterParty

-- ,Tgt\_FX.reportType

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Supplemental Informational

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_FR2052a]

(

[ReportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[sid2]

,[Subproduct2]

,[MarketValue]

,[CollateralClass]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,asofdate

,creationdate

)

SELECT

Tgt\_IS.reportScope AS [ReportScope]

,Tgt\_IS.Currency AS [Currency]

,Tgt\_IS.Converted AS [Converted]

,SUBSTRING(Tgt\_IS.PID, 5, 100) AS [PID]

,Tgt\_IS.product AS [Product]

,Tgt\_IS.[SID] AS [SID]

,Tgt\_IS.[subProduct] AS [SubProduct]

,Tgt\_IS.SID2 AS [SID2]

,Tgt\_IS.[subProduct2] AS [SubProduct2]

,SUM(Tgt\_IS.marketValue) AS [MarketValue]

,Tgt\_IS.collateralClass AS [CollateralClass]

,Tgt\_IS.internal AS [Internal]

,Tgt\_IS.internalCounterParty AS [InternalCounterParty]

,Tgt\_IS.reportType AS [ReportType]

,Tgt\_IS.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [Asofdate]

,creationDate AS [CreationDate]

FROM

APP.TGT\_informationalSupplemental Tgt\_IS

WHERE PID IN ('S.I.2','S.I.5','S.I.15','S.I.16','S.I.17','S.I.18')

GROUP BY

Tgt\_IS.reportScope

,Tgt\_IS.Currency

,Tgt\_IS.Converted

,Tgt\_IS.PID

,Tgt\_IS.product

,Tgt\_IS.[SID]

,Tgt\_IS.[subProduct]

,Tgt\_IS.SID2

,Tgt\_IS.[subProduct2]

,Tgt\_IS.collateralClass

,Tgt\_IS.internal

,Tgt\_IS.internalCounterParty

,Tgt\_IS.reportType

,Tgt\_IS.primeBrokerage

,AsofDate

,creationDate

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

--Call the procedure that checks all the tables for FR validations

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC [dbo].[Proc\_InsertIssueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Agg\_BOC] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Agg\_BOC]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from target tables to aggregate tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Agg\_BOC]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

DROP INDEX IX\_Tgt\_AssetInflow ON app.tgt\_assetInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

ON app.tgt\_assetInflow(TGT\_assetInflowID);

SET NOCOUNT ON

BEGIN TRY

--Insert scripts for aggregate table

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[DESC1]

,[PID]

,[Portfolio]

,[Product]

,[SID]

,[SubProduct]

,[lendableValue]

,[MarketValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[Forwardstartvalue]

,[ForwardStart]

,[ForwardEnd]

,[ReportType]

,[CollateralClass]

,[TreasuryControl]

,AsofDate

,CreationDate

)

--Code for cash - (not aggregating monetary values from ref tables)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.1 (BOLI), I.A.4 (CASH), I.A.5 & I.A.6 (Treasury Investment)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,(SUM(ABS(Tgt\_AI.LendableValue))) AS [LendableValue]

,(sum(ABS(tgt\_AI.marketValue))) AS [MarketValue]

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

[app].[TGT\_assetInflow] tgt\_AI

WHERE ((PID = 'I.A.1'

AND Portfolio = 'BOLI')

OR

(PID IN ('I.A.4')

AND Portfolio = 'CASH')

OR

(PID IN ('I.A.5', 'I.A.6')

AND Portfolio = 'Treasury Investment')

)

GROUP BY tgt\_AI.reportScope

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.DESC1

,tgt\_AI.PID

,tgt\_AI.Portfolio

,tgt\_AI.Product

,tgt\_AI.[SID]

,TGT\_AI.SubProduct

,tgt\_AI.maturityBucket

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,tgt\_AI.CompanyName

,Tgt\_AI.reportType

,[ForwardStartValue]

,forwardStartBucket

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.3 (CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION SELECT DISTINCT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,SUM(ABS(TGT\_AI.marketValue)) - (SELECT SUM(ABS(TGT\_AI.marketValue))

FROM app.Tgt\_assetInflow TGT\_AI

WHERE DESC1 = '9003'

AND PID ='I.A.4') AS [lendableValue]

,SUM(ABS(TGT\_AI.marketValue)) - (SELECT SUM(ABS(TGT\_AI.marketValue))

FROM app.Tgt\_assetInflow TGT\_AI

WHERE DESC1 = '9003'

AND PID ='I.A.4') AS [MarketValue]

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(ABS(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

[app].[TGT\_assetInflow] tgt\_AI

WHERE (

(PID IN ('I.A.3')

AND Portfolio = 'CASH')

)

and marketValue is not null

and lendableValue is not null

and marketValue <> 0

and lendableValue <> 0

and tgt\_ai.DESC1 = '0060'

GROUP BY tgt\_AI.reportScope

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.DESC1

,tgt\_AI.PID

,tgt\_AI.Portfolio

,tgt\_AI.Product

,tgt\_AI.[SID]

,TGT\_AI.SubProduct

,tgt\_AI.maturityBucket

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,tgt\_AI.CompanyName

,Tgt\_AI.reportType

,[ForwardStartValue]

,forwardStartBucket

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflow - I.A.1 (Treasury Investment)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--=================================================================================================

--Create temp table containing copy of data from TGT\_assetInflow table

IF OBJECT\_ID('tempdb..#Temp')IS NOT NULL DROP TABLE #Temp

SELECT \*

INTO #Temp

FROM [app].[TGT\_assetInflow]

WHERE PID = 'I.A.1'

AND Portfolio = 'Treasury Investment'

--Test temp table

--SELECT \* FROM #tEMP WHERE SrcID LIKE '%US94974BGG95%' OR SrcID LIKE '%US94974BGS34%'

--Create fields holding Security ID and Branch (BR) information. This will be used to perform calculations at Security and Branch level

IF OBJECT\_ID('tempdb..#Temp\_A')IS NOT NULL DROP TABLE #Temp\_A

SELECT

CASE LEFT(A.SRCID, 4)

WHEN 'TPOS' THEN SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

ELSE SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

END AS SECID

, SUBSTRING(SrcID, 6, 2) AS BR

, A.\*

INTO #Temp\_A

FROM #Temp A

--==================================================================================================

--Create aggregate of TPOS and SSAH based on subqueries aggregating TPOS and SSAH separately

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[DESC1]

,[PID]

,[Portfolio]

,[Product]

,[SID]

,[SubProduct]

,[lendableValue]

,[MarketValue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[Forwardstartvalue]

,[ForwardStartBucket]

,[ForwardStart]

,[ForwardEnd]

,[ReportType]

,[CollateralClass]

,[TreasuryControl]

,AsofDate

,CreationDate

)

SELECT

tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,tgt\_AI.PID AS [PID]

,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.product AS [Product]

,tgt\_AI.[SID] AS [SID]

,tgt\_AI.SubProduct AS [SubProduct]

,SUM(CAST(Tgt\_AI.LendableValue AS DECIMAL (30,15))) AS [LendableValue]

,SUM(CAST(tgt\_AI.marketValue AS DECIMAL (30,15))) AS [MarketValue]

,TGT\_AI.maturityBucket AS [MaturityBucket]

,Tgt\_AI.maturityStart AS [maturityStart]

,Tgt\_AI.maturityEnd AS [maturityEnd]

,Tgt\_ai.ForwardStartValue AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,tgt\_AI.ForwardStart AS [ForwardStart]

,tgt\_AI.ForwardEnd AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,tgt\_AI.AsofDate AS [AsOfDate]

,tgt\_AI.creationDate AS [CreationDate]

FROM (

SELECT

TPOS.SECID

,TPOS.BR

,TPOS.reportScope AS [ReportScope]

,TPOS.CompanyName AS [CompanyName]

,TPOS.Currency AS [Currency]

,TPOS.Converted AS [Converted]

,TPOS.DESC1 AS [DESC1]

,TPOS.PID AS [PID]

,TPOS.Portfolio AS [Portfolio]

,TPOS.Product AS [Product]

,TPOS.[SID] AS [SID]

,TPOS.SubProduct AS [Subproduct]

,SUM(ABS(CAST(TPOS.LendableValue AS DECIMAL (30,15)))) AS [TPOS\_LendableValue]

,SUM(ABS(CAST(TPOS.marketValue AS DECIMAL(30,15)))) AS [TPOS\_MarketValue]

,SUM(TPOS.QTY) AS TPOS\_QTY

,SUM(SSAH.QTY) AS SSAH\_QTY

,((SUM(CAST(TPOS.QTY AS DECIMAL (30,15))) - SUM(isnull(CAST(SSAH.QTY AS DECIMAL(30,15)),0))) / SUM(CAST(TPOS.QTY AS DECIMAL (30,15)))) \* SUM(ABS(CAST(TPOS.marketValue AS DECIMAL(30,15)))) AS [MarketValue]

,(((SUM(CAST(TPOS.QTY AS DECIMAL (30,15))) - SUM(isnull(CAST(SSAH.QTY AS DECIMAL(30,15)),0))) / SUM(CAST(TPOS.QTY AS DECIMAL (30,15)))) \* SUM(ABS(CAST(TPOS.marketValue AS DECIMAL (30,15))))) \* (1-(CAST(SECM.SYMBOLID AS FLOAT)/100)) AS [LendableValue]

,TPOS.[MaturityBucket]

,TPOS.[maturityStart]

,TPOS.[maturityEnd]

,ABS(SUM(TPOS.forwardStartValue)) AS [ForwardStartValue]

,TPOS.[ForwardStart]

,TPOS.[ForwardEnd]

,TPOS.reportType AS [ReportType]

,TPOS.collateralClass AS [CollateralClass]

,TPOS.treasuryControl AS [TreasuryControl]

,TPOS.AsofDate AS [AsOfDate]

,TPOS.creationDate AS [CreationDate]

FROM

(

---SUBQUERY to get Aggregate of TPOS at Security and BR levels

SELECT

tgt\_AI.SECID

, tgt\_AI.BR

, tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,SUM(CAST(Tgt\_AI.LendableValue AS DECIMAL(30,15))) AS [LendableValue]

,SUM(CAST(tgt\_AI.marketValue AS DECIMAL(30,15))) AS [MarketValue]

,SUM(CAST(tgt\_AI.QTY AS DECIMAL(30,15))) AS QTY

--,(SUM(CASE WHEN LEFT(tgt\_AI.SRCID, 4) = 'TPOS' THEN tgt\_AI.QTY ELSE (-1 \* tgt\_AI.QTY) END)/SUM(CASE WHEN LEFT(tgt\_AI.SRCID, 4) = 'TPOS' THEN tgt\_AI.QTY ELSE 0 END)\*(SUM(CAST(ISNULL(tgt\_AI.[marketValue],1)AS DECIMAL(30,15))))) AS apportionedMarketValueScript

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM #Temp\_A tgt\_AI

WHERE tgt\_AI.PID = 'I.A.1' AND LEFT(tgt\_AI.SRCID, 4) = 'TPOS' --AND tgt\_AI.SECID IN ('US94974BGG95', 'US94974BGS34')

GROUP BY

tgt\_AI.SECID, tgt\_AI.BR

, tgt\_AI.reportScope, tgt\_AI.Currency, tgt\_AI.Converted, tgt\_AI.DESC1, tgt\_AI.PID, tgt\_AI.Portfolio, tgt\_AI.Product, tgt\_AI.[SID], TGT\_AI.SubProduct, tgt\_AI.maturityBucket

, tgt\_AI.collateralClass, tgt\_AI.treasuryControl, tgt\_AI.CompanyName, Tgt\_AI.reportType, tgt\_AI.forwardStartValue, tgt\_AI.forwardStartBucket, Tgt\_AI.AsofDate, CREATIONDATE

) AS TPOS

LEFT JOIN

(

---SUBQUERY to get Aggregate of SSAH at Security and BR levels

SELECT

tgt\_AI.SECID

,tgt\_AI.BR

,tgt\_AI.reportScope AS [ReportScope]

,tgt\_AI.CompanyName AS [CompanyName]

,tgt\_AI.Currency AS [Currency]

,tgt\_AI.Converted AS [Converted]

,tgt\_AI.DESC1 AS [DESC1]

,SUBSTRING(tgt\_AI.PID, 5, 100) AS [PID]

,tgt\_AI.Portfolio AS [Portfolio]

,tgt\_AI.Product AS [Product]

,tgt\_AI.[SID] AS [SID]

,TGT\_AI.SubProduct AS [Subproduct]

,SUM(CAST (Tgt\_AI.LendableValue AS DECIMAL(30,15))) AS [LendableValue]

,SUM(CAST(tgt\_AI.marketValue AS DECIMAL(30,15))) AS [MarketValue]

,SUM(CAST(tgt\_AI.QTY AS DECIMAL(30,15))) AS QTY

,CASE WHEN tgt\_AI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_AI.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,0, CHARINDEX('-', TGT\_AI.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_AI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_AI.maturityBucket,CHARINDEX('-', TGT\_AI.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_AI.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,0, CHARINDEX('-', Tgt\_AI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_AI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_AI.forwardStartBucket,CHARINDEX('-', Tgt\_AI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,Tgt\_AI.reportType AS [ReportType]

,tgt\_AI.collateralClass AS [CollateralClass]

,tgt\_AI.treasuryControl AS [TreasuryControl]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM #Temp\_A tgt\_AI

WHERE tgt\_AI.PID = 'I.A.1' AND LEFT(tgt\_AI.SRCID, 4) = 'SSAH' --AND tgt\_AI.SECID IN ('US94974BGG95', 'US94974BGS34') --AND LEFT(tgt\_AI\_SSAH.SRCID, 4) = 'SSAH'

GROUP BY

tgt\_AI.SECID, tgt\_AI.BR

, tgt\_AI.reportScope, tgt\_AI.Currency, tgt\_AI.Converted, tgt\_AI.DESC1, tgt\_AI.PID, tgt\_AI.Portfolio, tgt\_AI.Product, tgt\_AI.[SID], TGT\_AI.SubProduct, tgt\_AI.maturityBucket

, tgt\_AI.collateralClass, tgt\_AI.treasuryControl, tgt\_AI.CompanyName, Tgt\_AI.reportType, tgt\_AI.forwardStartValue, tgt\_AI.forwardStartBucket, TGT\_AI.AsofDate, CREATIONDATE

) AS SSAH

ON

TPOS.reportScope = SSAH.reportScope

AND TPOS.BR = SSAH.BR

AND TPOS.SECID = SSAH.SECID

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.AsofDate = SECM.ASOFDATE

GROUP BY

TPOS.SECID, TPOS.BR

, TPOS.reportScope, TPOS.Currency, TPOS.Converted, TPOS.DESC1, TPOS.PID, TPOS.Portfolio, TPOS.Product, TPOS.[SID], TPOS.SubProduct, TPOS.maturityBucket

, TPOS.maturityStart, TPOS.maturityEnd, TPOS.ForwardStart, TPOS.ForwardEnd, TPOS.collateralClass, TPOS.treasuryControl, TPOS.CompanyName, TPOS.reportType,TPOS.AsofDate, SECM.SYMBOLID, TPOS.ASOFDATE, TPOS.CREATIONDATE

) tgt\_AI

GROUP BY

tgt\_AI.reportScope

,tgt\_AI.CompanyName

,tgt\_AI.Currency

,tgt\_AI.Converted

,tgt\_AI.DESC1

,tgt\_AI.PID

,tgt\_AI.Portfolio

,tgt\_AI.product

,tgt\_AI.[SID]

,tgt\_AI.SubProduct

,TGT\_AI.maturityBucket

,Tgt\_AI.maturityStart

,Tgt\_AI.maturityEnd

,Tgt\_ai.ForwardStartValue

,tgt\_AI.ForwardStart

,tgt\_AI.ForwardEnd

,Tgt\_AI.reportType

,tgt\_AI.collateralClass

,tgt\_AI.treasuryControl

,tgt\_AI.AsofDate

,tgt\_AI.CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU1, IU2 (Wholesale Funding Transactions)

, IU3 NB (Other Receivable),

IU6 NB (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Portfolio]

,[DESC1]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_UI.reportScope AS [ReportScope]

,tgt\_UI.CompanyName AS [CompanyName]

,TGT\_UI.Portfolio AS [Portfolio]

,tgt\_UI.DESC1 AS [DESC1]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

,sum(Tgt\_UI.maturityValue) AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_UI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

[app].Tgt\_unsecuredInflow Tgt\_UI

WHERE (

(PID IN ('I.U.1' , 'I.U.2')

AND Portfolio = 'Wholesale Funding Transactions')

OR

( PID IN ('I.U.3')

AND Portfolio = 'Other Receivable')

OR

( PID IN ('I.U.6')

AND Portfolio = 'Wholesale Funding Transactions')

)

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,Tgt\_UI.DESC1

,Tgt\_UI.CompanyName

,Tgt\_UI.Portfolio

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU5,IU6 (Commercial and Retail Loans)

IU6 (Trade Finance), IU3(CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Portfolio]

,[DESC1]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,AsofDate

,CreationDate

)

SELECT

Tgt\_UI.reportScope AS [ReportScope]

,tgt\_UI.CompanyName AS [CompanyName]

,TGT\_UI.Portfolio AS [Portfolio]

,tgt\_UI.DESC1 AS [DESC1]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

,sum(ABS(Tgt\_UI.maturityValue)) AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_UI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

[app].Tgt\_unsecuredInflow Tgt\_UI

WHERE ((PID IN ('I.U.5','I.U.6')

AND Portfolio = 'Commercial and Retail Loans')

OR

(PID = 'I.U.6'

AND Portfolio = 'Trade Finance'

)

OR

( PID IN ('I.U.3')

AND Portfolio = 'Cash')

)

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,Tgt\_UI.DESC1

,Tgt\_UI.CompanyName

,Tgt\_UI.Portfolio

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - IU4 (CASH)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Portfolio]

,[DESC1]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[forwardStartValue]

,[forwardStartBucket]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

,AsofDate

,CreationDate

)

SELECT

Tgt\_UI.reportScope AS [ReportScope]

,tgt\_UI.CompanyName AS [CompanyName]

,TGT\_UI.Portfolio AS [Portfolio]

,tgt\_UI.DESC1 AS [DESC1]

,Tgt\_UI.Currency AS [Currency]

,Tgt\_UI.Converted AS [Converted]

,SUBSTRING(tgt\_UI.PID, 5, 100) AS [PID]

,TGT\_UI.product AS [Product]

,tgt\_UI.[CID] AS [CID]

,tgt\_UI.[counterPartySector] AS [counterPartySector]

, SUM(CASE WHEN TGT\_UI.SrcID like 'FMD\_ALL | %' THEN (-1)\*TGT\_UI.maturityValue

WHEN TGT\_UI.SrcID like 'MULTI%' THEN (-1)\* TGT\_UI.maturityValue

WHEN TGT\_UI.SrcID like 'T\_FMDDATA | %' THEN (-1)\* ABS(TGT\_UI.maturityValue)

END ) AS [MaturityValue]

-- ,CASE WHEN tgt\_ui.SrcID like 'FMD\_ALL | %' THEN SUM(TGT\_UI.maturityValue) -

-- (CASE WHEN tgt\_ui.SrcID like 'MULTI%' THEN SUM(TGT\_UI.maturityValue) -

--(CASE WHEN tgt\_ui.SrcID like 'FMD | %' THEN SUM(TGT\_UI.maturityValue) ELSE 0 END

-- )

-- ELSE 0 END )

--ELSE 0 END AS [MarketValue]

,CASE WHEN tgt\_UI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_UI.forwardStartBucket,0, CHARINDEX('-', Tgt\_UI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_UI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_UI.forwardStartBucket,CHARINDEX('-', Tgt\_UI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (TGT\_UI.maturityBucket,0, CHARINDEX('-', TGT\_UI.maturityBucket))

END AS [maturityStart]

,CASE WHEN TGT\_UI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (TGT\_UI.maturityBucket,CHARINDEX('-', TGT\_UI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_UI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_UI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_UI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_UI.internal AS [Internal]

,tgt\_UI.internalCounterParty AS [InternalCounterParty]

,Tgt\_UI.reportType AS [ReportType]

,tgt\_UI.primeBrokerage AS [PrimeBrokerage]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

[app].Tgt\_unsecuredInflow Tgt\_UI

WHERE PID IN ('I.U.4')

AND Portfolio = 'Cash'

GROUP BY

tgt\_UI.[CID]

,tgt\_UI.reportScope

,tgt\_UI.Currency

,tgt\_UI.Converted

,tgt\_UI.PID

,tgt\_UI.product

,tgt\_UI.maturityBucket

,reportType

,tgt\_UI.[counterPartySector]

,tgt\_UI.internal

,tgt\_UI.internalCounterParty

,tgt\_UI.primeBrokerage

,Tgt\_UI.forwardStartBucket

,Tgt\_UI.DESC1

,Tgt\_UI.CompanyName

,Tgt\_UI.Portfolio

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Inflows IS1 (Wholesale Funding Transactions), IS7 (Commercial and Retail Loans)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,CompanyName

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[SID]

,[SUbProduct]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[EffectiveMaturityBucket]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Unencumbered]

,treasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[Settlementmechanism]

,[CID]

,[counterPartySector]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT

Tgt\_SI.reportScope AS [ReportScope]

,tgt\_SI.CompanyName AS [CompanyName]

,Tgt\_SI.Currency AS [Currency]

,Tgt\_SI.Converted AS [Converted]

,SUBSTRING(tgt\_SI.PID, 5, 100) AS [PID]

,Tgt\_SI.product AS [Product]

,tgt\_SI.DESC1 AS [DESC1]

,tgt\_SI.Portfolio AS [Portfolio]

,tgt\_SI.[SID] AS [SID]

,tgt\_SI.[SubProduct] AS [SubProduct]

,SUM(Tgt\_SI.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_SI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_SI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SI.forwardStartBucket,0, CHARINDEX('-', Tgt\_SI.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_SI.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SI.forwardStartBucket,CHARINDEX('-', Tgt\_SI.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_SI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SI.maturityBucket,0, CHARINDEX('-', Tgt\_SI.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_SI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SI.maturityBucket,CHARINDEX('-', Tgt\_SI.maturityBucket)+1,50)

END AS [maturityEnd]

,tgt\_SI.EffectiveMaturityBucket AS [EffectiveMaturityDate]

,SUM(Tgt\_SI.forwardStartValue) AS [Forward Start]

,CASE WHEN tgt\_SI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SI.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_SI.collateralClass AS [CollateralClass]

,SUM(Tgt\_SI.collateralValue) AS [CollateralValue]

,Tgt\_SI.CollateralCurrency AS [CollateralCurrency]

,tgt\_SI.Unencumbered AS [Unencumbered]

,Tgt\_SI.treasuryControl AS [TreasuryControl]

,Tgt\_SI.internal AS [Internal]

,Tgt\_SI.internalCounterParty AS [InternalCounterparty]

,Tgt\_SI.primeBrokerage AS [PrimeBrokerage]

,tgt\_SI.settlementMechanism AS [Settlement]

,tgt\_SI.CID AS [CID]

,tgt\_SI.counterPartySector AS [COunterParty]

,Tgt\_SI.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_securedInflow tgt\_SI

WHERE ((PID = 'I.S.1'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'I.S.7'

AND Portfolio = 'Commercial and Retail Loans'))

GROUP BY

Tgt\_SI.reportScope

,Tgt\_SI.Currency

,Tgt\_SI.Converted

,Tgt\_SI.PID

,Tgt\_SI.product

,tgt\_SI.DESC1

,tgt\_SI.[SID]

,tgt\_SI.Portfolio

,tgt\_SI.[SubProduct]

,Tgt\_SI.maturityBucket

,tgt\_SI.EffectiveMaturityBucket

,reportType

,tgt\_SI.CID

,tgt\_SI.counterPartySector

,Tgt\_SI.collateralClass

,Tgt\_SI.CollateralCurrency

,tgt\_SI.Unencumbered

,Tgt\_SI.forwardStartBucket

,tgt\_SI.settlementMechanism

,Tgt\_SI.internal

,Tgt\_SI.treasuryControl

,Tgt\_SI.internalCounterParty

,Tgt\_SI.primeBrokerage

,tgt\_SI.CompanyName

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - IO8 (Treasury Investments)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT INTO app.AGG\_BOC

--(

-- [ReportScope]

-- ,[CompanyName]

-- ,[Currency]

-- ,[Converted]

-- ,[PID]

-- ,[Product]

-- ,[Maturityvalue]

-- ,[MaturityBucket]

-- ,[maturityStart]

-- ,[maturityEnd]

-- ,[ForwardStartValue]

-- ,[ForwardStartBucket]

-- ,collateralClass

-- ,collateralValue

-- ,CollateralCurrency

-- ,treasuryControl

-- ,[Internal]

-- ,[internalCounterParty]

-- ,[primeBrokerage]

-- ,[ReportType]

-- ,[Portfolio]

-- ,[DESC1]

--)

-- SELECT DISTINCT

-- tgt\_OI.reportScope AS [ReportScope]

-- ,tgt\_OI.CompanyName AS [CompanyName]

-- ,tgt\_OI.Currency AS [Currency]

-- ,tgt\_OI.Converted AS [Converted]

-- ,SUBSTRING(tgt\_OI.PID, 5, 100) AS [PID]

-- ,tgt\_OI.product AS [Product]

-- ,SUM(ABS(tgt\_OI.maturityValue)) AS [MaturityAmount]

-- ,CASE WHEN tgt\_OI.maturityBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.maturityBucket

-- END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE

-- SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket))

-- END AS [maturityStart]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_OI.maturityBucket,CHARINDEX('-', Tgt\_OI.maturityBucket)+1,50)

-- END AS [maturityEnd]

-- ,SUM(tgt\_OI.forwardStartValue) AS [ForwardStart]

-- ,CASE WHEN tgt\_OI.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.forwardStartBucket

-- END AS [ForwardBucket]

-- ,tgt\_OI.collateralClass AS [CollateralClass]

-- ,SUM(tgt\_OI.collateralValue) AS [CollateralValue]

-- ,tgt\_OI.CollateralCurrency AS [CollateralCurrency]

-- ,tgt\_OI.treasuryControl AS [TreasuryControl]

-- ,tgt\_OI.internal AS [Internal]

-- ,tgt\_OI.internalCounterParty AS [InternalCounterparty]

-- ,tgt\_OI.primeBrokerage AS [PrimeBrokerage]

-- ,Tgt\_OI.reportType AS [ReportType]

-- ,Tgt\_OI.Portfolio AS [Portfolio]

-- ,Tgt\_OI.DESC1 AS [DESC1]

-- FROM app.tgt\_otherInflow tgt\_OI

-- WHERE PID = 'I.O.8'

-- and Portfolio = 'Treasury Investments'

-- GROUP BY

-- Tgt\_OI.reportScope

-- ,tgt\_OI.CompanyName

-- ,Tgt\_OI.Currency

-- ,Tgt\_OI.Converted

-- ,Tgt\_OI.PID

-- ,Tgt\_OI.product

-- ,Tgt\_OI.maturityBucket

-- ,reportType

-- ,tgt\_OI.collateralValue

-- ,tgt\_OI.collateralClass

-- ,tgt\_OI.CollateralCurrency

-- ,tgt\_OI.forwardStartValue

-- ,tgt\_OI.forwardStartBucket

-- ,Tgt\_OI.internal

-- ,tgt\_OI.treasuryControl

-- ,Tgt\_OI.internalCounterParty

-- ,Tgt\_OI.primeBrokerage

-- ,Tgt\_OI.Portfolio

-- ,Tgt\_OI.DESC1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - IO8 (Treasury Investments)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

----Create temp table containing copy of data from TGT\_assetInflow table

IF OBJECT\_ID('tempdb..#Temp\_IO8')IS NOT NULL DROP TABLE #Temp\_IO8

SELECT \*

INTO #Temp\_IO8

FROM [app].[TGT\_otherInflow]

WHERE PID = 'I.O.8'

AND Portfolio = 'Treasury Investment'

--Test temp table

--SELECT \* FROM #tEMP WHERE SrcID LIKE '%US94974BGG95%' OR SrcID LIKE '%US94974BGS34%'

--Create fields holding Security ID and Branch (BR) information. This will be used to perform calculations at Security and Branch level

IF OBJECT\_ID('tempdb..#Temp\_A\_IO8')IS NOT NULL DROP TABLE #Temp\_A\_IO8

SELECT

CASE LEFT(A.SRCID, 4)

WHEN 'TPOS' THEN SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

ELSE SUBSTRING(SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)+1) - CHARINDEX('|', SrcID, CHARINDEX('|', SrcID)+1)-1)

END AS SECID

, SUBSTRING(SrcID, 6, 2) AS BR

, A.\*

INTO #Temp\_A\_IO8

FROM #Temp\_IO8 A

--==================================================================================================

--Create aggregate of TPOS and SSAH based on subqueries aggregating TPOS and SSAH separately

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,CollateralCurrency

,treasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,collateralClass

,collateralValue

,[Portfolio]

,[DESC1]

,AsofDate

,CreationDate

)

SELECT

tgt\_IO\_outer.reportScope AS [ReportScope]

,tgt\_IO\_outer.CompanyName AS CompanyName

,tgt\_IO\_outer.Currency AS [Currency]

,tgt\_IO\_outer.Converted AS [Converted]

,tgt\_IO\_outer.PID AS [PID]

,tgt\_IO\_outer.product AS [Product]

,SUM(CAST(tgt\_IO\_outer.maturityValue AS DECIMAL (30,15))) AS [maturityValue]

,tgt\_IO\_outer.maturityBucket AS [MaturityBucket]

,tgt\_IO\_outer.maturityStart AS [maturityStart]

,tgt\_IO\_outer.maturityEnd AS [maturityEnd]

,tgt\_IO\_outer.ForwardStartValue AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS CollateralCurrency

,tgt\_IO\_outer.treasuryControl AS [TreasuryControl]

,tgt\_IO\_outer.Internal AS Internal

,tgt\_IO\_outer.InternalCounterParty AS InternalCounterParty

,tgt\_IO\_outer.PrimeBrokage AS primeBrokage

,tgt\_IO\_outer.ReportType AS [ReportType]

,tgt\_IO\_outer.collateralClass AS [CollateralClass]

,SUM(CAST(tgt\_IO\_outer.maturityValue AS DECIMAL (30,15))) AS [CollateralValue]

,tgt\_IO\_outer.Portfolio AS [Portfolio]

,tgt\_IO\_outer.DESC1 AS [DESC1]

,tgt\_IO\_outer.AsofDate AS [AsOfDate]

,tgt\_IO\_outer.creationDate AS [CreationDate]

FROM (

SELECT

TPOS.SECID

,TPOS.BR

,TPOS.reportScope AS [ReportScope]

,TPOS.CompanyName AS [CompanyName]

,TPOS.Currency AS [Currency]

,TPOS.Converted AS [Converted]

,TPOS.DESC1 AS [DESC1]

,TPOS.PID AS [PID]

,TPOS.Portfolio AS [Portfolio]

,TPOS.Product AS [Product]

--,TPOS.[SID] AS [SID]

--,TPOS.SubProduct AS [Subproduct]

--,SUM(ABS(TPOS.LendableValue)) AS [TPOS\_LendableValue]

,SUM(ABS(CAST(TPOS.maturityValue AS DECIMAL (30,15)))) AS [TPOS\_maturityValue]

,SUM(CAST(TPOS.Original\_QTY AS DECIMAL (30,15))) AS TPOS\_QTY

,SUM(CAST(SSAH.Original\_QTY AS DECIMAL (30,15))) AS SSAH\_QTY

,((SUM(CAST(TPOS.Original\_QTY AS DECIMAL (30,15))) - SUM(isnull(CAST(SSAH.Original\_QTY AS DECIMAL (30,15)),0))) / SUM(CAST(TPOS.Original\_QTY AS DECIMAL (30,15)))) \* SUM(ABS(CAST(TPOS.maturityValue AS DECIMAL(30,15)))) AS MaturityValue

--,(((SUM(TPOS.Original\_QTY) - SUM(isnull(SSAH.Original\_QTY,0))) / SUM(TPOS.Original\_QTY)) \* SUM(ABS(TPOS.maturityValue))) \* (1-(ref\_el.Value/100)) AS [LendableValue]

,TPOS.[MaturityBucket]

,TPOS.[maturityStart]

,TPOS.[maturityEnd]

,ABS(SUM(TPOS.forwardStartValue)) AS [ForwardStartValue]

,TPOS.[ForwardStart]

,TPOS.[ForwardEnd]

,TPOS.reportType AS [ReportType]

,TPOS.collateralClass AS [CollateralClass]

,TPOS.treasuryControl AS [TreasuryControl]

,TPOS.Internal AS [Internal]

,TPOS.InternalCounterParty AS [InternalCounterParty]

,TPOS.PrimeBrokage AS [PrimeBrokage]

,TPOS.AsofDate AS [AsOfDate]

,TPOS.creationDate AS [CreationDate]

FROM

(

---SUBQUERY to get Aggregate of TPOS at Security and BR levels

SELECT

tgt\_IO.SECID

,tgt\_IO.BR

,tgt\_IO.reportScope AS [ReportScope]

,tgt\_IO.CompanyName AS [CompanyName]

,tgt\_IO.Currency AS [Currency]

,tgt\_IO.Converted AS [Converted]

,tgt\_IO.DESC1 AS [DESC1]

,SUBSTRING(tgt\_IO.PID, 5, 100) AS [PID]

,tgt\_IO.Portfolio AS [Portfolio]

,tgt\_IO.Product AS [Product]

--,tgt\_IO.[SID] AS [SID]

--,tgt\_IO.SubProduct AS [Subproduct]

--,SUM(tgt\_IO.LendableValue) AS [LendableValue]

, tgt\_IO.[CollateralClass] AS [CollateralClass]

,SUM(CAST(tgt\_IO.maturityValue AS DECIMAL (30,15))) AS [maturityValue]

,SUM(CAST(tgt\_IO.QTY AS DECIMAL (30,15))) AS Original\_QTY

,Tgt\_IO.reportType AS reportType

,tgt\_IO.treasuryControl AS TreasuryControl

,Tgt\_IO.internal AS Internal

,Tgt\_IO.internalCounterParty AS internalCounterParty

,Tgt\_IO.primeBrokerage AS primeBrokage

--,(SUM(CASE WHEN LEFT(tgt\_IO.SRCID, 4) = 'TPOS' THEN tgt\_IO.QTY ELSE (-1 \* tgt\_IO.QTY) END)/SUM(CASE WHEN LEFT(tgt\_IO.SRCID, 4) = 'TPOS' THEN tgt\_IO.QTY ELSE 0 END)\*(SUM(CAST(ISNULL(tgt\_IO.[maturityValue],1)AS DECIMAL(30,15))))) AS apportionedmaturityValueScript

,CASE WHEN tgt\_IO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_IO.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,0, CHARINDEX('-', tgt\_IO.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,CHARINDEX('-', tgt\_IO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(abs(tgt\_IO.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,0, CHARINDEX('-', tgt\_IO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,CHARINDEX('-', tgt\_IO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM #Temp\_A\_IO8 tgt\_IO

WHERE tgt\_IO.PID = 'I.O.8' AND LEFT(tgt\_IO.SRCID, 4) = 'TPOS' --AND tgt\_IO.SECID IN ('US94974BGG95', 'US94974BGS34')

GROUP BY

tgt\_IO.SECID, tgt\_IO.BR

, tgt\_IO.reportScope, tgt\_IO.Currency, tgt\_IO.Converted, tgt\_IO.DESC1, tgt\_IO.PID, tgt\_IO.Portfolio, tgt\_IO.Product, tgt\_IO.[CollateralClass], tgt\_IO.maturityBucket

, tgt\_IO.collateralClass, tgt\_IO.treasuryControl, tgt\_IO.CompanyName, tgt\_IO.reportType, tgt\_IO.forwardStartValue, tgt\_IO.forwardStartBucket

, tgt\_IO.internal, tgt\_IO.internalCounterParty, Tgt\_IO.primeBrokerage, AsofDate, CREATIONDATE

) AS TPOS

LEFT JOIN

(

---SUBQUERY to get Aggregate of SSAH at Security and BR levels

SELECT

tgt\_IO.SECID

,tgt\_IO.BR

,tgt\_IO.reportScope AS [ReportScope]

,tgt\_IO.CompanyName AS [CompanyName]

,tgt\_IO.Currency AS [Currency]

,tgt\_IO.Converted AS [Converted]

,tgt\_IO.DESC1 AS [DESC1]

,SUBSTRING(tgt\_IO.PID, 5, 100) AS [PID]

,tgt\_IO.Portfolio AS [Portfolio]

,tgt\_IO.Product AS [Product]

,SUM(CAST(tgt\_IO.maturityValue AS DECIMAL (30,15))) AS [maturityValue]

,SUM(CAST(tgt\_IO.QTY AS DECIMAL (30,15))) AS Original\_QTY

,CASE WHEN tgt\_IO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_IO.maturityBucket END AS [MaturityBucket]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,0, CHARINDEX('-', tgt\_IO.maturityBucket))

END AS [maturityStart]

,CASE WHEN tgt\_IO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.maturityBucket,CHARINDEX('-', tgt\_IO.maturityBucket)+1,50)

END AS [maturityEnd]

,ABS(SUM(tgt\_IO.forwardStartValue)) AS [ForwardStartValue]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,0, CHARINDEX('-', tgt\_IO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN tgt\_IO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (tgt\_IO.forwardStartBucket,CHARINDEX('-', tgt\_IO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,tgt\_IO.reportType AS [ReportType]

,tgt\_IO.collateralClass AS [CollateralClass]

,tgt\_IO.treasuryControl AS [TreasuryControl]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM #Temp\_A\_IO8 tgt\_IO

WHERE tgt\_IO.PID = 'I.O.8' AND LEFT(tgt\_IO.SRCID, 4) = 'SSAH' --AND tgt\_IO.SECID IN ('US94974BGG95', 'US94974BGS34') --AND LEFT(tgt\_IO\_SSAH.SRCID, 4) = 'SSAH'

GROUP BY

tgt\_IO.SECID, tgt\_IO.BR

, tgt\_IO.reportScope, tgt\_IO.Currency, tgt\_IO.Converted, tgt\_IO.DESC1, tgt\_IO.PID, tgt\_IO.Portfolio, tgt\_IO.Product, tgt\_IO.maturityBucket

, tgt\_IO.collateralClass, tgt\_IO.treasuryControl, tgt\_IO.CompanyName, tgt\_IO.reportType, tgt\_IO.forwardStartValue, tgt\_IO.forwardStartBucket, AsofDate, CREATIONDATE

) AS SSAH

ON

TPOS.reportScope = SSAH.reportScope

AND TPOS.Portfolio = SSAH.Portfolio

AND TPOS.PID = SSAH.PID

AND TPOS.BR = SSAH.BR

AND TPOS.SECID = SSAH.SECID

GROUP BY

TPOS.SECID, TPOS.BR

, TPOS.reportScope, TPOS.Currency, TPOS.Converted, TPOS.DESC1, TPOS.PID, TPOS.Portfolio, TPOS.Product, TPOS.maturityBucket

, TPOS.maturityStart, TPOS.maturityEnd, TPOS.ForwardStart, TPOS.ForwardEnd, TPOS.treasuryControl, TPOS.CompanyName, TPOS.reportType, TPOS.collateralClass, TPOS.Internal

, TPOS.internalCounterParty, TPOS.primeBrokage, TPOS.AsOfDate, TPOS.CREATIONDATE

) tgt\_IO\_outer

GROUP BY

tgt\_IO\_outer.reportScope

,tgt\_IO\_outer.CompanyName

,tgt\_IO\_outer.Currency

,tgt\_IO\_outer.Converted

,tgt\_IO\_outer.DESC1

,tgt\_IO\_outer.PID

,tgt\_IO\_outer.Portfolio

,tgt\_IO\_outer.product

--,tgt\_IO.[SID]

--,tgt\_IO.SubProduct

,tgt\_IO\_outer.maturityBucket

,tgt\_IO\_outer.maturityStart

,tgt\_IO\_outer.maturityEnd

,tgt\_IO\_outer.ForwardStartValue

,tgt\_IO\_outer.ForwardStart

,tgt\_IO\_outer.ForwardEnd

,tgt\_IO\_outer.reportType

,tgt\_IO\_outer.collateralClass

,tgt\_IO\_outer.treasuryControl

,tgt\_IO\_outer.Internal

,tgt\_IO\_outer.InternalCounterParty

,tgt\_IO\_outer.PrimeBrokage

,AsofDate

,CreationDate

-- INSERT INTO [dbo].[AGG\_FR2052a]

--(

-- [ReportScope]

-- ,[Currency]

-- ,[Converted]

-- ,[PID]

-- ,[Product]

-- ,[Maturityvalue]

-- ,[MaturityBucket]

-- ,[maturityStart]

-- ,[maturityEnd]

-- ,[ForwardStartValue]

-- ,[ForwardStartBucket]

-- ,collateralClass

-- ,collateralValue

-- ,CollateralCurrency

-- ,treasuryControl

-- ,[Internal]

-- ,[internalCounterParty]

-- ,[primeBrokerage]

-- ,[ReportType]

--)

-- SELECT

-- tgt\_OI.reportScope AS [ReportScope]

-- ,tgt\_OI.Currency AS [Currency]

-- ,tgt\_OI.Converted AS [Converted]

-- ,SUBSTRING(tgt\_OI.PID, 5, 100) AS [PID]

-- ,tgt\_OI.product AS [Product]

-- ,SUM(ABS(tgt\_OI.maturityValue)) AS [MaturityAmount]

-- ,CASE WHEN tgt\_OI.maturityBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.maturityBucket

-- END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE

-- SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket))

-- END AS [maturityStart]

-- ,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_OI.maturityBucket,CHARINDEX('-', Tgt\_OI.maturityBucket)+1,50)

-- END AS [maturityEnd]

-- ,SUM(tgt\_OI.forwardStartValue) AS [ForwardStart]

-- ,CASE WHEN tgt\_OI.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE tgt\_OI.forwardStartBucket

-- END AS [ForwardBucket]

-- ,tgt\_OI.collateralClass AS [CollateralClass]

-- ,SUM(tgt\_OI.collateralValue) AS [CollateralValue]

-- ,tgt\_OI.CollateralCurrency AS [CollateralCurrency]

-- ,tgt\_OI.treasuryControl AS [TreasuryControl]

-- ,tgt\_OI.internal AS [Internal]

-- ,tgt\_OI.internalCounterParty AS [InternalCounterparty]

-- ,tgt\_OI.primeBrokerage AS [PrimeBrokerage]

-- ,Tgt\_OI.reportType AS [ReportType]

-- FROM app.tgt\_otherInflow tgt\_OI

-- WHERE PID = 'I.O.8'

-- and Portfolio = 'Treasury Investments'

-- GROUP BY

-- Tgt\_OI.reportScope

-- ,tgt\_OI.CompanyName

-- ,Tgt\_OI.Currency

-- ,Tgt\_OI.Converted

-- ,Tgt\_OI.PID

-- ,Tgt\_OI.product

-- ,Tgt\_OI.maturityBucket

-- ,reportType

-- ,tgt\_OI.collateralValue

-- ,tgt\_OI.collateralClass

-- ,tgt\_OI.CollateralCurrency

-- ,tgt\_OI.forwardStartValue

-- ,tgt\_OI.forwardStartBucket

-- ,Tgt\_OI.internal

-- ,tgt\_OI.treasuryControl

-- ,Tgt\_OI.internalCounterParty

-- ,Tgt\_OI.primeBrokerage

-- ,Tgt\_OI.Portfolio

-- ,Tgt\_OI.DESC1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - IO9 (Trade Finance), IO6 (Wholesale Funding Transactions)

,IO9 (Commercial and Retail Loans), IO2 NB (Other Payables and margin collaterals)

,IO9 NB (Other Receivable)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,[Portfolio]

,[DESC1]

,AsofDate

,CreationDate

)

SELECT DISTINCT

tgt\_OI.reportScope AS [ReportScope]

,tgt\_OI.CompanyName AS [CompanyName]

,tgt\_OI.Currency AS [Currency]

,tgt\_OI.Converted AS [Converted]

,SUBSTRING(tgt\_OI.PID, 5, 100) AS [PID]

,tgt\_OI.product AS [Product]

,SUM(tgt\_OI.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_OI.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OI.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OI.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OI.maturityBucket,CHARINDEX('-', Tgt\_OI.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(tgt\_OI.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OI.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OI.forwardStartBucket

END AS [ForwardBucket]

,tgt\_OI.collateralClass AS [CollateralClass]

,SUM(tgt\_OI.collateralValue) AS [CollateralValue]

,tgt\_OI.CollateralCurrency AS [CollateralCurrency]

,tgt\_OI.treasuryControl AS [TreasuryControl]

,tgt\_OI.internal AS [Internal]

,tgt\_OI.internalCounterParty AS [InternalCounterparty]

,tgt\_OI.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OI.reportType AS [ReportType]

,Tgt\_OI.Portfolio AS [Portfolio]

,Tgt\_OI.DESC1 AS [DESC1]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otherInflow tgt\_OI

WHERE ((PID = 'I.O.9'

AND Portfolio = 'Trade Finance')

OR

(PID = 'I.O.6'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'I.O.9'

AND Portfolio = 'Commercial and Retail Loans')

OR

(PID = 'I.O.2'

AND Portfolio = 'Other Payables and margin collaterals')

OR

(PID = 'I.O.9'

AND Portfolio = 'Other Receivable')

OR

(PID = 'I.O.6'

AND Portfolio = 'Treasury Investment')

OR

(PID = 'I.O.7'

AND Portfolio = 'Derivatives')

OR

(PID = 'I.O.1'

AND Portfolio = 'Derivatives')

)

GROUP BY

Tgt\_OI.reportScope

,tgt\_OI.CompanyName

,Tgt\_OI.Currency

,Tgt\_OI.Converted

,Tgt\_OI.PID

,Tgt\_OI.product

,Tgt\_OI.maturityBucket

,reportType

,tgt\_OI.collateralValue

,tgt\_OI.collateralClass

,tgt\_OI.CollateralCurrency

,tgt\_OI.forwardStartValue

,tgt\_OI.forwardStartBucket

,Tgt\_OI.internal

,tgt\_OI.treasuryControl

,Tgt\_OI.internalCounterParty

,Tgt\_OI.primeBrokerage

,Tgt\_OI.Portfolio

,Tgt\_OI.DESC1

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: secured Outflows OS1 (Wholesale Funding Transactions)

,OS9 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,TreasuryControl

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,settlementMechanism

,Rehypothecated

,[CID]

,[counterPartySector]

,[ReportType]

,[Portfolio]

,[DESC1]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_SO.reportScope AS [ReportScope]

,Tgt\_SO.CompanyName AS [CompanyName]

,Tgt\_SO.Currency AS [Currency]

,Tgt\_SO.Converted AS [Converted]

,SUBSTRING(tgt\_SO.PID, 5, 100) AS [PID]

,Tgt\_SO.product AS [Product]

,Tgt\_SO.[SID] AS [SID]

,Tgt\_SO.[Subproduct] AS [SubProduct]

,SUM(Tgt\_SO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_SO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_SO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SO.forwardStartBucket,0, CHARINDEX('-', Tgt\_SO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_SO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SO.forwardStartBucket,CHARINDEX('-', Tgt\_SO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_SO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_SO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_SO.maturityBucket,CHARINDEX('-', Tgt\_SO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_SO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_SO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_SO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_SO.collateralClass AS [CollateralClass]

,SUM(Tgt\_SO.collateralValue) AS [CollateralValue]

,Tgt\_SO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_SO.treasuryControl AS [TreasuryControl]

,Tgt\_SO.internal AS [Internal]

,Tgt\_SO.internalCounterParty AS [InternalCounterparty]

,Tgt\_SO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_SO.settlementMechanism AS [Settlement]

,Tgt\_SO.rehypothecated AS [Rehypothecated]

,Tgt\_SO.CID AS [CID]

,Tgt\_SO.counterPartySector AS [CounterParty]

,Tgt\_SO.reportType AS [ReportType]

,Tgt\_SO.Portfolio AS [Portfolio]

,Tgt\_SO.DESC1 AS [DESC1]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_securedOutflow Tgt\_SO

WHERE ((PID = 'O.S.1'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.S.9'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.S.8'

AND Portfolio = 'Treasury Investment')

)

GROUP BY

Tgt\_SO.reportScope

,Tgt\_SO.Currency

,Tgt\_SO.Converted

,Tgt\_SO.PID

,Tgt\_SO.[SID]

,Tgt\_SO.Subproduct

,Tgt\_SO.product

,Tgt\_SO.maturityBucket

,reportType

,Tgt\_SO.collateralClass

,Tgt\_SO.CollateralCurrency

,Tgt\_SO.forwardStartBucket

,Tgt\_SO.internal

,Tgt\_SO.treasuryControl

,Tgt\_SO.internalCounterParty

,Tgt\_SO.primeBrokerage

,Tgt\_SO.settlementMechanism

,Tgt\_SO.rehypothecated

,Tgt\_SO.CID

,Tgt\_SO.counterPartySector

,Tgt\_SO.CompanyName

,Tgt\_SO.Portfolio

,Tgt\_SO.DESC1

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows - OW9,OW11 (Wholesale Funding Transactions)

, OW19 (Trade Finance)

, OW18 (Other payables and margin collateralss)

, OW19 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[ForwardStartValue]

,[ForwardStartBucket]

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_WO.reportScope AS [ReportScope]

,tgt\_WO.CompanyName AS [CompanyName]

,Tgt\_WO.Currency AS [Currency]

,Tgt\_WO.Converted AS [Converted]

,SUBSTRING(tgt\_WO.PID, 5, 100) AS [PID]

,Tgt\_WO.product AS [Product]

,tgt\_WO.DESC1 AS [DESC1]

,tgt\_WO.Portfolio AS [Portfolio]

,tgt\_WO.CID AS [CID]

,tgt\_WO.counterPartySector AS [Counterparty]

,SUM(Tgt\_WO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_WO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.forwardStartBucket,0, CHARINDEX('-', Tgt\_WO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.forwardStartBucket,CHARINDEX('-', Tgt\_WO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.maturityBucket,CHARINDEX('-', Tgt\_WO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_WO.collateralClass AS [CollateralClass]

,SUM(Tgt\_WO.collateralValue ) AS [CollateralValue]

,Tgt\_WO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_WO.forwardStartValue AS [ForwardStart]

,CASE WHEN tgt\_WO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_WO.internal AS [Internal]

,Tgt\_WO.internalCounterParty AS [InternalCounterparty]

,Tgt\_WO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_WO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_wholesaleoutflow tgt\_WO

WHERE ((PID = 'O.W.19'

AND Portfolio = 'Trade Finance'

)

OR

(PID IN ('O.W.9', 'O.W.11')

AND Portfolio = 'Wholesale Funding Transactions'

)

OR

(PID IN ('O.W.18')

AND Portfolio = 'Other payables and margin collaterals')

OR

(PID IN ('O.W.19')

AND Portfolio = 'Wholesale Funding Transactions')

)

GROUP BY

Tgt\_WO.reportScope

,tgt\_WO.CompanyName

,Tgt\_WO.Currency

,Tgt\_WO.Converted

,Tgt\_WO.PID

,Tgt\_WO.product

,tgt\_WO.DESC1

,tgt\_WO.Portfolio

,Tgt\_WO.maturityBucket

,reportType

,Tgt\_WO.collateralValue

,Tgt\_WO.collateralClass

,Tgt\_WO.CollateralCurrency

,Tgt\_WO.forwardStartValue

,Tgt\_WO.forwardStartBucket

,Tgt\_WO.internal

,tgt\_WO.CID

,tgt\_WO.counterPartySector

,Tgt\_WO.internalCounterParty

,Tgt\_WO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows - OW10 & OW16 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[ForwardStart]

,[ForwardEnd]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[ForwardStartValue]

,[ForwardStartBucket]

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_WO.reportScope AS [ReportScope]

,tgt\_WO.CompanyName AS [CompanyName]

,Tgt\_WO.Currency AS [Currency]

,Tgt\_WO.Converted AS [Converted]

,SUBSTRING(tgt\_WO.PID, 5, 100) AS [PID]

,Tgt\_WO.product AS [Product]

,tgt\_WO.DESC1 AS [DESC1]

,tgt\_WO.Portfolio AS [Portfolio]

,tgt\_WO.CID AS [CID]

,tgt\_WO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_WO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_WO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.forwardStartBucket,0, CHARINDEX('-', Tgt\_WO.forwardStartBucket))

END AS [ForwardStart]

,CASE WHEN Tgt\_WO.forwardStartBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.forwardStartBucket,CHARINDEX('-', Tgt\_WO.forwardStartBucket)+1,50)

END AS [ForwardEnd]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_WO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_WO.maturityBucket,CHARINDEX('-', Tgt\_WO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_WO.collateralClass AS [CollateralClass]

,SUM(Tgt\_WO.collateralValue ) AS [CollateralValue]

,Tgt\_WO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_WO.forwardStartValue AS [ForwardStart]

,CASE WHEN tgt\_WO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_WO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_WO.internal AS [Internal]

,Tgt\_WO.internalCounterParty AS [InternalCounterparty]

,Tgt\_WO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_WO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_wholesaleoutflow tgt\_WO

WHERE (PID = 'O.W.10' AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.W.16' AND Portfolio = 'Commercial and Retail deposits')

GROUP BY

Tgt\_WO.reportScope

,tgt\_WO.CompanyName

,Tgt\_WO.Currency

,Tgt\_WO.Converted

,Tgt\_WO.PID

,Tgt\_WO.product

,tgt\_WO.DESC1

,tgt\_WO.Portfolio

,Tgt\_WO.maturityBucket

,reportType

,Tgt\_WO.collateralValue

,Tgt\_WO.collateralClass

,Tgt\_WO.CollateralCurrency

,Tgt\_WO.forwardStartValue

,Tgt\_WO.forwardStartBucket

,Tgt\_WO.internal

,tgt\_WO.CID

,tgt\_WO.counterPartySector

,Tgt\_WO.internalCounterParty

,Tgt\_WO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows-OO2 (Other Payables and margin collaterals), 001 Derivatives Payables

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(Tgt\_OO.maturityValue) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE (PID = 'O.O.2' AND Portfolio = 'Other Payables and margin collaterals')

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows - OO4 (Credit and liquidity Facility)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

, SUM(CASE WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_TOTAL |%' THEN ABS(TGT\_OO.maturityValue)

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN (-1)\* ABS(TGT\_OO.maturityValue)

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_OS |%' THEN ABS(TGT\_OO.maturityValue)

END ) AS [MaturityValue]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE (

(PID = 'O.O.4'

AND Portfolio = 'Credit and Liquidity Facility')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows - OO5 (Credit and liquidity Facility)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

, SUM(CASE WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_AVAIL\_4027 |%' THEN TGT\_OO.maturityValue

WHEN TGT\_OO.SrcID like 'FBNK\_LIMIT\_OS\_4027 |%' THEN ABS( TGT\_OO.maturityValue)

END ) AS [MaturityValue]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE (

(PID = 'O.O.5'

AND Portfolio = 'Credit and Liquidity Facility')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows- OO7 (Trade Finance), OO19 (Wholesale Funding Transactions)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE ((PID = 'O.O.7'

AND Portfolio = 'Trade Finance'

)

OR

(PID = 'O.O.19'

AND Portfolio = 'Wholesale Funding Transactions')

OR

(PID = 'O.O.1'

AND Portfolio = 'Derivatives')

)

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type:Outflows Deposits OO19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE (PID = 'O.O.19' AND Portfolio = 'Commercial and Retail Deposits')

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflows - OO20 (Derivatives)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.AGG\_BOC

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[DESC1]

,[Portfolio]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,[ForwardStartValue]

,[ForwardStartBucket]

,collateralClass

,collateralValue

,CollateralCurrency

,[Internal]

,[internalCounterParty]

,[primeBrokerage]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT DISTINCT

Tgt\_OO.reportScope AS [ReportScope]

,Tgt\_OO.CompanyName AS [CompanyName]

,Tgt\_OO.Currency AS [Currency]

,Tgt\_OO.Converted AS [Converted]

,SUBSTRING(tgt\_OO.PID, 5, 100) AS [PID]

,Tgt\_OO.product AS [Product]

,tgt\_OO.DESC1 AS [DESC1]

,tgt\_OO.Portfolio AS [Portfolio]

,Tgt\_OO.CID AS [CID]

,Tgt\_OO.counterPartySector AS [Counterparty]

,SUM(ABS(Tgt\_OO.maturityValue)) AS [MaturityAmount]

,CASE WHEN tgt\_OO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.maturityBucket

END AS [MaturityBucket]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_OO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_OO.maturityBucket,CHARINDEX('-', Tgt\_OO.maturityBucket)+1,50)

END AS [maturityEnd]

,SUM(Tgt\_OO.forwardStartValue) AS [ForwardStart]

,CASE WHEN tgt\_OO.forwardStartBucket = 'P1827D-'

THEN 'P1827D'

ELSE tgt\_OO.forwardStartBucket

END AS [ForwardBucket]

,Tgt\_OO.collateralClass AS [CollateralClass]

,SUM(Tgt\_OO.collateralValue) AS [CollateralValue]

,Tgt\_OO.CollateralCurrency AS [CollateralCurrency]

,Tgt\_OO.internal AS [Internal]

,Tgt\_OO.internalCounterParty AS [InternalCounterparty]

,Tgt\_OO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_OO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM app.tgt\_otheroutflow Tgt\_OO

WHERE PID = 'O.O.20' AND Portfolio = 'Derivatives'

GROUP BY

Tgt\_OO.reportScope

,Tgt\_OO.CompanyName

,Tgt\_OO.Currency

,Tgt\_OO.Converted

,Tgt\_OO.PID

,Tgt\_OO.product

,tgt\_OO.DESC1

,tgt\_OO.Portfolio

,Tgt\_OO.maturityBucket

,reportType

,Tgt\_OO.collateralClass

,Tgt\_OO.CollateralCurrency

,Tgt\_OO.forwardStartValue

,Tgt\_OO.forwardStartBucket

,Tgt\_OO.internal

,Tgt\_OO.CID

,Tgt\_OO.counterPartySector

,Tgt\_OO.internalCounterParty

,Tgt\_OO.primeBrokerage

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Operational Deposits

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[DESC1]

,[PID]

,[Portfolio]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[Insured]

,[Trigger]

,[Rehypothecated]

,[Internal]

,[internalCounterParty]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT

Tgt\_DO.reportScope AS [ReportScope]

,Tgt\_DO.CompanyName AS [CompanyName]

,Tgt\_DO.Currency AS [Currency]

,Tgt\_DO.Converted AS [Converted]

,Tgt\_DO.DESC1 AS [DESC1]

,SUBSTRING(Tgt\_DO.PID, 5, 100) AS [PID]

,Tgt\_DO.Portfolio AS [Portfolio]

,Tgt\_DO.product AS [Product]

,Tgt\_DO.CID AS [CID]

,Tgt\_DO.counterPartySector AS [Counterparty]

,SUM(Tgt\_DO.maturityValue) AS [MaturityAmount]

,CASE WHEN Tgt\_DO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE Tgt\_DO.maturityBucket

END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

-- THEN 'P0D'

--ELSE

-- SUBSTRING (Tgt\_DO.forwardStartBucket,0, CHARINDEX('-', Tgt\_DO.forwardStartBucket))

--END AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

--THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_DO.forwardStartBucket,CHARINDEX('-', Tgt\_DO.forwardStartBucket)+1,50)

--END AS [ForwardEnd]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_DO.maturityBucket,0, CHARINDEX('-', Tgt\_DO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_DO.maturityBucket,CHARINDEX('-', Tgt\_DO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_DO.collateralClass AS [CollateralClass]

,sum(Tgt\_DO.collateralValue) AS [CollateralValue]

,Tgt\_DO.CollateralCurrency AS [CollateralCurrency]

,tgt\_do.insured AS [Insured]

,Tgt\_DO.[Trigger] AS [Trigger]

,Tgt\_do.Rehypothecated AS [Rehypothecated]

--,Tgt\_DO.forwardStartValue AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_DO.forwardStartBucket

-- END AS [ForwardBucket]

,Tgt\_DO.internal AS [Internal]

,Tgt\_DO.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_DO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_DO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

--SELECT \*

FROM app.tgt\_depositsOutflow Tgt\_DO

WHERE PID NOT IN ('O.D.11')

GROUP BY

Tgt\_DO.reportScope

,Tgt\_DO.CompanyName

,Tgt\_DO.Currency

,Tgt\_DO.Converted

,Tgt\_DO.PID

,Tgt\_DO.Portfolio

,Tgt\_DO.product

,Tgt\_DO.maturityBucket

,reportType

,Tgt\_DO.collateralValue

,Tgt\_DO.collateralClass

,Tgt\_DO.CollateralCurrency

-- ,Tgt\_DO.forwardStartValue

-- ,Tgt\_DO.forwardStartBucket

,Tgt\_do.Insured

,Tgt\_DO.[Trigger]

,Tgt\_do.Rehypothecated

,Tgt\_DO.internal

,Tgt\_DO.CID

,Tgt\_DO.counterPartySector

,Tgt\_DO.internalCounterParty

--,Tgt\_DO.primeBrokerage

,Tgt\_DO.DESC1

,AsofDate

,CreationDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Operational Deposits - OD11

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,[CompanyName]

,[Currency]

,[Converted]

,[DESC1]

,[PID]

,[Portfolio]

,[Product]

,[CID]

,[counterPartySector]

,[Maturityvalue]

,[MaturityBucket]

,[maturityStart]

,[maturityEnd]

,collateralClass

,collateralValue

,CollateralCurrency

,[Insured]

,[Trigger]

,[Rehypothecated]

,[Internal]

,[internalCounterParty]

,[ReportType]

,AsofDate

,CreationDate

)

SELECT

Tgt\_DO.reportScope AS [ReportScope]

,Tgt\_DO.CompanyName AS [CompanyName]

,Tgt\_DO.Currency AS [Currency]

,Tgt\_DO.Converted AS [Converted]

,Tgt\_DO.DESC1 AS [DESC1]

,SUBSTRING(Tgt\_DO.PID, 5, 100) AS [PID]

,Tgt\_DO.Portfolio AS [Portfolio]

,Tgt\_DO.product AS [Product]

,Tgt\_DO.CID AS [CID]

,Tgt\_DO.counterPartySector AS [Counterparty]

,SUM(Tgt\_DO.maturityValue) AS [MaturityAmount]

,CASE WHEN Tgt\_DO.maturityBucket = 'P1827D-'

THEN 'P1827D'

ELSE Tgt\_DO.maturityBucket

END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

-- THEN 'P0D'

--ELSE

-- SUBSTRING (Tgt\_DO.forwardStartBucket,0, CHARINDEX('-', Tgt\_DO.forwardStartBucket))

--END AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'Open'

--THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_DO.forwardStartBucket,CHARINDEX('-', Tgt\_DO.forwardStartBucket)+1,50)

--END AS [ForwardEnd]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE

SUBSTRING (Tgt\_DO.maturityBucket,0, CHARINDEX('-', Tgt\_DO.maturityBucket))

END AS [maturityStart]

,CASE WHEN Tgt\_DO.maturityBucket = 'Open'

THEN 'P0D'

ELSE SUBSTRING (Tgt\_DO.maturityBucket,CHARINDEX('-', Tgt\_DO.maturityBucket)+1,50)

END AS [maturityEnd]

,Tgt\_DO.collateralClass AS [CollateralClass]

,sum(Tgt\_DO.collateralValue) AS [CollateralValue]

,Tgt\_DO.CollateralCurrency AS [CollateralCurrency]

,tgt\_do.insured AS [Insured]

,Tgt\_DO.[Trigger] AS [Trigger]

,Tgt\_do.Rehypothecated AS [Rehypothecated]

--,Tgt\_DO.forwardStartValue AS [ForwardStart]

--,CASE WHEN Tgt\_DO.forwardStartBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_DO.forwardStartBucket

-- END AS [ForwardBucket]

,Tgt\_DO.internal AS [Internal]

,Tgt\_DO.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_DO.primeBrokerage AS [PrimeBrokerage]

,Tgt\_DO.reportType AS [ReportType]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

--SELECT \*

FROM app.tgt\_depositsOutflow Tgt\_DO

WHERE PID IN ('O.D.11')

GROUP BY

Tgt\_DO.reportScope

,Tgt\_DO.CompanyName

,Tgt\_DO.Currency

,Tgt\_DO.Converted

,Tgt\_DO.PID

,Tgt\_DO.Portfolio

,Tgt\_DO.product

,Tgt\_DO.maturityBucket

,reportType

,Tgt\_DO.collateralValue

,Tgt\_DO.collateralClass

,Tgt\_DO.CollateralCurrency

-- ,Tgt\_DO.forwardStartValue

-- ,Tgt\_DO.forwardStartBucket

,Tgt\_do.Insured

,Tgt\_DO.[Trigger]

,Tgt\_do.Rehypothecated

,Tgt\_DO.internal

,Tgt\_DO.CID

,Tgt\_DO.counterPartySector

,Tgt\_DO.internalCounterParty

--,Tgt\_DO.primeBrokerage

,Tgt\_DO.DESC1

,AsofDate

,CreationDate

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Foreign Exchange

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT INTO [APP].[AGG\_BOC]

-- (

-- [ReportScope]

-- ,[CompanyName]

-- ,[Currency]

-- ,[Converted]

-- ,[PID]

-- ,[Product]

-- ,[DESC1]

-- --,[Portfolio]

-- ,[Maturityvalue]

-- ,[MaturityBucket]

-- ,[maturityStart]

-- ,[maturityEnd]

-- ,[Internal]

-- ,[internalCounterParty]

-- ,[ReportType]

-- )

-- SELECT

-- Tgt\_FX.reportScope AS [ReportScope]

-- ,Tgt\_FX.CompanyName AS [CompanyName]

-- ,Tgt\_FX.Currency\_1 AS [Currency]

-- ,Tgt\_FX.Currency\_1\_Converted AS [Converted]

-- ,SUBSTRING(Tgt\_FX.PID, 6, 100) AS [PID]

-- ,Tgt\_FX.product AS [Product]

-- ,NULL AS [DESC1]

-- -- AS [Portfolio]

-- ,SUM(ABS(Tgt\_FX.[MaturityAmountCurrency1])) AS [MaturityAmount]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'P1827D-'

-- THEN 'P1827D'

-- ELSE Tgt\_FX.maturityBucket

-- END AS [MaturityBucket]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE

-- SUBSTRING (Tgt\_FX.maturityBucket,0, CHARINDEX('-', Tgt\_FX.maturityBucket))

-- END AS [maturityStart]

-- ,CASE WHEN Tgt\_FX.maturityBucket = 'Open'

-- THEN 'P0D'

-- ELSE SUBSTRING (Tgt\_FX.maturityBucket,CHARINDEX('-', Tgt\_FX.maturityBucket)+1,50)

-- END AS [maturityEnd]

-- ,Tgt\_FX.internal AS [Internal]

-- ,Tgt\_FX.internalCounterParty AS [InternalCounterparty]

-- ,Tgt\_FX.reportType AS [ReportType]

-- FROM app.tgt\_ForeignExchange Tgt\_FX

-- GROUP BY

-- Tgt\_FX.reportScope

-- ,tgt\_FX.CompanyName

-- ,Tgt\_FX.Currency\_1

-- ,Tgt\_FX.Currency\_1\_Converted

-- ,Tgt\_FX.PID

-- ,Tgt\_FX.product

-- ,Tgt\_FX.maturityBucket

-- ,reportType

-- ,Tgt\_FX.internal

-- ,Tgt\_FX.internalCounterParty

-- ,Tgt\_FX.reportType

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Supplemental Informational

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[AGG\_BOC]

(

[ReportScope]

,CompanyName

,desc1

,[Currency]

,[Converted]

,[PID]

,[Product]

,[SID]

,[SubProduct]

,[sid2]

,[Subproduct2]

,[MarketValue]

,[CollateralClass]

,[Internal]

,[internalCounterParty]

,[ReportType]

,[primeBrokerage]

--,[CompanyName]

,[Portfolio]

,AsofDate

,CreationDate

)

SELECT

Tgt\_IS.reportScope AS [ReportScope]

,TGT\_IS.COMPANYNAME AS [CompanyName]

,desc1

,Tgt\_IS.Currency AS [Currency]

,Tgt\_IS.Converted AS [Converted]

,SUBSTRING(Tgt\_IS.PID, 5, 100) AS [PID]

,Tgt\_IS.product AS [Product]

,Tgt\_IS.[SID] AS [SID]

,Tgt\_IS.[subProduct] AS [SubProduct]

,Tgt\_IS.SID2 AS [SID2]

,Tgt\_IS.[subProduct2] AS [SubProduct2]

,SUM(Tgt\_IS.marketValue) AS [MarketValue]

,Tgt\_IS.collateralClass AS [CollateralClass]

,Tgt\_IS.internal AS [Internal]

,Tgt\_IS.internalCounterParty AS [InternalCounterParty]

,Tgt\_IS.reportType AS [ReportType]

,Tgt\_IS.primeBrokerage AS [PrimeBrokerage]

,Tgt\_IS.Portfolio AS [portfolio]

,AsofDate AS [AsOfDate]

,creationDate AS [CreationDate]

FROM

app.tgt\_informationalSupplemental Tgt\_IS

WHERE PID IN ('S.I.2','S.I.5','S.I.15','S.I.16','S.I.17','S.I.18')

GROUP BY

Tgt\_IS.reportScope

,Tgt\_IS.Currency

,Tgt\_IS.Converted

,Tgt\_IS.PID

,Tgt\_IS.product

,Tgt\_IS.[SID]

,Tgt\_IS.[subProduct]

,Tgt\_IS.SID2

,Tgt\_IS.[subProduct2]

,Tgt\_IS.collateralClass

,Tgt\_IS.internal

,Tgt\_IS.internalCounterParty

,Tgt\_IS.reportType

,Tgt\_IS.primeBrokerage

,Tgt\_IS.Portfolio

,TGT\_IS.COMPANYNAME

,desc1

,AsofDate

,CreationDate

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

--If user parameter is entered as Null, throw a customized error

EXEC [dbo].[Proc\_InsertIssueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_FR\_Validation] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_FR\_Validation]

--@ASOFDATE DATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Validate data from app.AGG\_FR2052a table against FR2052a Validation rules

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_FR\_Validation]

\*\* Report Type: All Inflows and Outflows

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

--#################################################################################################

-- Validation 1: weekend maturity bucket

-- FRB Validation Logic: Check results in warning if [MaturityBucket] <= "Day 60" and "Day T" +

-- [Maturity Bucket] = "Saturday" or "Sunday" and [MaturityAmount] <= 10000000 in USD equivalent

-- Check fails if [Maturity Bucket] <= "Day 60" and "Day T" + [Maturity Bucket] = "Saturday" or

-- " Sunday" AND some [MaturityAmount] > 10000000

--################################################################################################

-- If the FR2052a Validation returns an Error, then ReportType and PID is used in the dbo.FRValidationTracer to trace the error and validation number

-- Insert results of Validation into #temp\_validation

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT

1 AS FRValidation

, CASE WHEN maturityValue <= 10000000 THEN 'Warning' ELSE 'Fail' END AS ResultType

, CASE WHEN maturityValue <= 10000000 THEN 'Check results in warning if [MaturityBucket] <= "Day 60" and "Day T" + [Maturity Bucket] = "Saturday" or "Sunday" and [MaturityAmount] <= 10000000 in USD equivalent'

ELSE 'Check fails if [Maturity Bucket] <= "Day 60" and "Day T" + [Maturity Bucket] = "Saturday" or " Sunday" AND some [MaturityAmount] > 10000000' END AS Logic

, Agg.ReportType AS ReportType

,PID As PID

,SUSER\_SNAME() AS UserName

,Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS asofdate

--, CAST((CAST(SUM(CASE WHEN AA.Result = 'Fail' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 AS VARCHAR(100)) + '% of validation test aggregate records fail test' [Description]

,@CreationDate AS creationDate

, CASE WHEN ISNULL(MaturityBucket, '') <> '' THEN (

'MaturityBucket = ' + CAST(agg.MaturityBucket AS VARCHAR(20))

+ ' - Maturity Value = ' + CAST(agg.MaturityValue AS VARCHAR(1000))

+ ' Maturity Date = ' + CAST(DATEADD(d, CAST(SUBSTRING(maturityBucket, 2, CHARINDEX('D',maturityBucket,1)-2) AS INT), @ASOFDATE) AS VARCHAR(20))) ELSE '' END --AS One

+ CASE WHEN ISNULL(ForwardStartBucket, '') <> '' THEN (

' \*\*\*\* ForwardStartBucket = ' + CAST(agg.ForwardStartBucket AS VARCHAR(20))

+ ' - ForwardStart Value = ' + CAST(agg.ForwardStartValue AS VARCHAR(1000))

+ ' Forward Date = ' + CAST(DATEADD(d, CAST(SUBSTRING(ForwardStartBucket, 2, CHARINDEX('D',ForwardStartBucket,1)-2) AS INT), @ASOFDATE) AS VARCHAR(20))) ELSE '' END --AS Two

+ CASE WHEN ISNULL(EffectiveMaturityBucket, '') <> '' THEN (

' \*\*\*\* EffectiveBucket = ' + CAST(agg.EffectiveMaturityBucket AS VARCHAR(20))

+ ' - Effective Value = ' --+ CAST(agg.MaturityValue AS VARCHAR(1000))

+ ' Effective Date = ' + CAST(DATEADD(d, CAST(SUBSTRING(EffectiveMaturityBucket, 2, CHARINDEX('D',EffectiveMaturityBucket,1)-2) AS INT), @ASOFDATE) AS VARCHAR(20))) ELSE '' END

AS [Description]

,Agg.AGG\_FR2052aID AS AggFR2052aID

FROM app.AGG\_FR2052a Agg

WHERE

ReportType IN (

'Inflows-Assets'

,'Inflows-Unsecured'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

)

AND (

(SUBSTRING(maturityBucket, 2, CHARINDEX('D',maturityBucket,1)-2) <= 60

AND DATEPART(weekday, DATEADD(d, CAST(SUBSTRING(maturityBucket, 2, CHARINDEX('D',maturityBucket,1)-2) AS INT), @ASOFDATE)) IN (1, 7))

OR

(SUBSTRING(ForwardStartBucket, 2, CHARINDEX('D',ForwardStartBucket,1)-2) <= 60

AND DATEPART(weekday, DATEADD(d, CAST(SUBSTRING(ForwardStartBucket, 2, CHARINDEX('D',ForwardStartBucket,1)-2) AS INT), @ASOFDATE)) IN (1, 7))

OR

(SUBSTRING(EffectiveMaturityBucket, 2, CHARINDEX('D',EffectiveMaturityBucket,1)-2) <= 60

AND DATEPART(weekday, DATEADD(d, CAST(SUBSTRING(EffectiveMaturityBucket, 2, CHARINDEX('D',EffectiveMaturityBucket,1)-2) AS INT), @ASOFDATE)) IN (1, 7))

)

--#################################################################################################

-- Validation 2: Internal transactions Reported on Consolidated Reporting Entity

-- FRB Validation Logic: Check fails if [Reporting Entity] reflects “Consolidated”

-- group entity AND [Internal] = “Y”

--################################################################################################

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Test is not applicable as BOC is not Consolidated Reporting entity

--INSERT INTO app.FRValidationTracer

--(FRValidationID, ResultType, Logic, AggFR2052aID, [description], ReportType, PID, UserName, HostName, asofdate, creationDate)

-- SELECT DISTINCT

-- 2 AS FRValidation

-- ,'Fail' AS ResultType

-- ,'Check fails if [Reporting Entity] reflects “Consolidated” group entity AND [Internal] = “Y”' AS Logic

-- ,AGG\_FR2052aID AS AggFR2052aID

-- ,'Internal Value - ' + Internal + ' AND Reporting Entity is ' +ReportScope AS [description]

-- ,ReportType

-- ,PID

-- ,SUSER\_SNAME() AS UserName

-- ,Host\_NAME() AS HostName

-- ,@ASOFDATE AS asofdate

-- ,GETDATE() AS creationDate

-- FROM

-- app.AGG\_FR2052a A

-- WHERE

-- ReportType IN (

-- 'Inflows-Assets'

-- ,'Inflows-Unsecured'

-- ,'Inflows-Secured'

-- ,'Inflows-Other'

-- ,'Outflows-Deposits'

-- ,'Outflows-Wholesale'

-- ,'Outflows-Secured'

-- ,'Outflows-Other'

-- )

-- AND Internal = 'True'

-- and A.ReportScope IS A CONSOLIDATED ENTITY

--#################################################################################################

-- Validation 3: Internal Transactions Reported Without Internal Counterparty

-- FRB Validation Logic: Check fails if [Internal] = “Y” AND [Internal Counterparty] is empty

--####################################################################################################

-- If the FR2052a Validation returns an Error, then ReportType and PID is used in the dbo.FRValidationTracer to trace the error and validation number

-- Insert results of Validation into #temp\_validation

INSERT INTO app.FRValidationTracer

(FRValidationID, ResultType, Logic, AggFR2052aID, [description], ReportType, PID, UserName, HostName, asofdate, creationDate)

SELECT DISTINCT

3 AS FRValidation

,'Fail' AS ResultType

,'Check fails if [Internal] = “Y” AND [Internal Counterparty] is empty' AS Logic

,'Internal Value - ' + Internal + ' AND Reporting Entity is ' +InternalCounterParty AS [description]

,AGG\_FR2052aID AS AggFR2052aID

,ReportType

,PID

,SUSER\_SNAME() AS UserName

,Host\_NAME() AS HostName

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

FROM app.AGG\_FR2052a

WHERE

ReportType IN

('Inflows-Assets'

,'Inflows-Unsecured'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

,'Supplemental-Informational')

AND InternalCounterParty IS NULL

AND INTERNAL = 'True'

--####################################################################################################

-- Validation 4

--FRB Validation: Check fails if [Lendable Value] > [Market Value]

--####################################################################################################

INSERT INTO app.FRValidationTracer

(FRValidationID, ResultType, Logic, AggFR2052aID, [description], ReportType, PID, UserName, HostName, asofdate, creationDate)

SELECT DISTINCT

4 AS FRValidation

,'Fail' AS ResultType

,'Check fails if [Lendable Value] > [Market Value]' AS Logic

,'Lendable Value - ' + CAST(LendableValue AS VARCHAR(1000)) + ' AND Market value is ' + CAST(MarketValue AS VARCHAR(1000)) AS [description]

,AGG\_FR2052aID AS AggFR2052aID

,ReportType

,PID

,SUSER\_SNAME() AS UserName

,Host\_NAME() AS HostName

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

FROM

app.AGG\_FR2052a

WHERE

ReportType = 'Inflows-Assets'

AND ABS(LendableValue) > ABS(MarketValue)

/\*###########################################################################################################################

-- Validation 5: 3rd Party Reporting Entity Exposures Versus Consolidated

--FRB Validation Logic : o Step 1: Sum the “value fields” identified above, grouped by the distinct values identified

as the “aggregation fields” for the “Consolidated” [Reporting Entity]

o Step 2: Sum the “value fields” identified above, grouped by the distinct values identified

as the “aggregation fields” for the 1st tier [Reporting Entity] values

(defined uniquely for each FR 2052a submitter) where [Internal] is set to false.

o Step 3: Compare each of the “Consolidated” and “1st tier” value fields where aggregation fields match.

• Check fails if the absolute difference for value fields exceed 0.05

(all currencies are denominated in millions, therefore this represents 50,000 currency units)

• Check fails if a combination of aggregation fields exists for the

“Consolidated” reporting entity, but not the “1st tier” reporting entities

• Check fails if a combination of aggregation fields exists for the “1st tier”

reporting entities, but not the “Consolidated” reporting entity

###########################################################################################################################\*/

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Test is not applicable as BOC is not Consolidated Reporting entity

--/####################################################################################################

-- Validation 6

-- FRB Validation Logic

--o Step 1: Sum the “value fields” identified above, grouped by the distinct values identified as

--the “aggregation fields” for all products classified as Inflows.

--o Step 2: Sum the “value fields” identified above, grouped by the distinct values identified as the

--“aggregation fields” for all products classified as Outflows.

--o Step 3: Match the Inflows [Reporting Entity] values with the Outflows [Internal Counterparty] values

--and vice versa, joining on all other aggregation fields (including themapped product).

-- • Check results in a warning if absolute difference between Inflows and Outflows [Maturity Amount]

--values is greater than 0 for less than or equal to 1% of aggregated records defined above.

-- • Check results in a failure if absolute difference between Inflows and Outflows [Maturity Amount]

--values is greater than 0 for more than 1% of aggregated records defined above.

--Check if there are A) Outflow records but no Inflow record OR B) Inflow records but no Outflow record

--####################################################################################################

--o Step 1: Sum the “value fields” identified above, grouped by the distinct values identified as the “aggregation fields” for all products classified as Inflows.

IF OBJECT\_ID('tempdb..#Temp\_Inflow') IS NOT NULL DROP TABLE #Temp\_Inflow

SELECT SUM(isnull(a.[MaturityValue],0)) AS [MaturityValue], a.ReportScope, A.[InternalCounterParty], A.[Currency], Internal --, ReportType, a.PID, a.CounterPartySector, a.MaturityBucket

INTO #Temp\_Inflow

FROM app.AGG\_FR2052a a

INNER JOIN [src].[US\_FR2052A\_REF\_FR\_ValidationNo\_6] b

ON SUBSTRING(A.ReportType, 1, CHARINDEX('-', A.ReportType) -1) = b.[Type]

WHERE b.[TYPE] = 'inflows'

AND MaturityValue IS NOT NULL

AND Internal = 'True'

AND InternalCounterParty IN ('Bank of China - All Other US operations', 'Bank of China - Combined US Branches and Agencies')

GROUP BY

A.ReportScope

, [InternalCounterParty]

, [Currency]

, Internal

--o Step 2: Sum the “value fields” identified above, grouped by the distinct values identified as the aggregation fields” for all products classified as Outflows.

IF OBJECT\_ID('tempdb..#Temp\_Outflow') IS NOT NULL DROP TABLE #Temp\_Outflow

SELECT SUM(isnull(a.[MaturityValue],0)) AS [MaturityValue], a.ReportScope, A.[InternalCounterParty], A.[Currency], Internal --, ReportType, a.PID, a.CounterPartySector, a.MaturityBucket

INTO #Temp\_Outflow

FROM app.AGG\_FR2052a a

INNER JOIN [src].[US\_FR2052A\_REF\_FR\_ValidationNo\_6] b

ON SUBSTRING(A.ReportType, 1, CHARINDEX('-', A.ReportType) -1) = b.[Type]

WHERE b.[TYPE] = 'Outflows'

AND MaturityValue IS NOT NULL

AND Internal = 'True'

AND InternalCounterParty IN ('Bank of China - All Other US operations', 'Bank of China - Combined US Branches and Agencies')

GROUP BY

A.ReportScope

, [InternalCounterParty]

, [Currency]

, Internal

--o Step 3: Match the Inflows [Reporting Entity] values with the Outflows [Internal Counterparty] values and vice versa, joining on all other aggregation fields (including themapped product).

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 6 AS FRValidationID

, 'Fail' AS ResultType

, 'Fail if absolute difference between Inflows and Outflows [Maturity Amount] values is greater than 0 for more than 1% of aggregated records defined above.' AS Logic

, Null AS ReportType

, Null AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, CAST((CAST(SUM(CASE WHEN AA.Result = 'Fail' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 AS VARCHAR(100)) + '% of validation test aggregate records fail test' [Description]

, Null AS AggFR2052aID

FROM

(

SELECT --I.\*, '######' Filler1,

O.\*, '######' Filler2, ABS(ABS(isnull(i.[MaturityValue],0)) - ABS(isnull(o.[MaturityValue],0))) AS Diff

, (CASE WHEN ABS(ABS(ISNULL(i.[MaturityValue],0)) - ABS(ISNULL(o.[MaturityValue],0))) > 0

THEN 'Fail' ELSE 'pass' END) as [Result]

FROM #Temp\_Inflow I

INNER JOIN #Temp\_Outflow O

ON

I.ReportScope = O.InternalCounterParty

AND I.InternalCounterParty = O.ReportScope

--AND ISNULL(I.InternalCounterParty, 'X') = ISNULL(O.InternalCounterParty, 'X')

AND I.Currency = O.Currency

AND I.Internal = O.Internal

) AS AA

HAVING (CAST(SUM(CASE WHEN AA.Result = 'Fail' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 > 1

--Insert FR Validation message if there are A) Outflow records but no Inflow record OR B) Inflow records but no Outflow record

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT

6 AS FRValidationID

, 'Fail' AS ResultType

, 'Fail if absolute difference between Inflows and Outflows [Maturity Amount] values is greater than 0 for more than 1% of aggregated records defined above.' AS Logic

, Null AS ReportType

, Null AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, CASE

WHEN (SELECT COUNT(1) FROM #Temp\_Inflow) = 0 AND (SELECT COUNT(1) FROM #Temp\_Outflow) > 0 THEN 'There are Outflow records but no Inflow record'

WHEN (SELECT COUNT(1) FROM #Temp\_Inflow) > 0 AND (SELECT COUNT(1) FROM #Temp\_Outflow) = 0 THEN 'There are Inflo records but no Outflow record'

END [Description]

, Null AS AggFR2052aID

/\*####################################################################################################

-- Validation 7

-- FRB Validation Logic:

--Check raises warning if the absolute value of ([Collateral Value] – [Maturity Amount] + .01)/([Maturity Amount] + .01) > 75% for all [Product] values}; where both the [Collateral Value] and [Maturity Amount] have been converted to USD, and where [Collateral Value] – [Maturity Amount] > 1.

--Check results in failure if more than 0.5% of records exceed these defined tolerance thresholds.

--####################################################################################################\*/

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 7 AS FRValidationID

, CASE

WHEN (CAST(SUM(CASE WHEN AA.Result = 'Warning' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 > 0.5 THEN 'Fail'

ELSE 'Warning'

END AS ResultType

, 'Fail if more than 0.5% of records exceed defined tolerance thresholds.' AS Logic

, Null AS ReportType

, Null AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, CAST((CAST(SUM(CASE WHEN AA.Result = 'Warning' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 AS VARCHAR(100)) + '% of validation test aggregate records fail test' [Description]

, Null AS AggFR2052aID

FROM

(

SELECT

(CASE --Check 150% threshold for IS5: Margin Loans

WHEN ((ABS(ISNULL(X.[CollateralValue], 0))-ABS(ISNULL(X.[MaturityValue], 0))) + (.01\*1000000))/(ABS(ISNULL(X.[MaturityValue], 0)) + (.01\*1000000)) > 1.5 AND ReportType = 'Inflows-Secured' AND PID = 5

THEN 'Warning'

--Check 75% threshold for other products

WHEN ((ABS(ISNULL(X.[CollateralValue], 0))-ABS(ISNULL(X.[MaturityValue], 0))) + (.01\*1000000))/(ABS(ISNULL(X.[MaturityValue], 0)) + (.01\*1000000)) > 0.75

THEN 'Warning'

ELSE 'Pass'

END) AS [Result]

, ABS(ISNULL(X.[CollateralValue], 0))-ABS(ISNULL(X.[MaturityValue], 0)) Diff

, ((ABS(ISNULL(X.[CollateralValue], 0))-ABS(ISNULL(X.[MaturityValue], 0))) + (.01\*1000000))/(ABS(ISNULL(X.[MaturityValue], 0)) + (.01\*1000000)) Threshold

, \*

FROM app.AGG\_FR2052a X

WHERE

ReportType in ('Inflows-Secured','Outflows-Secured') AND (ABS(ISNULL([CollateralValue], 0)) - ABS(ISNULL([MaturityValue], 0))) > 1

) AS AA

--HAVING (CAST(SUM(CASE WHEN AA.Result = 'Warning' THEN 1 ELSE 0 END) AS FLOAT) / COUNT(\*)) \* 100 > 0.5

/\*####################################################################################################

-- Validation 8

-- FRB Validation Logic:

--Check raises warning if the absolute value of ([Collateral Value] – [Maturity Amount] + .01)/([Maturity Amount] + .01) > 75% for all [Product] values}; where both the [Collateral Value] and [Maturity Amount] have been converted to USD, and where [Collateral Value] – [Maturity Amount] > 1.

--Check results in failure if more than 0.5% of records exceed these defined tolerance thresholds.

--####################################################################################################\*/

--Not relevant as all monteray values are in USD

--####################################################################################################

-- Validation 9

-- FRB Validation Logic:

-- Check fails if the [Product] values reported under a [Reporting Entity] classified as a

-- Consolidated, Stand-alone Parent Company, Lead Bank, Other Bank, Branch or Broker-Dealer

-- Do not include all of the required products associated with that entity type.

--####################################################################################################

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 9 AS FRValidationID

, 'Fail' AS ResultType

, 'Highlights potential errors where products that are generally expected for a particular reporting entity, given its type (e.g., consolidated firm, bank, broker dealer), are not reported.' AS Logic

, Agg.ReportType AS ReportType

, Agg.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, 'To Come' + '% of validation test aggregate records fail test' [Description]

, Agg.AGG\_FR2052aID AS AggFR2052aID

FROM app.AGG\_FR2052a Agg RIGHT JOIN src.US\_FR2052A\_REF\_FR\_ValidationNo\_9 REF

ON AGG.PID = LTRIM(RTRIM(SUBSTRING(REF.Product, CHARINDEX( '.', REF.Product, 3)+1, 10)))

AND (LTRIM(RTRIM(AGG.Product)) = LTRIM(RTRIM(ref.ProductDescription)) OR LTRIM(RTRIM(AGG.Product)) = LTRIM(RTRIM(ref.XML\_ProductDescription)))

WHERE 1=1

AND ReportType IN (

'Inflows-Assets'

,'Inflows-Unsecured'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

,'Supplemental-Informational'

)

AND LTRIM(RTRIM(REF.Type\_Bank)) = 'Branch'

AND AGG.PID IS NULL

----####################################################################################################

-- Validation 10: Improper Intra-entity Consolidation

-- FRB Validation Logic: Check fails if, for transactions with the [Internal] field set to true,

--[Reporting Entity] =[Internal Counterparty].

--####################################################################################################

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 10 AS FRValidationID

, 'Fail' AS ResultType

, 'Verifies that transactions flagged with the [Internal] field set to true do not have the same [Reporting Entity] and [Internal Counterparty] values.' AS Logic

, rpt.ReportType AS ReportType

, rpt.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, 'Reporting Entity (' + Rpt.[ReportScope] + ') <> Internal Counterparty (' + Rpt.[InternalCounterParty] + ')' [Description]

, rpt.AGG\_FR2052aID AS AggFR2052aID

FROM app.AGG\_FR2052a Rpt

WHERE ReportType IN (

'Inflows-Unsecured'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

,'Supplemental-Informational'

)

AND Internal = 'True'

AND ReportScope = InternalCounterParty

--####################################################################################################

-- Validation 11: Duplicate Records

-- FRB Validation Logic: Check fails if two or more FR 2052a records have fully matching non-numeric fields.

--[Reporting Entity] =[Internal Counterparty].

--####################################################################################################

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 11 AS FRValidationID

, 'Fail' AS ResultType

, 'Identifies errors where FR 2052a records submitted do not reflect distinct groupings of non-numeric fields.' AS Logic

, CombinedColumns.ReportType AS ReportType

, CombinedColumns.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, 'AGG IDs: ' + CAST(MAX([AGG\_FR2052aID]) AS VARCHAR(10)) + ' and ' + CAST(MIN([AGG\_FR2052aID]) AS VARCHAR(10)) + ' are duplicates' [Description]

, MAX([AGG\_FR2052aID]) AS AggFR2052aID

FROM (

SELECT

[AGG\_FR2052aID],

reporttype,

pid,

ISNULL(CollateralClass, '')

+ ' || ' + ISNULL(ForwardStartBucket, '')

+ ' || ' + ISNULL(MaturityStart, '')

+ ' || ' + ISNULL(MaturityEnd, '')

+ ' || ' + ISNULL(MaturityBucket, '')

+ ' || ' + ISNULL(Product, '')

+ ' || ' + ISNULL(TreasuryControl, '')

+ ' || ' + ISNULL(CounterPartySector, '')

+ ' || ' + ISNULL(Insured, '')

+ ' || ' + ISNULL(Internal, '')

+ ' || ' + ISNULL(InternalCounterParty, '')

+ ' || ' + ISNULL(Rehypothecated, '')

+ ' || ' + ISNULL([Trigger], '')

+ ' || ' + ISNULL(PrimeBrokerage, '')

+ ' || ' + ISNULL(EffectiveMaturityBucket, '')

+ ' || ' + ISNULL(SettlementMechanism, '')

+ ' || ' + ISNULL(Unencumbered, '')

+ ' || ' + ISNULL(ReportType, '')

+ ' || ' + ISNULL(CollateralCurrency, '')

+ ' || ' + ISNULL(SubProduct, '')

+ ' || ' + ISNULL(SubProduct2, '') AS CombinedFields

FROM app.AGG\_FR2052a

) CombinedColumns

GROUP BY CombinedFields, ReportType, PID

HAVING COUNT(DISTINCT CombinedFields) > 1

/\*####################################################################################################

-- Validation 12

o Check fails if [Counterparty] field is empty where a counterparty value is required

OR [Counterparty] value is not valid per the product designations in Appendix I.

####################################################################################################\*/

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT DISTINCT 12 AS FRValidationID

, 'Fail' AS ResultType

, 'Identifies errors where values reported in the [Counterparty] field are either (1) invalid based on the value in the [Product] field and the list of applicable [Counterparty] values or (2) missing but required based on the value in [Product] field.' AS Logic

, A.ReportType AS ReportType

, A.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, 'Required counterparty is either missing or invalid. CounterPartySector = ' + ISNULL(A.CounterPartySector, '') [Description]

, A.[AGG\_FR2052aID] AS AggFR2052aID

--, 'Filler' Filler1, B.\*, 'Filler' Filler2, A.\*

FROM app.AGG\_FR2052a A INNER JOIN [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_12] B

ON B.PID\_NO = CAST(A.PID AS INT) AND REPLACE(REPLACE( B.[report\_type], CHAR(13), '' ), CHAR(160), '') = A.ReportType

WHERE A.ReportType IN (

'Inflows-Unsecured'

,'Inflows-Secured'

--,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

)

AND ISNULL(A.CounterPartySector, '') NOT IN

(

SELECT REPLACE(REPLACE( [CounterParty\_Values], CHAR(13), '' ), CHAR(160), '')

FROM [REPORT\_FR2052A].[SRC].[US\_FR2052A\_REF\_FR\_ValidationNo\_12] B

WHERE REPLACE(REPLACE( B.[report\_type], CHAR(13), '' ), CHAR(160), '') = A.ReportType

AND B.PID\_NO = CAST(A.PID AS INT)

)

/\*####################################################################################################

-- Validation 13

--Check fails if [Collateral Class] field is empty where a collateral designation is required

--OR [Collateral Class] is populated for a product where it is not applicable per Appendix II.

####################################################################################################\*/

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 13 AS FRValidationID

, 'Fail' AS ResultType

, 'Identifies errors where values reported in the [Collateral Class] field are either (1) missing but required based on the value in [Product] field or (2) reported but not applicable based on the value in [Product] field.' AS Logic

, A.ReportType AS ReportType

, A.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, CASE WHEN b.[Required] = 1 AND A.CollateralClass IS NULL THEN 'Collateral Class is required but not available. CollateralClass = ' + ISNULL(A.CollateralClass, '""')

ELSE 'Collateral Class is available but not required. CollateralClass = ' + ISNULL(A.CollateralClass, '')

END [Description]

, A.[AGG\_FR2052aID] AS AggFR2052aID

FROM [src].[US\_FR2052A\_REF\_FR\_ValidationNo\_13] B FULL OUTER JOIN app.AGG\_FR2052a A

ON SUBSTRING(B.PID, CHARINDEX( '.', B.PID, 3)+1, 10) = A.PID

AND B.ReportType = A.ReportType

WHERE A.ReportType IN (

'Inflows-Assets'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

,'Supplemental-Informational'

)

AND (B.[Required] = 1 AND A.CollateralClass IS NULL)

OR (B.[Not Applica] = 1 AND A.CollateralClass IS NOT NULL)

--####################################################################################################

-- Validation 14

--Sum the [Maturity Amount] field across the aggregation fields

--Warning is issued if calculated amount is > $1 billion USD.

--####################################################################################################

INSERT INTO [app].[FRValidationTracer] ([FRValidationID], [ResultType], [Logic], [ReportType], [PID], [UserName], [HostName], [IssueDate], ASOFDATE, [creationDate], [description], AggFR2052aID)

SELECT 14 AS FRValidationID

, 'Fail' AS ResultType

, 'Identifies instances where balances in excess of $1bn (converted into USD) were reported in one the “other” residual categories of [Product] or [Counterparty].' AS Logic

, A.ReportType AS ReportType

, A.PID AS PID

, SUSER\_SNAME() AS Username

, Host\_NAME() AS HostName

, Null AS IssueDate

,@ASOFDATE AS AsOfDate

,@CreationDate AS creationDate

, 'Balance over $1B for “other” residual categories of [Product] or [Counterparty]. \*\*ReportType = '

+ ISNULL(ReportType, '') + ' \*\*PID = ' + CAST(ISNULL(PID, '') AS VARCHAR(5)) + ' \*\*CounterPartySector = '

+ ISNULL(CounterPartySector, '') + ' \*\*ReportScope = ' + ISNULL(ReportScope, '') + ' \*\*SUM MaturityValue = ' + CAST(ISNULL(MaturityValue, '') AS VARCHAR(100)) AS [Description]

, NULL AS AggFR2052aID

FROM

(

SELECT A.ReportType, A.PID, A.ReportScope, A.CounterPartySector, SUM(maturityValue) MaturityValue

FROM app.AGG\_FR2052a A

WHERE ReportType IN (

'Inflows-Assets'

,'Inflows-Secured'

,'Inflows-Other'

,'Outflows-Deposits'

,'Outflows-Wholesale'

,'Outflows-Secured'

,'Outflows-Other'

,'Supplemental-Informational'

)

AND (

(ReportType = 'Inflows-Other' AND PID IN (9))

OR (ReportType = 'Outflows-Deposits' AND PID IN (12, 13))

OR (ReportType = 'Outflows-Other' AND PID IN (21, 22))

OR (ReportType = 'Outflows-Secured' AND PID IN (9))

OR (ReportType = 'Outflows-Wholesale' AND PID IN (7, 19))

)

OR

(

CounterPartySector = 'Other'

)

GROUP BY ReportType

,[PID]

,ReportScope

,CounterPartySector

HAVING SUM(maturityValue) > 1000000000

) A

END TRY

BEGIN CATCH

EXEC Proc\_InsertIssueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_ForeignExchange] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_ForeignExchange]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_ForeignExchange]

\*\* Report Type: Foreign Exchange

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.Tgt\_ForeignExchange') AND NAME ='IX\_Tgt\_ForeignExchange')

DROP INDEX IX\_Tgt\_ForeignExchange ON APP.Tgt\_ForeignExchange;

CREATE NONCLUSTERED INDEX IX\_Tgt\_ForeignExchange

ON APP.Tgt\_ForeignExchange(Tgt\_ForeignExchange\_ID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

DECLARE @PID VARCHAR(10)

DECLARE @Portfolio VARCHAR(100)

DECLARE @Currency VARCHAR(3)

DECLARE @AsofDate1 DATE

DECLARE @BR VARCHAR(3)

DECLARE @BR1 VARCHAR(3)

DECLARE @BR2 VARCHAR(3)

DECLARE @SPOTFWDIND VARCHAR(1)

DECLARE @ACCTNGTYPE VARCHAR(50)

DECLARE @ACCTNGTYPE1 VARCHAR(50)

DECLARE @ACCTNGTYPE2 VARCHAR(50)

DECLARE @SETTLEMENT VARCHAR(50)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

---- SET VARIABLES

SET @PID = 'S.FX.2'

SET @Portfolio = 'Derivatives'

SET @Currency = 'USD'

SET @BR = 'BR'

SET @ACCTNGTYPE = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND Module = 'YANKEECD'

AND Filter = 'YANKEECD')

SET @SETTLEMENT = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SETTLEMENT')

SET @SPOTFWDIND = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SPOTFWDIND')

IF NOT EXISTS (SELECT \* FROM APP.IssueTracer

WHERE IssueProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail')

INSERT INTO APP.Tgt\_ForeignExchange

(

SRCID

,ReportScope

,Currency\_1

,Currency\_1\_Converted

,Currency\_2

,Currency\_2\_Converted

,PID

,Product

,MaturityAmountCurrency1

,MaturityAmountCurrency2

,MaturityBucket

,ForeignExchangeOptionDirection

,ForwardStartAmountCurrenc1

,ForwardStartAmountCurrenc2

,ForwardStartBucket

,Settlement

,PrimeBrokeage

,Internal

,InternalCounterParty

,CompanyName

,ReportType

,[Portfolio]

,maturityDate

,CreationDate

,AsofDate

,[Event]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

)

SELECT

F\_REVAL.BR+F\_REVAL.DEALNO AS SRCID

,ref\_es.reportingentityDescription AS [ReportScope]

,@Currency AS [Currency\_1]

,CASE WHEN F\_REVAL.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Currency\_1\_Converted]

,@currency AS [Currency\_2]

,CASE WHEN F\_REVAL.CTRCCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Currency\_2\_Converted]

,ref\_rep.PID AS [PID]

,ref\_rep.Product AS [Product]

, CASE WHEN F\_REVAL.CCY = 'USD' THEN F\_REVAL.CCYAMT

ELSE (DBO.fn\_CurrencyExchange

(F\_REVAL.CCY,F\_REVAL.CCYAMT , @AsofDate)) END AS [MaturityAmountCurrency1\_CCYAMT] -- Calcuation performed with Foreign Currency Exchange function

, CASE WHEN F\_REVAL.CTRCCY = 'USD' THEN F\_REVAL.CTRAMT

ELSE (DBO.fn\_CurrencyExchange

(F\_REVAL.CTRCCY,F\_REVAL.CTRAMT, @AsofDate)) END AS [MaturityAmountCurrency2\_CTRAMT] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,CAST(FXDH.VDATE AS DATE)) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CAST(FXDH.VDATE AS DATE), 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,CAST(FXDH.VDATE AS DATE)) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CAST(FXDH.VDATE AS DATE), 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,CAST(FXDH.VDATE AS DATE)) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,CAST(FXDH.VDATE AS DATE), '> 5')

END

END

END AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardExchangeOptionDirection]

,NULL AS [ForwardStartAmountCurrency1]

,NULL AS [ForwardStartAmountCurrency2]

,NULL AS [ForwardStartBucket]

,'Triparty' AS [Settlement]

,NULL AS [PrimeBrokerage]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterparty]

,ref\_es.BranchName AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@Portfolio AS [Portfolio]

,FXDH.VDATE AS maturityDate

,@CreationDate AS [CreationDate]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN FXDH.VDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN F\_REVAL.CCY IS NULL OR F\_REVAL.CCY = ''

OR F\_REVAL.CTRCCY IS NULL OR F\_REVAL.CTRCCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN F\_REVAL.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

OR F\_REVAL.CTRCCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] --Reasonableness Test

,'N/A' AS reasonableness\_Sector --Reasonableness Test

,CASE WHEN F\_REVAL.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] --Reasonableness Test

,CASE WHEN F\_REVAL.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo --Reasonableness Test

,CUST.CNO AS customerNumber --Reasonableness Test

,NULL AS orig\_marketValue --Reasonableness Test

,F\_REVAL.CCYAMT AS orig\_maturityValue --Reasonableness Test

,F\_REVAL.CCY AS orig\_currency --Reasonableness Test

FROM

SRC.[US\_OPICS\_FXDH\_REVAL] F\_REVAL

LEFT JOIN SRC.[US\_OPICS\_FXDH] FXDH

ON FXDH.DEALNO = F\_REVAL.DEALNO

AND FXDH.BR = F\_REVAL.BR

AND FXDH.ASOFDATE = F\_REVAL.ASOFDATE

LEFT JOIN SRC.[US\_OPICS\_CUST] CUST

ON CUST.CNO = FXDH.CUST

AND CUST.ASOFDATE = FXDH.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_es

ON ref\_es.BranchID = F\_REVAL.BR

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = Cust.CNO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_es.reportingentityDescription

AND (FXDH.REVREASON ='' OR LEN(FXDH.REVREASON) = 0)

AND FXDH.VDATE > @AsofDate -- Value date (settlement date) > previous branch processing date vDATE = Settlement date: vDATE > Report Date (e.g. 12/30/2016)

AND FXDH.DEALDATE <= @AsofDate -- Deal date (trade date) <= previous branch processing date DealDATE = Deal agreement date: DealDATE > Report Date (e.g. 12/30/2016)

AND F\_REVAL.ASOFDATE = @AsofDate

AND FXDH.BR IN (SELECT DISTINCT FILTER -- Filter by Multi\_LD.category = 21074

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR)

AND FXDH.SPOTFWDIND = @SPOTFWDIND --SPOTFWDIND = F

AND CUST.ACCTNGTYPE NOT IN (SELECT DISTINCT FILTER --ACCTNGTYPE not = YANKEECD

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND Module = @ACCTNGTYPE

AND Filter = @ACCTNGTYPE)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle errors

EXEC [dbo].[Proc\_InsertIssueDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_RequiredFieldsCheck] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_RequiredFieldsCheck]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Checks if required fields are blank/Null for each target table

\*\* Author: Deloitte

\*\* Create Date: 05/30/2017

\*\* Name: Exec Procedure [dbo].[sp\_RequiredFieldsCheck]

\*\* Report Type: All Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

IF OBJECT\_ID ('tempdb..#TEMP\_CheckRequiredFields') IS NOT NULL

DROP TABLE #TEMP\_CheckRequiredFields

CREATE TABLE #TEMP\_CheckRequiredFields

(CheckRequiredFields\_ID INT Identity(1,1) PRIMARY KEY

,Target\_Table\_Name VARCHAR(100)

,PID VARCHAR(100)

,ReportingEntity VARCHAR(100)

,Currency VARCHAR(100)

,Converted VARCHAR(100)

,PID1 VARCHAR(100)

,Product VARCHAR(100)

,MarketValue VARCHAR(100)

,LendableValue VARCHAR(100)

,MaturityValue VARCHAR(100)

,MaturityBucket VARCHAR(100)

,CollateralClass VARCHAR(100)

,CollateralValue VARCHAR(100)

,CollateralCurrency VARCHAR(100)

,Insured VARCHAR(100)

,[Trigger] VARCHAR(100)

,Unencumbered VARCHAR(100)

,TreasuryControl VARCHAR(100)

,Internal VARCHAR(100)

,PrimeBrokerage VARCHAR(100)

,Settlement VARCHAR(100)

,CID VARCHAR(100)

,CounterParty VARCHAR(100)

,rehypothecated VARCHAR(100)

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------------------------------- INFLOWS -----------------------------------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID ('tempdb..#TEMP\_RequiredFields') IS NOT NULL

DROP TABLE #TEMP\_RequiredFields

SELECT \*

INTO #TEMP\_RequiredFields

FROM (

SELECT

DISTINCT

'Tgt\_assetInflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN AI.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN AI.currency IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS currency

,CASE WHEN AI.converted IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS converted

,CASE WHEN AI.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN AI.product IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS product

,CASE WHEN AI.marketValue IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS marketValue

,CASE WHEN AI.lendableValue IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS lendableValue

,NULL AS MaturityValue

,CASE WHEN AI.maturityBucket IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS maturityBucket

,CASE WHEN AI.collateralClass IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,CASE WHEN AI.treasuryControl IS NULL AND LEFT(AI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS treasuryControl

,NULL as internal

,NULL AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_assetInflow AI

UNION

SELECT

DISTINCT

'Tgt\_unsecuredInflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN UI.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN UI.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN UI.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN UI.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN UI.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN UI.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN UI.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,NULL AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,NULL AS treasuryControl

,CASE WHEN UI.internal IS NULL THEN 1 ELSE 0 END AS internal

,CASE WHEN UI.primeBrokerage IS NULL THEN 1 ELSE 0 END AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_unsecuredInflow UI

UNION

SELECT

DISTINCT

'Tgt\_securedInflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN SI.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN SI.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN SI.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN SI.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN SI.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN SI.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN SI.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,CASE WHEN SI.collateralClass IS NULL THEN 1 ELSE 0 END AS collateralClass

,CASE WHEN SI.collateralValue IS NULL OR CAST(SI.collateralValue AS FLOAT) = 0.00 THEN 1 ELSE 0 END AS collateralValue -- check for zero values and Null values

,CASE WHEN SI.CollateralCurrency IS NULL THEN 1 ELSE 0 END AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,CASE WHEN SI.Unencumbered IS NULL THEN 1 ELSE 0 END AS Unencumbered

,CASE WHEN SI.treasuryControl IS NULL THEN 1 ELSE 0 END AS treasuryControl

,CASE WHEN SI.internal IS NULL THEN 1 ELSE 0 END AS internal

,CASE WHEN SI.primeBrokerage IS NULL THEN 1 ELSE 0 END AS primeBrokerage

,CASE WHEN SI.settlementMechanism IS NULL THEN 1 ELSE 0 END AS Settlement

,CASE WHEN SI.CID IS NULL THEN 1 ELSE 0 END AS CID

,CASE WHEN SI.counterPartySector IS NULL THEN 1 ELSE 0 END AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_securedInflow SI

UNION

SELECT

DISTINCT

'Tgt\_otherInflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN OI.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN OI.currency IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS currency

,CASE WHEN OI.converted IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS converted

,CASE WHEN OI.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN OI.product IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN OI.maturityValue IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN OI.maturityBucket IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS maturityBucket

,NULL AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,CASE WHEN OI.treasuryControl IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS treasuryControl

,CASE WHEN OI.internal IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS internal

,CASE WHEN OI.primeBrokerage IS NULL AND LEFT(OI.SrcID, 4) <> 'SSAH' THEN 1 ELSE 0 END AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_otherInflow OI

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------------------------------- OUTFLOWS -----------------------------------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT

DISTINCT

'Tgt\_depositsOutflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN DO.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN DO.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN DO.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN DO.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN DO.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN DO.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN DO.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,NULL AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,CASE WHEN DO.insured IS NULL THEN 1 ELSE 0 END AS Insured

,CASE WHEN DO.[trigger] IS NULL THEN 1 ELSE 0 END AS [Trigger]

,NULL AS Unencumbered

,NULL AS treasuryControl

,NULL as internal

,NULL AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_depositsOutflow DO

UNION

SELECT

DISTINCT

'Tgt\_wholesaleOutflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN WO.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN WO.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN WO.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN WO.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN WO.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN WO.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN WO.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,NULL AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,NULL AS treasuryControl

,CASE WHEN WO.internal IS NULL THEN 1 ELSE 0 END AS internal

,CASE WHEN WO.primeBrokerage IS NULL THEN 1 ELSE 0 END AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_wholesaleOutflow WO

UNION

SELECT

DISTINCT

'Tgt\_securedOutflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN SO.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN SO.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN SO.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN SO.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN SO.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN SO.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN SO.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,CASE WHEN SO.collateralClass IS NULL THEN 1 ELSE 0 END AS collateralClass

,CASE WHEN SO.collateralValue IS NULL THEN 1 ELSE 0 END AS collateralClass

,CASE WHEN SO.CollateralCurrency IS NULL THEN 1 ELSE 0 END AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,CASE WHEN SO.treasuryControl IS NULL THEN 1 ELSE 0 END AS treasuryControl

,CASE WHEN SO.internal IS NULL THEN 1 ELSE 0 END AS internal

,CASE WHEN SO.primeBrokerage IS NULL THEN 1 ELSE 0 END AS primeBrokerage

,CASE WHEN SO.settlementMechanism IS NULL THEN 1 ELSE 0 END AS Settlement

,CASE WHEN (SO.CID IS NULL AND PID = 'O.S.1') THEN 1

WHEN (PID = 'O.S.8') THEN NULL ELSE 0 END AS CID

,CASE WHEN (SO.counterPartySector IS NULL AND PID = 'O.S.1') THEN 1

WHEN (PID = 'O.S.8') THEN NULL ELSE 0 END AS CounterParty

,CASE WHEN SO.rehypothecated IS NULL THEN 1 ELSE 0 END AS rehypothecated

FROM APP.Tgt\_securedOutflow SO

UNION

SELECT

DISTINCT

'Tgt\_otherOutflow' AS Target\_Table\_Name

,PID

,Portfolio

,CASE WHEN OO.reportScope IS NULL THEN 1 ELSE 0 END AS ReportingEntity

,CASE WHEN OO.currency IS NULL THEN 1 ELSE 0 END AS currency

,CASE WHEN OO.converted IS NULL THEN 1 ELSE 0 END AS converted

,CASE WHEN OO.PID IS NULL THEN 1 ELSE 0 END AS PID1

,CASE WHEN OO.product IS NULL THEN 1 ELSE 0 END AS product

,NULL AS marketValue

,NULL AS lendableValue

,CASE WHEN OO.maturityValue IS NULL THEN 1 ELSE 0 END AS maturityValue

,CASE WHEN OO.maturityBucket IS NULL THEN 1 ELSE 0 END AS maturityBucket

,NULL AS collateralClass

,NULL AS CollateralValue

,NULL AS CollateralCurrency

,NULL AS Insured

,NULL AS [Trigger]

,NULL AS Unencumbered

,NULL AS treasuryControl

,CASE WHEN OO.internal IS NULL THEN 1 ELSE 0 END AS internal

,CASE WHEN OO.primeBrokerage IS NULL THEN 1 ELSE 0 END AS primeBrokerage

,NULL AS Settlement

,NULL AS CID

,NULL AS CounterParty

,NULL AS rehypothecated

FROM APP.Tgt\_otherOutflow OO) X

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------INSERT RESULTS INTO TEMP TABLE----------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO #TEMP\_CheckRequiredFields

(Target\_Table\_Name

,pid

,ReportingEntity

,currency

,converted

,PID1

,product

,marketValue

,lendableValue

,MaturityValue

,maturityBucket

,collateralClass

,CollateralValue

,CollateralCurrency

,Insured

,[Trigger]

,Unencumbered

,treasuryControl

,internal

,primeBrokerage

,Settlement

,CID

,CounterParty

,rehypothecated)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------values greater than zero fail----------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SELECT DISTINCT

Target\_Table\_Name

,PID

,CASE WHEN SUM(ReportingEntity) > 0 THEN 'Fail' ELSE 'Pass' END AS ReportingEntity

,CASE WHEN SUM(currency) > 0 THEN 'Fail' ELSE 'Pass' END AS currency

,CASE WHEN SUM(converted) > 0 THEN 'Fail' ELSE 'Pass' END AS converted

,CASE WHEN SUM(PID1) > 0 THEN 'Fail' ELSE 'Pass' END AS PID1

,CASE WHEN SUM(product) > 0 THEN 'Fail' ELSE 'Pass' END AS product

,CASE WHEN SUM(marketValue) > 0 THEN 'Fail' ELSE 'Pass' END AS marketValue

,CASE WHEN SUM(lendableValue) > 0 THEN 'Fail' ELSE 'Pass' END AS lendableValue

,CASE WHEN SUM(MaturityValue) > 0 THEN 'Fail' ELSE 'Pass' END AS MaturityValue

,CASE WHEN SUM(maturityBucket) > 0 THEN 'Fail' ELSE 'Pass' END AS maturityBucket

,CASE WHEN SUM(collateralClass) > 0 THEN 'Fail' ELSE 'Pass' END AS collateralClass

,CASE WHEN SUM(CollateralValue) > 0 THEN 'Fail' ELSE 'Pass' END AS CollateralValue

,CASE WHEN SUM(CollateralCurrency) > 0 THEN 'Fail' ELSE 'Pass' END AS CollateralCurrency

,CASE WHEN SUM(Insured) > 0 THEN 'Fail' ELSE 'Pass' END AS Insured

,CASE WHEN SUM([Trigger]) > 0 THEN 'Fail' ELSE 'Pass' END AS [Trigger]

,CASE WHEN SUM(Unencumbered) > 0 THEN 'Fail' ELSE 'Pass' END AS Unencumbered

,CASE WHEN SUM(treasuryControl) > 0 THEN 'Fail' ELSE 'Pass' END AS treasuryControl

,CASE WHEN SUM(internal) > 0 THEN 'Fail' ELSE 'Pass' END AS internal

,CASE WHEN SUM(primeBrokerage) > 0 THEN 'Fail' ELSE 'Pass' END AS primeBrokerage

,CASE WHEN SUM(Settlement) > 0 THEN 'Fail' ELSE 'Pass' END AS Settlement

,CASE WHEN SUM(CID) > 0 THEN 'Fail' ELSE 'Pass' END AS CID

,CASE WHEN SUM(CounterParty) > 0 THEN 'Fail' ELSE 'Pass' END AS CounterParty

,CASE WHEN SUM(rehypothecated) > 0 THEN 'Fail' ELSE 'Pass' END AS rehypothecated

FROM #TEMP\_RequiredFields

GROUP BY Target\_Table\_Name, pid

--SELECT \* FROM #TEMP\_RequiredFields

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------Add results to issue Tracer by Field----------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF EXISTS (SELECT DISTINCT ReportingEntity

FROM #TEMP\_CheckRequiredFields A

WHERE ReportingEntity = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid ,'ReportingEntity not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.ReportingEntity = 'Fail'

IF EXISTS (SELECT DISTINCT Currency

FROM #TEMP\_CheckRequiredFields A

WHERE currency = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'Currency not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Currency = 'Fail'

IF EXISTS (SELECT DISTINCT Currency

FROM #TEMP\_CheckRequiredFields A

WHERE Converted = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'Converted not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Converted = 'Fail'

IF EXISTS (SELECT DISTINCT PID

FROM #TEMP\_CheckRequiredFields A

WHERE PID = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'PID not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.PID1 = 'Fail'

IF EXISTS (SELECT DISTINCT Product

FROM #TEMP\_CheckRequiredFields A

WHERE ReportingEntity = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'Product not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Product = 'Fail'

IF EXISTS (SELECT DISTINCT MarketValue

FROM #TEMP\_CheckRequiredFields A

WHERE MarketValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'MarketValue not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.MarketValue = 'Fail'

IF EXISTS (SELECT DISTINCT LendableValue

FROM #TEMP\_CheckRequiredFields A

WHERE ReportingEntity = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'LendableValue not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.LendableValue = 'Fail'

IF EXISTS (SELECT DISTINCT MaturityBucket

FROM #TEMP\_CheckRequiredFields A

WHERE MaturityBucket = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'MaturityBucket not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.MaturityBucket = 'Fail'

IF EXISTS (SELECT DISTINCT CollateralClass

FROM #TEMP\_CheckRequiredFields A

WHERE CollateralClass = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'CollateralClass not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.CollateralClass = 'Fail'

IF EXISTS (SELECT DISTINCT CollateralValue

FROM #TEMP\_CheckRequiredFields A

WHERE CollateralValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'CollateralValue not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.CollateralValue = 'Fail'

IF EXISTS (SELECT DISTINCT CollateralCurrency

FROM #TEMP\_CheckRequiredFields A

WHERE CollateralCurrency = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'CollateralCurrency not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.CollateralCurrency = 'Fail'

IF EXISTS (SELECT DISTINCT Insured

FROM #TEMP\_CheckRequiredFields A

WHERE Insured = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'Insured not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Insured = 'Fail'

IF EXISTS (SELECT DISTINCT [Trigger]

FROM #TEMP\_CheckRequiredFields A

WHERE [Trigger] = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'Trigger not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.[Trigger] = 'Fail'

IF EXISTS (SELECT DISTINCT Unencumbered

FROM #TEMP\_CheckRequiredFields A

WHERE Unencumbered = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'Unencumbered not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Unencumbered = 'Fail'

IF EXISTS (SELECT DISTINCT TreasuryControl

FROM #TEMP\_CheckRequiredFields A

WHERE TreasuryControl = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'TreasuryControl not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.TreasuryControl = 'Fail'

IF EXISTS (SELECT DISTINCT Internal

FROM #TEMP\_CheckRequiredFields A

WHERE Internal = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'Internal not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Internal = 'Fail'

IF EXISTS (SELECT DISTINCT PrimeBrokerage

FROM #TEMP\_CheckRequiredFields A

WHERE PrimeBrokerage = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid ,'PrimeBrokerage not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.PrimeBrokerage = 'Fail'

IF EXISTS (SELECT DISTINCT Settlement

FROM #TEMP\_CheckRequiredFields A

WHERE Settlement = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'Settlement not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.Settlement = 'Fail'

IF EXISTS (SELECT DISTINCT CID

FROM #TEMP\_CheckRequiredFields A

WHERE CID = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'CID not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.CID = 'Fail'

IF EXISTS (SELECT DISTINCT CounterParty

FROM #TEMP\_CheckRequiredFields A

WHERE CounterParty = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid , 'CounterParty not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.CounterParty = 'Fail'

IF EXISTS (SELECT DISTINCT rehypothecated

FROM #TEMP\_CheckRequiredFields A

WHERE rehypothecated = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','[sp\_RequiredFieldsCheck] | ' +a.pid, 'rehypothecated not populated',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #TEMP\_CheckRequiredFields A

where A.rehypothecated = 'Fail'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------Add results of Reasonableness test - Customer ---------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_assetInflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','assetInflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_assetInflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_ForeignExchange

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','ForeignExchange' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_ForeignExchange A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_unsecuredInflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','unsecuredInflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_unsecuredInflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_securedInflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedInflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedInflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_wholesaleOutflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','wholesaleOutflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_wholesaleOutflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_securedOutflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedOutflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedOutflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_otherOutflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','otherOutflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherOutflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_depositsOutflow

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','depositsOutflow' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_depositsOutflow A

where reasonableness\_Customer = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Customer

FROM app.Tgt\_informationalSupplemental

WHERE reasonableness\_Customer = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','informationalSupplemental' +a.pid ,'Reasonableness test failed - Customer missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_informationalSupplemental A

where reasonableness\_Customer = 'N'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------Add results of Reasonableness test - sector ---------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_assetInflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','assetInflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_assetInflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_ForeignExchange

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','ForeignExchange' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_ForeignExchange A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_unsecuredInflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','unsecuredInflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_unsecuredInflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_securedInflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedInflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedInflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_wholesaleOutflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','wholesaleOutflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_wholesaleOutflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_securedOutflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedOutflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedOutflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_otherOutflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','otherOutflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherOutflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_depositsOutflow

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','depositsOutflow' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_depositsOutflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Sector

FROM app.Tgt\_informationalSupplemental

WHERE reasonableness\_Sector = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','informationalSupplemental' +a.pid ,'Reasonableness test failed - Sector missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_informationalSupplemental A

where reasonableness\_Sector = 'N'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------Add results of Reasonableness test - Company ---------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_assetInflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','assetInflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_assetInflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_ForeignExchange

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','ForeignExchange' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_ForeignExchange A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_unsecuredInflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','unsecuredInflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_unsecuredInflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_securedInflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedInflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedInflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_wholesaleOutflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','wholesaleOutflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_wholesaleOutflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_securedOutflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedOutflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedOutflow A

where reasonableness\_Sector = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_otherOutflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','otherOutflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherOutflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_depositsOutflow

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','depositsOutflow' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_depositsOutflow A

where reasonableness\_Company = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Company

FROM app.Tgt\_informationalSupplemental

WHERE reasonableness\_Company = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','informationalSupplemental' +a.pid ,'Reasonableness test failed - Company missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_informationalSupplemental A

where reasonableness\_Company = 'N'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

----------Add results of Reasonableness test - Contract/dealNo ---------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_assetInflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','assetInflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_assetInflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_ForeignExchange

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','ForeignExchange' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_ForeignExchange A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_unsecuredInflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','unsecuredInflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_unsecuredInflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_securedInflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedInflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedInflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_wholesaleOutflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','wholesaleOutflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_wholesaleOutflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_securedOutflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','securedOutflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_securedOutflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_otherOutflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','otherOutflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_otherOutflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_depositsOutflow

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','depositsOutflow' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_depositsOutflow A

where reasonableness\_Contract\_DealNo = 'N'

IF EXISTS (SELECT DISTINCT reasonableness\_Contract\_DealNo

FROM app.Tgt\_informationalSupplemental

WHERE reasonableness\_Contract\_DealNo = 'N')

INSERT INTO APP.issueTracer

SELECT distinct '','','','','informationalSupplemental' +a.pid ,'Reasonableness test failed - Contract/dealNo missing in source',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM app.Tgt\_informationalSupplemental A

where reasonableness\_Contract\_DealNo = 'N'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle issues

EXEC [dbo].[Proc\_InsertIssueDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_other\_outflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_other\_outflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_other\_outflow]

\*\* Report Type: Other Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Entity') AND NAME ='IX\_Ref\_En')

DROP INDEX IX\_Ref\_En ON app.US\_FR2052A\_REF\_Entity;

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate\_1 Date

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR(10)

DECLARE @Settlement VARCHar(20)

DECLARE @CompanyName VARCHAR (50) -- O.O.19 (CHB AND LAB)

DECLARE @CompanyName1 VARCHAR (50) -- O.O.19 (CHB AND LAB)

DECLARE @BR VARCHAR(5)

DECLARE @AL VARCHAR(5)

DECLARE @reportType VARCHAR(50)

DECLARE @ReportEntity VARCHAR(50)

DECLARE @REVREASON VARCHAR(20)

DECLARE @CCYAMT INT

DECLARE @SCHTYPE VARCHAR(20)

DECLARE @INVTYPE VARCHAR(10)

DECLARE @ACCTNGTYPE VARCHAR(10)

DECLARE @SETTAVGCOST INT

DECLARE @PID\_OO1 VARCHAR(10)

DECLARE @PID\_OS8 VARCHAR(10)

DECLARE @AVAILAMT VARCHAR(10)

DECLARE @MAXTOTAL VARCHAR(10)

DECLARE @TOTALOS VARCHAR(10)

DECLARE @COMMITTED VARCHAR(10)

DECLARE @LIMREF VARCHAR(10)

DECLARE @LIMREF1 VARCHAR(10)

DECLARE @LIMREF2 VARCHAR(10)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives

\*\* PID: O.O.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'O.O.1'

SET @Portfolio = 'Derivatives'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter -- TreasuryControl = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @Category = (SELECT DISTINCT Filter -- Category = 21033

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Category'

)

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT DISTINCT

M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN M\_LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

, NULL AS [CID]

, NULL AS [CounterPartySector]

,(DBO.fn\_CurrencyExchange (M\_LD.CURRENCY,(BAL.INTEREST), @ASOFDATE))AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) > 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,ref\_en.Company\_Name AS [CompanyName]

,CASE WHEN M\_LD.CUSTOMER\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN M\_LD.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN M\_LD.ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,BAL.[DATE] AS [maturityDate] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@ASOFDATE AS [AsofDate]

,((SELECT CASE WHEN BAL.[DATE] < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN M\_LD.Currency IS NULL OR M\_LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN M\_LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,M\_LD.CUSTOMER\_ID AS customerNumber

,NULL AS orig\_marketValue

,BAL.INTEREST AS orig\_maturityValue

,M\_LD.CURRENCY AS orig\_currency

,@CreationDate AS CreationDate

FROM SRC.MULTI\_LD M\_LD

LEFT JOIN SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

ON M\_LD.ID = BAL.ID

AND M\_LD.CO\_CODE = BAL.CO\_CODE

AND M\_LD.ASOFDATE = BAL.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = M\_LD.CO\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = M\_LD.CUSTOMER\_ID

AND ref\_en.ReportingEntity = ref\_Internal\_cp.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

ISNULL (ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND M\_LD.CATEGORY = @Category

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

AND BAL.[DATE] > @ASOFDATE

AND M\_LD.ASOFDATE = @ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Credit and Liquidity Facility

\*\* PID: O.O.4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.O.4'

SET @Portfolio = 'Credit and Liquidity Facility'

SET @Currency = 'USD'

SET @AVAILAMT = (SELECT DISTINCT Filter -- AvailAmt = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'availamt')

SET @MAXTOTAL = (SELECT DISTINCT Filter -- MaximumTotal = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'MaximumTotal')

SET @Committed = (SELECT DISTINCT Filter -- Committed\_YN = Y

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Committed')

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O.O.4 - Total avail amount : all limit references \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FBNK\_LIMIT\_AVAIL\_TOTAL | '+LIMIT.[CREDIT\_LINE] AS [SrcID] -- Represents the total available amounts of all limit reefernces

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN LIMIT.LIMIT\_Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

, CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE (dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, @AsOfDate))END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

, dbo.fn\_MaturityDateCalculater(@AsOfDate, @AsOfDate, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A'

THEN NULL ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

-- Collateral Value is calculated as (Avail\_Amt/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A' THEN NULL

ELSE ((CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, @AsOfDate)

END /dbo.fn\_CurrencyExchange

(LIMIT.LIMIT\_currency,LIMIT.MAXIMUM\_TOTAL, @AsOfDate)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (dbo.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsOfDate))

END )))END AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN LEFT(LIMIT.CREDIT\_LINE,7) IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,CASE WHEN CUST.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector]

,CASE WHEN LIMIT.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'NA' AS [reasonableness\_contract\_dealno]

,@AsOfDate AS [MaturityDate]

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT

CASE WHEN LIMIT.LIMIT\_currency IS NULL OR LIMIT.LIMIT\_currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.LIMIT\_currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT]

,LEFT(LIMIT.CREDIT\_LINE,7) AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,LIMIT.AVAIL\_AMT AS orig\_maturityValue -- Reasonableness Test

,LIMIT.LIMIT\_currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.FBNK\_LIMIT LIMIT

-- Collateral right table is the intermediary table to get to collateral table from FBNK\_LIMIT

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = limit.id

AND limit.collateral\_code = RIG.collateral\_code

AND LIMIT.asofdate = RIG.asofdate

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND RIG.asofdate = COL.asofdate

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.ID = LEFT(LIMIT.CREDIT\_LINE,7)

AND CUST.ASOFDATE = LIMIT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = LIMIT.co\_code

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = CUST.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LEFT(LIMIT.CREDIT\_LINE,7)

WHERE

LIMIT.ASOFDATE = @AsOfDate

AND isnull(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

--To eliminate "divide by 0" ISSUE

AND AVAIL\_AMT <> @AVAILAMT

AND AVAIL\_AMT IS NOT NULL

AND (

MAXIMUM\_TOTAL <> @MAXTOTAL

AND MAXIMUM\_TOTAL IS NOT NULL

)

--To identify records at the global level

AND record\_parent IS NULL

AND LIMIT.COMMITTED\_YN = @COMMITTED

GROUP BY

ref\_en.ReportingEntityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,LIMIT.CO\_CODE

,LIMIT.credit\_line

,limit.limit\_currency

,RIG.ID

,LIMIT.AVAIL\_AMT

,TOTAL\_OS

,LIMIT.MAXIMUM\_TOTAL

,RIG.id

,col.collateral\_code

,LIMIT.COMMITTED\_YN

,ref\_en.Company\_name

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O.O.4 - Avail amount for 4027 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @LIMREF = (SELECT DISTINCT Filter -- limit reference = 4027

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'LIMREF')

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FBNK\_LIMIT\_AVAIL\_4027 | '+LIMIT.[CREDIT\_LINE\_NO] AS [SrcID] -- Represents avail amount fot 4027

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN LIMIT.LIMIT\_Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CustomerID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

, CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE (dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, @AsOfDate))END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

, dbo.fn\_MaturityDateCalculater(@AsOfDate, @AsOfDate, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A'

THEN NULL ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

-- Collateral Value is calculated as (Avail\_Amt/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,'0' AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN LEFT(LIMIT.CREDIT\_LINE,7) IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN CUST.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIMIT.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'NA' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,NULL AS [MaturityDate] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT

CASE WHEN LIMIT.LIMIT\_currency IS NULL OR LIMIT.LIMIT\_currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.LIMIT\_currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT]

,LEFT(LIMIT.CREDIT\_LINE,7) AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,LIMIT.AVAIL\_AMT AS orig\_maturityValue -- Reasonableness Test

,LIMIT.LIMIT\_currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.FBNK\_LIMIT LIMIT

-- Collateral right table is the intermediary table to get to collateral table from FBNK\_LIMIT

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = limit.id

AND limit.collateral\_code = RIG.collateral\_code

AND LIMIT.asofdate = RIG.asofdate

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND RIG.asofdate = COL.asofdate

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.ID = LEFT(LIMIT.CREDIT\_LINE,7)

AND CUST.ASOFDATE = LIMIT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = LIMIT.co\_code

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = CUST.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LEFT(LIMIT.CREDIT\_LINE,7)

WHERE

LIMIT.ASOFDATE = @AsOfDate

AND isnull(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

--To eliminate "divide by 0" ISSUE

AND AVAIL\_AMT<> @AVAILAMT

AND AVAIL\_AMT IS NOT NULL

AND (

MAXIMUM\_TOTAL <> @MAXTOTAL

AND MAXIMUM\_TOTAL IS NOT NULL

)

AND RECORD\_PARENT IS NULL

--To identify records at the global level

AND LIMIT.COMMITTED\_YN = @COMMITTED

and LIMIT.CREDIT\_LINE in (select CREDIT\_LINE

from SRC.FBNK\_LIMIT

WHERE LEFT(RIGHT(CREDIT\_LINE\_NO,7),4) = @LIMREF

)

GROUP BY

ref\_en.ReportingEntityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,LIMIT.CO\_CODE

,LIMIT.credit\_line\_NO

,LIMIT.credit\_line

,limit.limit\_currency

,RIG.ID

,LIMIT.AVAIL\_AMT

,TOTAL\_OS

,LIMIT.MAXIMUM\_TOTAL

,RIG.id

,col.collateral\_code

,LIMIT.COMMITTED\_YN

,ref\_en.Company\_name

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O.O.4 - Total Outstanding amount 3025 and 3011 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @LIMREF1 = (SELECT DISTINCT Filter -- Limit reference = 3025

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'LIMREF1')

SET @LIMREF2 = (SELECT DISTINCT Filter -- Limit reference = 3011

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'LIMREF2')

SET @TOTALOS = (SELECT DISTINCT Filter -- Total OS = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TotalOS')

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FBNK\_LIMIT\_OS | '+LIMIT.[CREDIT\_LINE\_NO] AS [SrcID] -- Represents total outstanding amoutns for 3025 and 3011

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN LIMIT.LIMIT\_Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

, CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE (dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.TOTAL\_OS, @AsOfDate))END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

, dbo.fn\_MaturityDateCalculater(@AsOfDate, @AsOfDate, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A'

THEN NULL ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

-- Collateral Value is calculated as (Avail\_Amt/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,'0' AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN LEFT(LIMIT.CREDIT\_LINE,7) IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN CUST.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIMIT.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'NA' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,NULL AS [MaturityDate] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT

CASE WHEN LIMIT.LIMIT\_currency IS NULL OR LIMIT.LIMIT\_currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.LIMIT\_currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT]

,LEFT(LIMIT.CREDIT\_LINE,7) AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,LIMIT.AVAIL\_AMT AS orig\_maturityValue -- Reasonableness Test

,LIMIT.LIMIT\_currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.FBNK\_LIMIT LIMIT

-- Collateral right table is the intermediary table to get to collateral table from FBNK\_LIMIT

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = limit.id

AND limit.collateral\_code = RIG.collateral\_code

AND LIMIT.asofdate = RIG.asofdate

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND RIG.asofdate = COL.asofdate

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.ID = LEFT(LIMIT.CREDIT\_LINE,7)

AND CUST.ASOFDATE = LIMIT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = LIMIT.co\_code

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = CUST.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LEFT(LIMIT.CREDIT\_LINE,7)

WHERE

LIMIT.ASOFDATE = @AsOfDate

AND isnull(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

--To eliminate "divide by 0" ISSUE

AND TOTAL\_OS <> @TOTALOS

AND TOTAL\_OS IS NOT NULL

AND (

MAXIMUM\_TOTAL <> @MAXTOTAL

AND MAXIMUM\_TOTAL IS NOT NULL

)

--To identify records at the global level

AND LIMIT.COMMITTED\_YN = @COMMITTED

and LEFT(RIGHT(CREDIT\_LINE\_NO,7),4) IN (@LIMREF1, @LIMREF2)

GROUP BY

ref\_en.ReportingEntityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,LIMIT.CO\_CODE

,LIMIT.credit\_line\_NO

,LIMIT.credit\_line

,limit.limit\_currency

,RIG.ID

,LIMIT.AVAIL\_AMT

,TOTAL\_OS

,LIMIT.MAXIMUM\_TOTAL

,RIG.id

,col.collateral\_code

,LIMIT.COMMITTED\_YN

,ref\_en.Company\_name

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Credit and Liquidity Facility

\*\* PID: O.O.5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.O.5'

SET @PID1 = 'O.O.4'

SET @Portfolio = 'Credit and Liquidity Facility'

SET @AVAILAMT = (SELECT DISTINCT Filter -- avail amt = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'availAmt')

SET @TOTALOS = (SELECT DISTINCT Filter -- Total OS = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'TotalOS')

SET @MAXTOTAL = (SELECT DISTINCT Filter -- Maximum Total = 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'MaximumTotal')

SET @Currency = 'USD'

set @COMMITTED = (SELECT DISTINCT Filter -- Committed = Y

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'Committed')

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

set @LIMREF = (SELECT DISTINCT Filter -- Limit reference = 4027

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID1

AND Portfolio = @Portfolio

AND FilterName = 'limref')

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O.O.5 - avail amount - 4027 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FBNK\_LIMIT\_AVAIL\_4027 | '+LIMIT.[CREDIT\_LINE\_NO] AS [SrcID] --Represents avail amount for 4027

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN LIMIT.LIMIT\_Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

, CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE (dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, @AsOfDate))END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

, dbo.fn\_MaturityDateCalculater(@AsOfDate, @AsOfDate, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A'

THEN NULL ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

-- Collateral Value is calculated as (Avail\_Amt/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A' THEN NULL

ELSE ((CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.AVAIL\_AMT, @AsOfDate)

END /dbo.fn\_CurrencyExchange

(LIMIT.LIMIT\_currency,LIMIT.MAXIMUM\_TOTAL, @AsOfDate)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (dbo.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsOfDate))

END )))END AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN LEFT(LIMIT.CREDIT\_LINE,7) IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN CUST.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN LIMIT.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'NA' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,NULL AS [MaturityDate]

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT

CASE WHEN LIMIT.LIMIT\_currency IS NULL OR LIMIT.LIMIT\_currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.LIMIT\_currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

) AS [EVENT]

,LEFT(LIMIT.CREDIT\_LINE,7) AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,LIMIT.AVAIL\_AMT AS orig\_maturityValue -- Reasonableness Test

,LIMIT.LIMIT\_currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.FBNK\_LIMIT LIMIT

-- Collateral right table is the intermediary table to get to collateral table from FBNK\_LIMIT

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = limit.id

AND limit.collateral\_code = RIG.collateral\_code

AND LIMIT.asofdate = RIG.asofdate

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND RIG.asofdate = COL.asofdate

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.ID = LEFT(LIMIT.CREDIT\_LINE,7)

AND CUST.ASOFDATE = LIMIT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = LIMIT.co\_code

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = CUST.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LEFT(LIMIT.CREDIT\_LINE,7)

WHERE

LIMIT.ASOFDATE = @AsOfDate

AND isnull(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

--To eliminate "divide by 0" ISSUE

AND AVAIL\_AMT<> @AVAILAMT

AND AVAIL\_AMT IS NOT NULL

AND (

MAXIMUM\_TOTAL <> @MAXTOTAL

AND MAXIMUM\_TOTAL IS NOT NULL

)

AND RECORD\_PARENT IS NULL

--To identify records at the global level

AND LIMIT.COMMITTED\_YN = @COMMITTED

and LIMIT.CREDIT\_LINE in (select CREDIT\_LINE

from SRC.FBNK\_LIMIT

WHERE LEFT(RIGHT(CREDIT\_LINE\_NO,7),4) = @LIMREF)

GROUP BY

ref\_en.ReportingEntityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,LIMIT.CO\_CODE

,LIMIT.credit\_line\_NO

,LIMIT.credit\_line

,limit.limit\_currency

,RIG.ID

,LIMIT.AVAIL\_AMT

,TOTAL\_OS

,LIMIT.MAXIMUM\_TOTAL

,RIG.id

,col.collateral\_code

,LIMIT.COMMITTED\_YN

,ref\_en.Company\_name

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O.O.5 - Total OS - 4027 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,[CID]

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'FBNK\_LIMIT\_OS\_4027 | '+LIMIT.[CREDIT\_LINE\_NO] AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN LIMIT.LIMIT\_Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,ISNULL((SELECT EXC.CID FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.CID) AS [CID] -- CID exception Logic

,ISNULL((SELECT EXC.COunterparty FROM

app.US\_FR2052A\_REF\_CID\_Exception EXC

WHERE EXC.Sector = CUST.SECTOR

AND (( EXC.INDUSTRY = cust.INDUSTRY )

OR

(EXC.CUSTOMERID = CUST.ID)

)),ref\_cp.COunterparty) AS [CounterParty]

, CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE (dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.TOTAL\_OS, @AsOfDate))END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

, dbo.fn\_MaturityDateCalculater(@AsOfDate, @AsOfDate, 'Open')

AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A'

THEN NULL ELSE ref\_col.CollateralClassFR2052A END AS [CollateralClass]

-- Collateral Value is calculated as (Avail\_Amt/Maximum\_Total)\*Nominal\_Value

--To assign collateral proportionately to the available amount

,CASE WHEN ref\_col.CollateralClassFR2052A = 'N/A' THEN NULL

ELSE ((CASE WHEN LIMIT.AVAIL\_AMT IS NULL THEN '0'

ELSE dbo.fn\_CurrencyExchange(LIMIT.LIMIT\_currency,LIMIT.TOTAL\_OS, @AsOfDate)

END /dbo.fn\_CurrencyExchange

(LIMIT.LIMIT\_currency,LIMIT.MAXIMUM\_TOTAL, @AsOfDate)\*

SUM(CASE WHEN COL.NOMINAL\_VALUE IS NULL THEN 0

ELSE (dbo.fn\_CurrencyExchange

(COL.currency,(COL.NOMINAL\_VALUE), @AsOfDate))

END )))END AS [CollateralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,CASE WHEN LEFT(LIMIT.CREDIT\_LINE,7) IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]-- Reasonableness Test

,CASE WHEN CUST.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector]-- Reasonableness Test

,CASE WHEN LIMIT.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]-- Reasonableness Test

,'NA' AS [reasonableness\_contract\_dealno]

,NULL AS [MaturityDate]

,ref\_rep.ReportType AS [ReportType]

,@AsOfDate AS [AsofDate]

,((SELECT

CASE WHEN LIMIT.LIMIT\_currency IS NULL OR LIMIT.LIMIT\_currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.LIMIT\_currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

+

(SELECT

CASE WHEN LIMIT.MAXIMUM\_TOTAL = 0

THEN '0 denominator identified'

ELSE '' END )

)

AS [EVENT]

,LEFT(LIMIT.CREDIT\_LINE,7) AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,LIMIT.AVAIL\_AMT AS orig\_maturityValue -- Reasonableness Test

,LIMIT.LIMIT\_currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.FBNK\_LIMIT LIMIT

-- Collateral right table is the intermediary table to get to collateral table from FBNK\_LIMIT

LEFT JOIN SRC.FBNK\_COLLATERAL\_RIGHT RIG

ON RIG.limit\_id = limit.id

AND limit.collateral\_code = RIG.collateral\_code

AND LIMIT.asofdate = RIG.asofdate

--join logic on only first two parts of collateral ID

LEFT JOIN SRC.FBNK\_COLLATERAL COL

ON LEFT(COL.id,7) = LEFT(RIG.id,7)

AND substring((substring (COL.id,CHARINDEX('.', COL.id)+1,50)), 1, charindex('.',(substring (COL.id,CHARINDEX('.', COL.id)+1,50)))-1)

= substring (RIG.id,CHARINDEX('.', RIG.id)+1,50)

AND RIG.asofdate = COL.asofdate

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.ID = LEFT(LIMIT.CREDIT\_LINE,7)

AND CUST.ASOFDATE = LIMIT.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = LIMIT.co\_code

LEFT JOIN app.US\_FR2052A\_REF\_COLLATERAL Ref\_col

on ref\_col.CollateralType = COL.Collateral\_type

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = CUST.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = LEFT(LIMIT.CREDIT\_LINE,7)

WHERE

LIMIT.ASOFDATE = @AsOfDate

AND isnull(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

--To eliminate "divide by 0" ISSUE

AND TOTAL\_OS <> @TOTALOS

AND TOTAL\_OS IS NOT NULL

AND (

MAXIMUM\_TOTAL <> @MAXTOTAL

AND MAXIMUM\_TOTAL IS NOT NULL

)

--To identify records at the global level

AND LIMIT.COMMITTED\_YN = @COMMITTED

and LEFT(RIGHT(CREDIT\_LINE\_NO,7),4) IN (@LIMREF)

GROUP BY

ref\_en.ReportingEntityDescription

,Product

,CUST.SECTOR

,CUST.Industry

,CUST.ID

,Ref\_cp.CID

,Ref\_cp.CounterParty

,ref\_col.CollateralClassFR2052A

,ref\_Internal\_cp.INTERNAL

,ref\_Internal\_cp.InternalCounterParty

,ref\_rep.ReportType

,LIMIT.CO\_CODE

,LIMIT.credit\_line\_NO

,LIMIT.credit\_line

,limit.limit\_currency

,RIG.ID

,LIMIT.AVAIL\_AMT

,TOTAL\_OS

,LIMIT.MAXIMUM\_TOTAL

,RIG.id

,col.collateral\_code

,LIMIT.COMMITTED\_YN

,ref\_en.Company\_name

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Trade Finance

\*\* PID: TSD - O.O.7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.O.7'

SET @Portfolio = 'Trade Finance'

SET @Currency = 'USD'

SET @DESC1 = 'DESC1'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,NULL AS [CID]

,NULL AS [CounterParty]

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.DEAL\_MAT\_DATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.DEAL\_MAT\_DATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.DEAL\_MAT\_DATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.DEAL\_MAT\_DATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.DEAL\_MAT\_DATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.DEAL\_MAT\_DATE, '> 1')

END

END

END AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.[t\_fmddata] FMD

LEFT JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = COMP.COMPANY\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = FMD.sector

AND ref\_cp.ReportingEntity = ref\_en.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.ReportingEntityDescription

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.ASOFDATE = @ASOFDATE

AND FMD.DESC1 IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @DESC1)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding

-- \*\* PID: O.O.19 (CHB and LAB)

-- Expected Count = 76

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.O.19'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @Category = (SELECT DISTINCT Filter -- Category 21038

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Category'

AND Module = 'O.O.19\_CHB\_LAB'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- primeBrokerage = N

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @CompanyName1 = (SELECT DISTINCT Filter -- CompanyName = 'CHI'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CompanyName'

AND Module = 'CHI')

SET @CompanyName = (SELECT DISTINCT Filter -- CompanyName = 'LAB'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CompanyName'

AND Module = 'LAB')

SET @Category = 'Category'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'T\_FMDDATA | '+CAST(fmd.[N\_ID] AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN MM.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,NULL AS [CID]

,NULL AS [CounterParty]

,DBO.fn\_CurrencyExchange

(MM.Currency,(mm.Tot\_interest\_amt), @AsofDate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater

(@AsofDate\_1,fmd.Deal\_Mat\_date, 'Day 1')

AS [maturityBucket]-- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.sector IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN MM.Currency IS NULL OR MM.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN MM.Currency COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_maturityValue -- Reasonableness Test

,FMD.CURRENCY AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.[t\_fmddata] FMD

INNER JOIN SRC.COMPANY COMP

ON COMP.COMPANY\_CODE = FMD.COMPANY\_CODE

AND COMP.ASOFDATE = FMD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.company\_code = COMP.COMPANY\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = FMD.sector

AND REF\_CP.ReportingEntity = REF\_EN.ReportingEntity

LEFT JOIN SRC.[Multi\_MM] mm

ON fmd.CONTACT\_ID = mm.CONTRACT\_NO COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = FMD.Customer\_No

-- To avoid FR2052a Validation fail - where Reporting Entity is not the same as internal counterparty

--AND ref\_Internal\_cp.InternalCounterParty <> ref\_en.reportingentityDescription

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty, '') <> ref\_en.reportingentityDescription

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.Category = '21038'

AND FMD.ASOFDATE = @ASOFDATE

AND FMD.Company IN (@CompanyName, @CompanyName1)

AND DATEDIFF(DAY, @ASOFDATE, fmd.Deal\_mat\_date) BETWEEN 0 AND 30

AND fmd.COMPANY = mm.COMPANY COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS

AND FMD.CATEGORY IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @pid

AND Portfolio = @Portfolio

AND FilterName = 'Category'

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: O.O.19(nyb)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'O.O.19'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REVREASON = (SELECT DISTINCT Filter -- RevReason = ''

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'REVREASON'

)

SET @CCYAMT = (SELECT DISTINCT Filter -- CCYAMT <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT'

)

SET @SCHTYPE = 'SCHTYPE'

SET @AL = 'AL'

SET @BR = 'BR'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,CounterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ

AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,REF\_REP.Product AS Product

,NULL AS [CID]

,NULL AS [CounterPary]

,DBO.fn\_CurrencyExchange

(SCHD.CCY,(SCHD.IntAmt), @Asofdate) AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(DLDT.MDate, @Asofdate, 'Day 1')

AS [MaturityBucket]-- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN SCHD.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN SCHD.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@Asofdate AS [AsofDate]

,(

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.IntAmt AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,DLDT.MDate AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.[US\_OPICS\_SCHD] SCHD

LEFT JOIN SRC.US\_OPICS\_DLDT DLDT

ON SCHD.DEALNO = DLDT.DEALNO

AND DLDT.ASOFDATE = SCHD.AsOfDate

LEFT JOIN SRC.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND SCHD.ASOFDATE = CUST.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = DLDT.CNO

-- To avoid FR2052a Validation fail - where Reporting Entity is not the same as internal counterparty

--AND ref\_Internal\_cp.InternalCounterParty <> ref\_en.reportingentityDescription

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @REVREASON

AND DLDT.CCYAMT <> @CCYAMT

AND SCHD.SCHDTYPE IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @SCHTYPE

)

AND DLDT.VDATE <= @ASOFDATE

AND DLDT.MDATE > @ASOFDATE

AND DLDT.ASOFDATE = @ASOFDATE

AND DATEDIFF(DAY, @Asofdate,schd.ipaydate) BETWEEN 0 AND 30

AND DLDT.AL IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND DLDT.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: O.O.19 (Bond)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.O.19'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @AsofDate\_1 = @AsofDate

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @INVTYPE = (SELECT DISTINCT Filter -- InvType = 'I'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'INVTYPE'

)

SET @ACCTNGTYPE = (SELECT DISTINCT Filter -- AcctngType = 'ISSUE'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'ACCTNGTYPE'

)

SET @SETTAVGCOST = (SELECT DISTINCT Filter -- SettavgCost <> 0

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'SETTAVGCOST'

)

SET @AL = 'AL'

SET @BR = 'BR'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,CounterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

LTRIM(RTRIM('TPOS | '+LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,REF\_REP.Product AS Product

,NULL AS [CID]

,NULL AS [CounterPary]

,(DBO.fn\_CurrencyExchange

(TPOS.CCY,SECS.INTPAYAMT\_8, @AsofDate))\*TPOS.QTY

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate, @AsofDate, 'Day 1') AS [MaturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SECS.INTPAYAMT\_8 AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,NULL AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_SECM SECM

LEFT JOIN SRC.US\_OPICS\_TPOS TPOS

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECS SECS

ON SECS.SECID = TPOS.SECID

AND SECM.ASOFDATE = SECS.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = TPOS.BR

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DATEDIFF (DAY, @AsofDate, SECS.IPAYDATE) BETWEEN 1 AND 30

--AND SECS.IPAYDATE > @AsofDate

AND TPOS.INVTYPE = @INVTYPE

AND SECM.ACCTNGTYPE = @ACCTNGTYPE

AND SECS.INTPAYAMT\_8 IS NOT NULL

AND SECS.INTPAYAMT\_8 <> 0

AND TPOS.SETTAVGCOST <> @SETTAVGCOST

AND TPOS.ASOFDATE = @AsofDate

AND TPOS.BR IN (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Deposits

-- \*\* PID: O.O.19 (Interest -STB)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.O.19'

SET @Portfolio = 'Commercial and Retail Deposits'

SET @Currency = 'USD'

SET @INVTYPE = 'I'

SET @BR = 'BR'

SET @ACCTNGTYPE = 'DEFAULT'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,CID

,counterPartySector

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,[CollateralClass]

,[CollateralValue]

,[CollateralCurrency]

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturitydate

,creationDate

)

SELECT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101)) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [CID]

,NULL AS [CounterPartySector]

,CASE WHEN TPOS.CCY = 'USD' THEN (SECS.INTPAYAMT\_8 \* TPOS.QTY)

ELSE (DBO.fn\_CurrencyExchange

(TPOS.CCY,(SECS.INTPAYAMT\_8 \* TPOS.QTY), @AsofDate\_1)) END

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate, @AsofDate, 'Day 1') AS [MaturityBucket]-- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStart]

,NULL AS [ForwardBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.INTERNALCOUNTERPARTY AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage ='Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM src.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SECS.INTPAYAMT\_8 AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,NULL AS maturitydate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

AND TPOS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECS SECS

ON SECM.SECID = SECS.SECID

AND SECM.ASOFDATE = SECS.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN src.US\_OPICS\_CUST CUST

ON Cust.CNO = SECM.ISSUER

AND CUST.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BranchID = TPOS.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp\_stb

ON ref\_cp\_stb.Ctype = Cust.CTYPE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DATEDIFF(DAY,@AsofDate,SECS.IPAYDATE) BETWEEN 0 AND 30

AND SECS.IPAYDATE > @AsofDate -- Reporting Date

AND TPOS.BR IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR)

AND TPOS.INVTYPE = @INVTYPE

AND SECM.ACCTNGTYPE = @ACCTNGTYPE

AND SECS.INTPAYAMT\_8 IS NOT NULL

AND SECS.INTPAYAMT\_8 <> 0

AND TPOS.SETTAVGCOST <> 0

--AND TPOS.ASOFDATE = SECM.ASOFDATE

--AND SECS.ASOFDATE = SECS.ASOFDATE

--AND CUST.ASOFDATE = SECM.ASOFDATE

AND TPOS.ASOFDATE = @AsofDate

AND SECS.ASOFDATE = @AsofDate

AND SECM.ASOFDATE = @AsofDate

AND CUST.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives

\*\* PID: O.O.20

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET the variables

SET @PID = 'O.O.20'

SET @Portfolio = 'Derivatives'

SET @PrimeBrokerage = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'PrimeBrokerage')

SET @reportType = (SELECT DISTINCT ReportType FROM app.US\_FR2052A\_REF\_ReportType WHERE PID = @PID)

SET @PID\_OO1 = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'PID\_OO1')

SET @PID\_OS8 = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'PID\_OS8')

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,[maturityDate]

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

SRCID AS [SRCID]

,ReportScope AS [ReportScope]

,ISNULL(Currency, 'USD') AS [Currency]

,Converted AS [Converted]

,@PID AS [PID]

-- ,PID AS PID\_ORIGINAL

,ref\_type.Product AS [Product]

,MaturityValue AS [MaturityValue]

,MaturityBucket AS [MaturityBucket]

,maturityDate AS [maturityDate]

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [collateralValue]

,NULL AS [CollateralCurrency]

--,CASE WHEN @TreasuryControl = 'N'

-- THEN 'False'

-- Else 'True'

--END AS [TreasuryControl]

,CASE WHEN Internal IS NULL

THEN 'FALSE'

ELSE Internal

END AS [Internal]

,InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokerage = 'N'

THEN 'False'

Else 'True'

END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,DESC1 AS [DESC1]

,CompanyName AS [CompanyName]

,@reportType AS [ReportType]

,outflows.AsofDate AS [AsofDate]

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM (

SELECT

SrcID AS SRCID

,PID AS PID

,reportScope AS ReportScope

,Currency AS Currency

,Converted AS Converted

,ABS(MaturityValue) AS MaturityValue

,MaturityBucket AS MaturityBucket

,MaturityStart AS MaturityStart

,Maturityend AS Maturityend

,NULL AS Internal

,NULL AS InternalCounterParty

,DESC1 AS DESC1

,CompanyName AS CompanyName

,reportType AS ReportType

,maturityDate AS MaturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

app.Tgt\_securedOutflow

WHERE PID = @PID\_OS8 AND [SID] = 15

UNION

SELECT

SrcID AS SRCID

,PID AS PID

,reportScope AS ReportScope

,Currency AS Currency

,Converted AS Converted

,ABS(MaturityValue) AS MaturityValue

,MaturityBucket AS MaturityBucket

,MaturityStart AS MaturityStart

,Maturityend AS Maturityend

,NULL AS Internal

,NULL AS InternalCounterParty

,DESC1 AS DESC1

,CompanyName AS CompanyName

,reportType AS ReportType

,maturityDate AS MaturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

app.Tgt\_otherOutflow

WHERE PID = @PID\_OO1

UNION

SELECT

A.SRCID AS SRCID

,PID AS PID

,reportScope AS ReportScope

,B.Currency AS Currency

,B.Converted AS Converted

,ABS(B.MaturityAmountCurrency) AS MaturityValue

-- ,B.MaturityAmountCurrency AS MaturityValue

,MaturityBucket AS MaturityBucket

,NULL AS MaturityStart

,NULL AS Maturityend

,Internal AS Internal

,InternalCounterParty AS InternalCounterParty

,NULL AS DESC1

,CompanyName AS CompanyName

,ReportType AS ReportType

,maturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

app.Tgt\_ForeignExchange A

INNER JOIN (SELECT X.SrcID, X.MaturityAmountCurrency, X.Converted, X.Currency

-- INTO #Temp\_MaturityAmountCurrency

FROM (

SELECT A.SrcID, A.MaturityAmountCurrency1 AS MaturityAmountCurrency, A.Currency\_1\_Converted AS Converted, A.Currency\_1 AS Currency

FROM app.Tgt\_ForeignExchange A

union ALL

SELECT B.SrcID, b.MaturityAmountCurrency2 AS MaturityAmountCurrency2, B.Currency\_2\_Converted AS CONVERTED, B.Currency\_2 AS Currency

FROM app.Tgt\_ForeignExchange b

) X

WHERE X.MaturityAmountCurrency < 0) B

ON A.SrcID = B.SrcID

) AS Outflows

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_type

ON ref\_type.PID = @PID

WHERE DATEDIFF(day, @ASOFDATE, maturityDate) BETWEEN 0 AND 30

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Non - Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: O.O.2- results in no data since maturity value is 0 in the table

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'O.O.2'

SET @Portfolio = 'Other Payables and Margin Collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_otherOutflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,maturityDate

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'NB\_OO | '+CAST(NB\_OO.ID AS VARCHAR) + ' | O.O.'+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OO.ReportingEntity +' | '+CAST(NB\_OO.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_OO.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,dbo.fn\_CurrencyExchange

(NB\_OO.OriginalCurrencyOfMaturityAmount,(NB\_OO.MaturityAmountOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OO.MaturityDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OO.MaturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OO.MaturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OO.MaturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_OO.MaturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_OO.MaturityDate, '> 1')

END

END

END AS [maturityBucket] -- Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,dbo.fn\_CurrencyExchange

(NB\_OO.OriginalCurrencyOfCollateralValue,(NB\_OO.CollateralValueOriginalCurrency), @AsofDate)

AS [ColletaralValue]

,@Currency AS [CollateralCurrency]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NB\_OO.MaturityDate AS [MaturityDate]

,'N/A' AS [DESC1]

,REF\_EN.CompaNY\_Name AS [CompanyName]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS [reasonableness\_Sector] -- Reasonableness Test

,CASE WHEN NB\_OO.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS [reasonableness\_contract\_dealno] -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_OO.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OO.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_OO.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_OO.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_OO.CollateralValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_OO.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_OO NB\_OO

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_OO.Reportingentity

AND NB\_OO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_OO.CollateralCode

and ref\_col.reportingentity =nb\_oo.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_OO.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_OO.MaturityAmountOriginalCurrency IS NOT NULL

---- Uncomment later after maturity amount has a value

AND NB\_OO.MaturityAmountOriginalCurrency <> 0

AND NB\_OO.Asofdate = @ASOFDATE

AND 'O.O.'+CAST(NB\_OO.PID\_ProductID AS VARCHAR) = @PID

and NB\_OO.PortfolioName = @Portfolio

AND nb\_OO.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_other\_Inflow] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_other\_Inflow]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_other\_Inflow]

\*\* Report Type: Other Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @PID2 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @DESC1\_2 VARCHAR(10)

DECLARE @DESC1\_3 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

--DECLARE @AsofDate Date

DECLARE @Category VARCHAR (150)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Category3 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @BR VARCHAR(5)

DECLARE @BR1 VARCHAR(5)

DECLARE @AL VARCHAR(5)

DECLARE @CO\_CODE VARCHAR(50)

DECLARE @reportType VARCHAR(50)

DECLARE @TRadefinanceCategory VARCHAR(50)

DECLARE @SCHTYPE VARCHAR(10)

DECLARE @REVREASON VARCHAR(10)

DECLARE @CCYAMT VARCHAR(10)

DECLARE @ReportEntity VARCHAR(50)

DECLARE @CUSIP VARCHAR(10)

DECLARE @ACCOUNTNO VARCHAR(50)

DECLARE @CollateralClass VARCHAR (30)

DECLARE @INVTYPE VARCHAR(10)

DECLARE @CreationDate DATETIME

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives

\*\* PID: I.O.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.O.1'

SET @Portfolio = 'Derivatives'

SET @Currency = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @Category = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'Category')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail')

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,maturityDate

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT DISTINCT

M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS [SrcID] --Create unique source ID

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN M\_LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, ref\_rep.Product AS Product

,(DBO.fn\_CurrencyExchange (M\_LD.CURRENCY,(BAL.INTEREST), @ASOFDATE)) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@ASOFDATE,BAL.[DATE]) > 365

THEN dbo.fn\_MaturityDateCalculater(@ASOFDATE,BAL.[DATE], '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,ref\_en.Company\_Name AS [CompanyName]

,CASE WHEN M\_LD.CUSTOMER\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN M\_LD.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,CASE WHEN M\_LD.ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo

,BAL.[DATE] AS maturityDate

,ref\_rep.ReportType AS [ReportType]

,@ASOFDATE AS [AsofDate]

,((SELECT CASE WHEN BAL.[DATE] < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN M\_LD.Currency IS NULL OR M\_LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN M\_LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )) AS [EVENT] -- Reasonableness Test

,M\_LD.CUSTOMER\_ID AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,BAL.INTEREST AS orig\_maturityValue -- Reasonableness Test

,M\_LD.CURRENCY AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.MULTI\_LD M\_LD

LEFT JOIN SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

ON M\_LD.ID = BAL.ID

AND M\_LD.CO\_CODE = BAL.CO\_CODE

AND M\_LD.ASOFDATE = BAL.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = M\_LD.CO\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = M\_LD.CUSTOMER\_ID

AND ref\_en.ReportingEntity = ref\_Internal\_cp.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

ISNULL (ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND M\_LD.CATEGORY = @Category -- Per the DAST, limit category filter to 21074

AND BAL.INTEREST <> 0

AND BAL.INTEREST IS NOT NULL

AND BAL.[DATE] > @ASOFDATE

AND M\_LD.ASOFDATE = @ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Wholesale Funding Transactions

-- \*\* PID: TSD - I.O.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.O.6'

SET @Portfolio = 'Wholesale Funding Transactions'

SET @Currency = 'USD'

SET @SCHTYPE = 'SCHDTYPE'

SET @BR = 'BR'

SET @AL = 'AL'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage')

SET @REVREASON = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'REVREASON')

SET @CCYAMT = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CCYAMT')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,MaturityDate

,creationDate

)

SELECT

'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ AS [SrcID] -- Create Unique Source ID

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN SCHD.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS Product

,DBO.fn\_CurrencyExchange

(SCHD.CCY ,(SCHD.INTAMT), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@AsofDate,@AsofDate, 'Day 30') AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,ref\_en.BranchName AS [CompanyName]

,CASE WHEN CUST.CNO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector

,CASE WHEN SCHD.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN SCHD.DEALNO IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN SCHD.CCY IS NULL OR SCHD.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN SCHD.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )) AS [EVENT] -- Reasonableness Test

,CUST.CNO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SCHD.INTAMT AS orig\_maturityValue -- Reasonableness Test

,SCHD.CCY AS orig\_currency -- Reasonableness Test

,NULL AS MaturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM

SRC.US\_OPICS\_DLDT DLDT

LEFT JOIN SRC.[US\_OPICS\_CUST] CUST

ON DLDT.CNO = CUST.CNO

AND DLDT.ASOFDATE = CUST.ASOFDATE

LEFT JOIN SRC.[US\_OPICS\_SCHD] SCHD

ON SCHD.DEALNO = DLDT.DEALNO

AND DLDT.ASOFDATE = SCHD.AsOfDate

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_en

ON ref\_en.BRANCHID = DLDT.BR

LEFT JOIN app.US\_FR2052A\_REF\_CounterPartySTB ref\_cp

ON ref\_cp.Ctype = CUST.Ctype

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = DLDT.CNO

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND DLDT.BR = SCHD.BR

AND DLDT.REVREASON = @REVREASON --Revreason is ''

AND DLDT.CCYAMT <> @CCYAMT --CCYAMT = 0

AND SCHD.SCHDTYPE IN (SELECT DISTINCT Filter --SCHDTYPE = M or R or P

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @SCHTYPE

)

AND DLDT.MDATE > @ASOFDATE

AND DLDT.ASOFDATE = @ASOFDATE

AND DATEDIFF(DAY, @ASOFDATE,IPAYDATE) BETWEEN 0 AND 30

AND DLDT.AL IN (SELECT DISTINCT Filter --AL = A

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @AL

)

AND DLDT.BR IN (SELECT DISTINCT Filter --BR = 01, or 02, or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investments

\*\* PID: I.O.6 (Data Mart)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--SET VARIABLES

SET @PID = 'I.O.6'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @Currencyfilter = 'USD'

SET @BR = 'BR'

SET @AL = 'AL'

SET @CO\_CODE = 'Company\_Code'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @Category = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CategoryCode')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,MaturityDate

,creationDate

)

SELECT

M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS [SrcID] --Unique Source ID

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN M\_LD.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,(DBO.fn\_CurrencyExchange (M\_LD.Currency ,(MLSDD.INTEREST), @ASOFDATE))AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@ASOFDATE,@ASOFDATE, 'Day 30') AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,NULL AS [DESC1]

,ref\_en.Company\_Name AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@ASOFDATE AS [AsofDate]

,(

(SELECT

CASE WHEN M\_LD.Currency IS NULL OR M\_LD.Currency = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN M\_LD.Currency NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,CASE WHEN M\_LD.CUSTOMER\_ID IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN M\_LD.CO\_CODE IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN M\_LD.CONTRACT\_NO IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,M\_LD.CUSTOMER\_ID AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,MLSDD.INTEREST AS orig\_maturityValue -- Reasonableness Test

,M\_LD.Currency AS orig\_currency -- Reasonableness Test

,NULL AS MaturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM [SRC].MULTI\_LD M\_LD

LEFT JOIN [SRC].[US\_T24\_MULTI\_LD\_BALANCE] MLSDD

ON M\_LD.ID = MLSDD.ID

AND M\_LD.ASOFDATE = MLSDD.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = M\_LD.CO\_CODE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = M\_LD.CUSTOMER\_ID

WHERE

ISNULL (ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND MLSDD.CO\_CODE IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @CO\_CODE) --Company Code - (US0010004 = BANK OF CHINA CHI BR;US0010005 = BANK OF CHINA CHI IBF;US0010006 = BANK OF CHINA LA BR;US0010007 = BANK OF CHINA LA IBF;)

AND M\_LD.CATEGORY = @Category --Category = 20156

AND MLSDD.INTEREST <> 0

AND MLSDD.INTEREST IS NOT NULL

AND DATEDIFF(DAY, @ASOFDATE,MLSDD.[DATE]) BETWEEN 0 AND 30

AND MLSDD.[DATE] > @ASOFDATE

AND M\_LD.ASOFDATE = @ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investments

\*\* PID: I.O.6 (MIS\_DW)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.O.6'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

--SET @AsofDate\_1 = '12/30/2016'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @BR = 'BR'

SET @AL = 'AL'

SET @INVTYPE = (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'INVTYPE')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate)

SELECT

LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(TPOS.SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50)))) AS [SrcID] --Unique Source ID

,ref\_stb.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS Product

,(DBO.fn\_CurrencyExchange (TPOS.CCY ,(SECS.INTPAYAMT\_8 ), @ASOFDATE)) \* TPOS.QTY AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,dbo.fn\_MaturityDateCalculater(@ASOFDATE,@ASOFDATE, 'Day 30') AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,NULL AS [DESC1]

,ref\_stb.BranchName AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@ASOFDATE AS [AsofDate]

,(

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,CASE WHEN SECM.ISSUER IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,SECS.INTPAYAMT\_8 AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,NULL AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECS SECS

ON TPOS.SECID = SECS.SECID

AND TPOS.ASOFDATE = SECS.ASOFDATE

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON SECS.SECID = SECM.SECID

AND SECS.ASOFDATE = SECM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON TPOS.BR = ref\_stb.BranchID

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_stb.ReportingEntityDescription

AND TPOS.BR IN (SELECT DISTINCT Filter --BR = 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = @BR

)

AND (DATEDIFF(DAY, @ASOFDATE, SECS.IPAYDATE) BETWEEN 1 AND 30)

AND TPOS.INVTYPE <> @INVTYPE --INVTYPE = I

AND TPOS.settavgcost <> 0

AND TPOS.SETTQTY <> 0

AND TPOS.ASOFDATE = @ASOFDATE

and SECS.INTPAYAMT\_8 <> 0

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Derivatives

\*\* PID: I.O.7

\*\* Comment: PID Consists of I.O.1+I.A.6+S.FX.2 The queries are unioned together

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET the variables

SET @PID = 'I.O.7'

SET @Portfolio = 'Derivatives'

SET @TreasuryControl = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = 'TreasuryControl')

SET @PrimeBrokerage = (SELECT DISTINCT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND Filter = 'N')

SET @reportType = (SELECT DISTINCT ReportType FROM app.US\_FR2052A\_REF\_ReportType WHERE PID = @PID)

SET @PID1 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'PID\_IO1')

SET @PID2 = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID AND FilterName = 'PID\_IA6')

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,maturityDate

,reportType

,AsofDate

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

SRCID AS [SRCID]

,ReportScope AS [ReportScope]

,ISNULL(Currency, 'USD') AS [Currency]

,Converted AS [Converted]

,@PID AS [PID]

,ref\_type.Product AS [Product]

,MaturityValue AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,MaturityBucket AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [collateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'N'

THEN 'False'

Else 'True'

END AS [TreasuryControl]

,Internal AS [Internal]

,InternalCounterParty AS [InternalCounterParty]

,CASE WHEN @PrimeBrokerage = 'N'

THEN 'False'

Else 'True'

END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,DESC1 AS [DESC1]

,CompanyName AS [CompanyName]

,maturityDate AS [maturityDate]

,@reportType AS [ReportType]

,inflows.AsofDate AS [AsofDate]

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

-- UNION QUERY BELOW (I.0.1 + I.A.6 + S.FX.2)

(

SELECT

SrcID AS SRCID

,@PID AS [PID]

,PID AS PID\_ORIGINAL

,reportScope AS ReportScope

,Currency AS Currency

,Converted AS Converted

,ABS(MaturityValue) AS MaturityValue

,MaturityBucket AS MaturityBucket

,MaturityStart AS MaturityStart

,Maturityend AS Maturityend

,Internal AS Internal

,InternalCounterParty AS InternalCounterParty

,DESC1 AS DESC1

,CompanyName AS CompanyName

,reportType AS ReportType

,maturityDate AS maturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

APP.Tgt\_otherInflow

WHERE

PID = @PID1

UNION

SELECT

SrcID AS SRCID

,@PID AS [PID]

,PID AS PID\_ORIGINAL

,reportScope AS ReportScope

,Currency AS Currency

,Converted AS Converted

,ABS(MarketValue) AS MaturityValue

,MaturityBucket AS MaturityBucket

,MaturityStart AS MaturityStart

,Maturityend AS Maturityend

,NULL AS Internal

,NULL AS InternalCounterParty

,DESC1 AS DESC1

,CompanyName AS CompanyName

,reportType AS ReportType

,maturityDate AS maturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

APP.Tgt\_assetInflow

WHERE

PID = @PID2

UNION

SELECT

A.SRCID AS SRCID

,@PID AS [PID]

,PID AS PID\_ORIGINAL

,reportScope AS ReportScope

,B.Currency AS Currency

,B.Converted AS Converted

,B.MaturityAmountCurrency AS MaturityValue

,MaturityBucket AS MaturityBucket

,NULL AS MaturityStart

,NULL AS Maturityend

,Internal AS Internal

,InternalCounterParty AS InternalCounterParty

,NULL AS DESC1

,CompanyName AS CompanyName

,ReportType AS ReportType

,maturityDate AS maturityDate

,AsofDate AS AsofDate

,reasonableness\_Customer AS reasonableness\_Customer

,reasonableness\_Sector AS reasonableness\_Sector

,reasonableness\_Company AS reasonableness\_Company

,reasonableness\_Contract\_DealNo AS reasonableness\_Contract\_DealNo

,customerNumber AS customerNumber

,orig\_marketValue AS orig\_marketValue

,orig\_maturityValue AS orig\_maturityValue

,orig\_currency AS orig\_currency

,@CreationDate AS CreationDate

FROM

APP.Tgt\_ForeignExchange A

INNER JOIN (SELECT X.SrcID, X.MaturityAmountCurrency, X.Converted, X.Currency

FROM (

SELECT A.SrcID, A.MaturityAmountCurrency1 AS MaturityAmountCurrency, A.Currency\_1\_Converted AS Converted, A.Currency\_1 AS Currency

FROM APP.Tgt\_ForeignExchange A

union ALL

SELECT B.SrcID, b.MaturityAmountCurrency2 AS MaturityAmountCurrency2, B.Currency\_2\_Converted AS CONVERTED, B.Currency\_2 AS Currency

FROM APP.Tgt\_ForeignExchange b

) X

WHERE X.MaturityAmountCurrency > 0) B -- Select positive values only

ON A.SrcID = B.SrcID

) AS Inflows

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_type

ON ref\_type.PID = @PID

WHERE DATEDIFF(day, @ASOFDATE, maturityDate) BETWEEN 0 AND 30

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Treasury Investments

\*\* PID: I.O.8 (MIS\_DW)

\*\* Comment: There are two sections (1) SSAH and (2) TPOS which are inserted into

APP.[TGT\_assetInflow] separately. Calculation is performed at aggregation.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- SET VARIABLES

SET @PID = 'I.O.8'

SET @Portfolio = 'Treasury Investment'

SET @Currency = 'USD'

SET @DESC1 = 'DESC1'

SET @BR = 'BR'

SET @ACCOUNTNO = 'AccountNO'

SET @ReportEntity = 'ReportEntity'

SET @INVTYPE = 'INVTYPE'

SET @MaturityBucket = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'MaturityBucket')

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl')

--SET @CollateralClass = (SELECT DISTINCT FILTER FROM src.US\_FR2052A\_REF\_TableFilters

-- WHERE PID = @PID

-- AND Portfolio = @Portfolio

-- AND FilterName = @CollateralClass)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Section 1: SSAH, Report all values from SSAH. This section is to populate the QTY Field in the

Target Table with the TDYPOOLQTY

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].[Tgt\_otherInflow]

(

[SrcID]

,QTY

,[Portfolio]

,AsofDate

,[PID]

,[ReportScope]

)

SELECT

LTRIM(RTRIM('SSAH-'+LTRIM(RTRIM(BR)) +'|' + LTRIM(RTRIM(ACCOUNTNO)) + '|' +LTRIM(RTRIM(SECID))+ '|' + (CAST(LTRIM(RTRIM(SSAH.ASOFDATE)) AS VARCHAR(100))))) AS [SrcID] -- Create Unique ID

,SSAH.TDYPOOLQTY AS QTY

,@Portfolio AS [Portfolio]

,@AsofDate AS [AsofDate]

,@PID AS [PID]

,ref\_stb.ReportingEntityDescription AS [ReportScope]

FROM SRC.US\_OPICS\_SSAH SSAH

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON ref\_stb.BranchID = SSAH.BR

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

WHERE

SSAH.ASOFDATE = @AsofDate

AND SSAH.TDYPOOLQTY <> 0

AND SSAH.ACCOUNTNO IN (SELECT FILTER

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @ACCOUNTNO) --TDYPOOLQTY = Today pool quantity

AND SSAH.BR IN (SELECT DISTINCT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR)

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO [APP].[Tgt\_otherInflow]

(

[SrcID]

,[reportScope]

,[Currency]

,[Converted]

,[PID]

,[Product]

,[maturityValue]

,[QTY]

,[maturityBucket]

,[collateralClass]

,CollateralValue

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,[DESC1]

,[CompanyName]

,[Portfolio]

,[ReportType]

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationdate

)

SELECT

LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(TPOS.SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(TPOS.PORT)) + '|' + LTRIM(RTRIM(TPOS.INVTYPE)) + '|' + CAST(TPOS.COST AS VARCHAR (50)))) AS [SrcID] -- Create Unique ID

,ref\_stb.ReportingEntityDescription AS [ReportScope]

,@Currency AS [Currency]

,CASE WHEN TPOS.CCY = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

,@PID AS [PID]

,ref\_rep.Product AS [Product]

,(DBO.fn\_CurrencyExchange (TPOS.CCY ,(TPOS.PRINAMT), @AsofDate)) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,TPOS.QTY AS [QTY]

,CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,SECM.MDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,SECM.MDATE, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,(SELECT REF\_CC.COLLATERALcLASS

FROM app.US\_FR2052A\_REF\_Collateral\_MISDW REF\_CC

WHERE CHARINDEX( REF\_CC.DESCR, secm.DESCR, 1 ) > 1

AND REF\_CC.PID = @PID

AND REF\_CC.Portfolio = @Portfolio) AS [CollateralClass]

,(DBO.fn\_CurrencyExchange (TPOS.CCY ,(TPOS.PRINAMT), @AsofDate)) AS [CollateralValue]

,CASE WHEN @TreasuryControl = 'Y'

THEN 'True'

ELSE 'False'

END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y'

THEN 'True'

ELSE 'False'

END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,NULL AS [DESC1]

,ref\_stb.BranchName AS [CompanyName]

,@Portfolio AS [Portfolio]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN SECM.MDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY IS NULL OR TPOS.CCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN TPOS.CCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,CASE WHEN SECM.ISSUER IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN TPOS.BR IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,SECM.ISSUER AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,TPOS.PRINAMT AS orig\_maturityValue -- Reasonableness Test

,TPOS.CCY AS orig\_currency -- Reasonableness Test

,SECM.MDATE AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_OPICS\_TPOS TPOS

LEFT JOIN SRC.US\_OPICS\_SECM SECM

ON TPOS.SECID = SECM.SECID

and tpos.ASOFDATE = secm.asofdate

LEFT JOIN app.US\_FR2052A\_REF\_EntitySTB ref\_stb

ON TPOS.BR = ref\_stb.BranchID

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = SECM.ISSUER

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_stb.ReportingEntityDescription

AND TPOS.BR IN (SELECT DISTINCT FILTER --Branches that are 01 or 02 or 07

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @BR)

AND TPOS.INVTYPE<> (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = @INVTYPE

AND PID = @PID) -- Filter by INVTYPE is not I

AND TPOS.PRINAMT <> 0

AND TPOS.PRINAMT IS NOT NULL

AND SECM.MDATE > @AsofDate

AND SETTAVGCOST <> 0

AND SETTQTY <> 0

AND TPOS.ASOFDATE = @AsofDate

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Portfolio: Trade Finance

\*\* PID: TSD - I.O.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

---- SET VARIABLES

SET @PID = 'I.O.9'

SET @Portfolio = 'Trade Finance'

SET @Currency = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @DESC1 = 'DESC1'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,collateralClass

,collateralValue

,CollateralCurrency

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

fmd.[N\_ID] AS [SrcID]

,ref\_en.ReportingEntityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN fmd.Currency = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

, Product AS Product

,DBO.fn\_CurrencyExchange

(fmd.Currency,(fmd.deal\_balance), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,fmd.Deal\_Mat\_date) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,fmd.Deal\_Mat\_date, '> 1')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralValue]

,NULL AS [CollateralCurrency]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,DESC1 AS [DESC1]

,COMP.COMPANY\_NAME AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN fmd.Deal\_Mat\_date < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY IS NULL OR FMD.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN FMD.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,CASE WHEN FMD.CUSTOMER\_NO IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Customer] -- Reasonableness Test

,CASE WHEN FMD.SECTOR IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN FMD.COMPANY\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,CASE WHEN FMD.CONTACT\_ID IS NOT NULL THEN 'Y' ELSE 'N' END AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,FMD.CUSTOMER\_NO AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,fmd.deal\_balance AS orig\_maturityValue -- Reasonableness Test

,FMD.DEAL\_BALANCE AS orig\_currency -- Reasonableness Test

,FMD.DEAL\_MAT\_DATE AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM [SRC].[t\_fmddata] FMD

LEFT JOIN SRC.COMPANY COMP

ON FMD.COMPANY\_CODE = COMP.COMPANY\_CODE

AND FMD.ASOFDATE = COMP.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = COMP.company\_code

LEFT JOIN app.US\_FR2052A\_REF\_CounterParty ref\_cp

ON ref\_cp.SECTOR = FMD.sector

AND ref\_CP.ReportingEntity = ref\_en.ReportingEntity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.Customer\_ID = FMD.Customer\_No

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.ReportingEntityDescription

AND FMD.DEAL\_LOCAL\_BAL IS NOT NULL

AND FMD.DEAL\_LOCAL\_BAL <> 0

AND FMD.DESC1 IN (SELECT FILTER -- DESC1 = 4580

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND FilterName = @DESC1)

AND FMD.ASOFDATE = @ASOFDATE

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Commercial and Retail Loans

-- \*\* PID: I.O.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.O.9'

SET @Portfolio = 'Commercial and Retail Loans'

SET @CURRENCY = 'USD'

SET @DESC1 = 'DESC1'

SET @Sector = 'Sector'

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @CUSIP = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'CUSIP'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @ReportEntity = (SELECT filter

FROM src.US\_FR2052A\_REF\_TableFilters

WHERE FilterName = 'ReportEntity')

IF OBJECT\_ID(N'tempdb..#Temp') IS NOT NULL

BEGIN

DROP TABLE #Temp

END

IF OBJECT\_ID(N'tempdb..#Temp2') IS NOT NULL

BEGIN

DROP TABLE #Temp2

END

CREATE TABLE #TEMP

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,REVIEW\_FREQUENCY VARCHAR(100)

,STARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

CREATE TABLE #TEMP2

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,STARTDATE DATE

,EFFECTIVESTARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

INSERT INTO #TEMP

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,REVIEW\_FREQUENCY

,STARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,LIMIT\_CURRENCY

--,REVIEW\_FREQUENCY

,ISNULL((CAST(AVAIL\_AMT AS DECIMAL(30,15)) \* (CAST(SUBSTRING(CUSIP\_NO,1,CHARINDEX('%',CUSIP\_NO)-1) AS DECIMAL(30,15))/100))/NULLIF((12/CAST(LEFT(SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) AS DECIMAL )),0),0) AS AMOUNT

,REVIEW\_FREQUENCY

,LEFT(REVIEW\_FREQUENCY, 8) AS STARTDATE

,EXPIRY\_DATE

,CAST(round((DATEDIFF(MONTH,cast(LEFT(REVIEW\_FREQUENCY,8) as DATE),CAST(EXPIRY\_DATE AS DATE)))/cast(LEFT(SUBSTRING (REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) as decimal)+0.4,0) AS INTEGER) AS FREQUENCY

,LIMIT.CO\_CODE

,ASOFDATE

FROM SRC.FBNK\_LIMIT LIMIT

WHERE AVAIL\_AMT NOT LIKE '%]%'

AND CUSIP\_NO IS NOT NULL

AND CUSIP\_NO <> @CUSIP

AND AVAIL\_AMT <> '0'

AND AVAIL\_AMT IS NOT NULL

AND LIMIT.ASOFDATE = @ASOFDATE

INSERT INTO #TEMP2

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,DATEADD(MONTH, (B.RN-1)\* LEFT((SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY))),2),STARTDATE) AS EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

FROM #TEMP A

CROSS JOIN (SELECT TOP (SELECT ISNULL(MAX(FREQUENCY)+1,0) FROM #TEMP) rn = ROW\_NUMBER()

OVER (ORDER BY [object\_id])

FROM sys.all\_columns

ORDER BY [object\_id])B

WHERE B.RN < = A.FREQUENCY

ORDER BY CREDIT\_LINE\_NO

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

--Insert scripts for target table

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,maturityDate

,CustomerNumber

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,DESC1

,CompanyName

,reportType

,AsofDate

,[EVENT]

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

lim.CREDIT\_LINE\_NO + CAST(EFFECTIVESTARTDATE AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS [ReportScope]

,@CURRENCY AS [Currency]

--If Currency is already in USD, the converted field will say 'N' else, 'Y'

,CASE WHEN lim.CURRENCY = @CURRENCY

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

-- --Get the product name from the reference table based on the PID

,Ref\_Rep.Product AS [Product]

,dbo.fn\_CurrencyExchange (currency,AMOUNT, @AsofDate) AS [MaturityAmount]

,CASE WHEN DATEDIFF(DAY,@AsofDate,EFFECTIVESTARTDATE) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,EFFECTIVESTARTDATE, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,EFFECTIVESTARTDATE) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,EFFECTIVESTARTDATE, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,EFFECTIVESTARTDATE) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,EFFECTIVESTARTDATE, '> 1')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartAmount]

,NULL AS [ForwardBucket]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [internalCounterParty]

,CASE WHEN @PrimeBrokerage = 'Y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS [Portfolio]

,LIM.EFFECTIVESTARTDATE AS [MaturityDate]

,CUST.CUSTOMER\_CODE AS [CustomerNo]

,CASE WHEN CUST.CUSTOMER\_CODE IS NOT NULL

THEN 'Y'

ELSE 'N'

END AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN LIM.CO\_CODE IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,NULL AS [DESC1]

,ref\_en.Company\_Name AS [CompanyName]

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN EFFECTIVESTARTDATE < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN lim.CURRENCY IS NULL OR lim.CURRENCY = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN lim.CURRENCY NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,AMOUNT AS orig\_maturityValue -- Reasonableness Test

,currency AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM

#TEMP2 LIM

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Ref\_Rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.Company\_Code = LIM.Co\_Code

LEFT JOIN SRC.FBNK\_CUSTOMER CUST

ON CUST.CUSTOMER\_CODE = LEFT (LIM.CREDIT\_LINE\_NO,7)

AND CUST.ASOFDATE = LIM.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterPartySTB ref\_Internal\_cp

ON ref\_Internal\_cp.CNO = LEFT (LIM.CREDIT\_LINE\_NO,7)

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* NON BRANCH

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other payables and margin collaterals

-- \*\* PID: I.O.2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.O.2'

SET @Portfolio = 'Other Payables and Margin Collaterals'

SET @CURRENCY = 'USD'

SET @TradeFinanceCategory = 'CATEGORY'

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,[CollateralClass]

,CollateralCurrency

,collateralValue

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,maturityDate

,desc1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,creationDate

)

SELECT

'NB\_IO | '+ CAST(NB\_IO.ID AS VARCHAR) + ' | I.O.'+ CAST(NB\_IO.PID\_ProductID AS VARCHAR) + ' | '+NB\_IO.ReportingEntity +' | '+CAST(NB\_IO.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,dbo.fn\_CurrencyExchange (NB\_IO.OriginalCurrencyOfMaturityAmount,(NB\_IO.MaturityAmountOriginalCurrency), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,@Currency AS [CollateralCurrency]

,dbo.fn\_CurrencyExchange

(NB\_IO.OriginalCurrencyofCollateralValue,NB\_IO.CollateralValueOriginalCurrency, @AsofDate) AS [CollateraLValue]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS PortfolIO

,NB\_IO.MaturityDate AS [MaturityDate]

,'N/A' AS [DESC1]

,ref\_en.company\_name AS [CompanyName]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_IO.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

, ((SELECT CASE WHEN NB\_IO.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_IO.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT] -- Reasonableness Test

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_IO.CollateralValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_IO.OriginalCurrencyOfMaturityAmount AS orig\_currency -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_FR2052A\_NB\_IO NB\_IO

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IO.Reportingentity

AND NB\_IO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_IO.CollateralCode

AND ref\_col.reportingentity = NB\_IO.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_IO.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_IO.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_IO.MaturityAmountOriginalCurrency <> 0

AND NB\_IO.Asofdate = @ASOFDATE

AND 'I.O.'+CAST(NB\_IO.PID\_ProductID AS VARCHAR) = @PID

AND NB\_IO.PortfolioName = @Portfolio

and NB\_IO.ReportingEntity IN (select Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other receivable

-- \*\* PID: I.O.9

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'I.O.9'

SET @Portfolio = 'Other Receivable'

SET @CURRENCY = 'USD'

SET @TreasuryControl = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'TreasuryControl'

)

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @ReportEntity = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM APP.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO APP.Tgt\_otherInflow

(

SrcID

,reportScope

,Currency

,Converted

,PID

,product

,maturityValue

,maturityBucket

,forwardStartValue

,forwardStartBucket

,[CollateralClass]

,CollateralCurrency

,collateralValue

,treasuryControl

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,reportType

,maturityDate

,DESC1

,CompanyName

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,creationDate

)

SELECT

'NB\_IO | '+CAST(NB\_IO.ID AS VARCHAR) + ' | I.O.'+ CAST(NB\_IO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IO.ReportingEntity +' | '+CAST(NB\_IO.asofdate AS VARCHAR) AS [SrcID] -- Create Unique Source ID

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,dbo.fn\_CurrencyExchange

(NB\_IO.OriginalCurrencyOfMaturityAmount,(NB\_IO.MaturityAmountOriginalCurrency), @AsofDate) AS [maturityValue] --Calcuation performed with Foreign Currency Exchange function

,CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) BETWEEN 1 AND 60

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, 'Day1-60')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) BETWEEN 61 AND 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, 'Day61-365')

ELSE CASE WHEN DATEDIFF(DAY,@AsofDate,NB\_IO.MaturityDate) > 365

THEN dbo.fn\_MaturityDateCalculater(@AsofDate,NB\_IO.MaturityDate, '> 5')

END

END

END AS [maturityBucket] --Calculate maturity bucket using maturity bucket function

,NULL AS [ForwardStartValue]

,NULL AS [ForwardStartBucket]

,NULL AS [CollateralClass]

,NULL AS [CollateralCurrency]

,NULL AS [CollateraLValue]

,CASE WHEN @TreasuryControl = 'Y' THEN 'True'

ELSE 'False' END AS [TreasuryControl]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokeRage = 'Y' Then 'True'

Else 'False' END AS [PrimeBrokerage]

,@Portfolio AS PortfolIO

,ref\_rep.ReportType AS [ReportType]

,NB\_IO.MaturityDate AS [MaturityDate]

,'N/A' AS [DESC1]

,ref\_en.Company\_name AS [CompanyName]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_IO.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,@AsofDate AS [AsofDate]

,((SELECT CASE WHEN NB\_IO.MaturityDate < @ASOFDATE

THEN 'Maturity Date earlier that reporting date identified |' ELSE NULL END )

+

(SELECT

CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount IS NULL OR NB\_IO.OriginalCurrencyOfMaturityAmount = ''

THEN 'Null or blank currency code identified |' ELSE NULL END )

+

(SELECT

CASE WHEN NB\_IO.OriginalCurrencyOfMaturityAmount NOT IN (SELECT DISTINCT Currency\_code

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE NULL END )

) AS [EVENT] -- Reasonableness Test

,NULL AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_IO.CollateralValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,@CreationDate AS CreationDate -- Reasonableness Test

FROM SRC.US\_FR2052A\_NB\_IO NB\_IO

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_IO.Reportingentity

AND NB\_IO.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_IO.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_IO.MaturityAmountOriginalCurrency IS NOT NULL

AND NB\_IO.MaturityAmountOriginalCurrency <> 0

AND NB\_IO.Asofdate = @ASOFDATE

AND 'I.O.'+CAST(NB\_IO.PID\_ProductID AS VARCHAR) = @PID

and NB\_IO.PortfolioName = @Portfolio+ 's and deposit'

and NB\_IO.ReportingEntity IN (select Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity)

--and NB\_IO.PortfolioName = @Portfolio

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_MonetaryValueCheck] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_MonetaryValueCheck]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Check if the signs (positive or negative) of all the monetary values are aligned

\*\* for each target table

\*\* Author: Deloitte

\*\* Create Date: 4/13/2017

\*\* Name: Exec Procedure [dbo].[sp\_MonetaryValueCheck]

\*\* Report Type: All Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

BEGIN TRY

IF OBJECT\_ID ('tempdb..#Temp\_TargetTableMonetaryChecks') IS NOT NULL

DROP TABLE #Temp\_TargetTableMonetaryChecks

CREATE TABLE #Temp\_TargetTableMonetaryChecks

(

PID VARCHAR(20)

,portfolio VARCHAR(500)

,Check\_MarketValue VARCHAR(20)

,Check\_MaturityValue VARCHAR(20)

,Check\_LendableValue VARCHAR(20)

,Check\_ForwardValue VARCHAR(20)

,Check\_CollateralValue VARCHAR(20)

)

INSERT INTO #Temp\_TargetTableMonetaryChecks

(

PID

,portfolio

,Check\_MarketValue

,Check\_MaturityValue

,Check\_LendableValue

,Check\_ForwardValue

,Check\_CollateralValue

)

SELECT

PID,

PORTFOLIO,

CASE WHEN (

(MIN(ISNULL(Tgt\_AI.marketValue,0)) <= 0 AND MAX(ISNULL(Tgt\_AI.marketValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_AI.marketValue,0)) >= 0 AND MAX(ISNULL(Tgt\_AI.marketValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_AI.lendablevalue,0)) <= 0 AND MAX(ISNULL(Tgt\_AI.lendablevalue,0))<=0)

OR

(MIN(ISNULL(Tgt\_AI.lendablevalue,0)) >= 0 AND MAX(ISNULL(Tgt\_AI.lendablevalue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_lendablevalue,

NULL,

NULL

FROM app.Tgt\_assetInflow Tgt\_AI

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,

PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_UI.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_UI.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_UI.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_UI.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_UI.lendablevalue,0)) <= 0 AND MAX(ISNULL(Tgt\_UI.lendablevalue,0))<=0)

OR

(MIN(ISNULL(Tgt\_UI.lendablevalue,0)) >= 0 AND MAX(ISNULL(Tgt\_UI.lendablevalue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_lendablevalue,

CASE WHEN (

(MIN(ISNULL(Tgt\_UI.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_UI.forwardStartValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_UI.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_UI.forwardStartValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

NULL

FROM app.Tgt\_unsecuredInflow Tgt\_UI

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_SI.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SI.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SI.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SI.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_SI.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SI.forwardStartValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SI.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SI.forwardStartValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_SI.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SI.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SI.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SI.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_securedInflow Tgt\_SI

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_OI.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OI.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OI.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OI.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_OI.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OI.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OI.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OI.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_otherInflow Tgt\_OI

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_WO.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_WO.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_WO.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_WO.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_WO.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_WO.forwardStartValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_WO.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_WO.forwardStartValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_WO.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_WO.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_WO.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_WO.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_wholesaleOutflow Tgt\_WO

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_SO.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SO.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SO.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SO.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_SO.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SO.forwardStartValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SO.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SO.forwardStartValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_SO.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_SO.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_SO.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_SO.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_securedOutflow Tgt\_SO

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_OO.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OO.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OO.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OO.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_OO.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OO.forwardStartValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OO.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OO.forwardStartValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_OO.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OO.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OO.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OO.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_otherOutflow Tgt\_OO

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_OD.maturityValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OD.MaturityValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OD.maturityValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OD.maturityValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

NULL,

--CASE WHEN (

-- --(MIN(ISNULL(Tgt\_OD.forwardStartValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OD.forwardStartValue,0))<=0)

-- --OR

-- --(MIN(ISNULL(Tgt\_OD.forwardStartValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OD.forwardStartValue,0))>=0)

-- --) THEN 'PASS'

-- --ELSE 'FAIL'END AS Check\_forwardStartValue,

CASE WHEN (

(MIN(ISNULL(Tgt\_OD.CollateralValue,0)) <= 0 AND MAX(ISNULL(Tgt\_OD.CollateralValue,0))<=0)

OR

(MIN(ISNULL(Tgt\_OD.CollateralValue,0)) >= 0 AND MAX(ISNULL(Tgt\_OD.CollateralValue,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_CollateralValue

FROM app.Tgt\_depositsOutflow Tgt\_OD

GROUP BY PORTFOLIO,PID

UNION

SELECT

PID,PORTFOLIO,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_FX.MaturityAmountCurrency1,0)) <= 0 AND MAX(ISNULL(Tgt\_FX.MaturityAmountCurrency1,0))<=0)

OR

(MIN(ISNULL(Tgt\_FX.MaturityAmountCurrency2,0)) <= 0 AND MAX(ISNULL(Tgt\_FX.MaturityAmountCurrency2,0))<=0)

OR

(MIN(ISNULL(Tgt\_FX.MaturityAmountCurrency1,0)) >= 0 AND MAX(ISNULL(Tgt\_FX.MaturityAmountCurrency1,0))>=0)

OR

(MIN(ISNULL(Tgt\_FX.MaturityAmountCurrency2,0)) >= 0 AND MAX(ISNULL(Tgt\_FX.MaturityAmountCurrency2,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_maturityValue,

NULL,

CASE WHEN (

(MIN(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc1,0)) <= 0 AND MAX(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc1,0))<=0)

OR

(MIN(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc2,0)) <= 0 AND MAX(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc2,0))<=0)

OR

(MIN(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc1,0)) >= 0 AND MAX(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc1,0))>=0)

OR

(MIN(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc2,0)) >= 0 AND MAX(ISNULL(Tgt\_FX.ForwardStartAmountCurrenc2,0))>=0)

) THEN 'PASS'

ELSE 'FAIL'END AS Check\_forwardStartValue,

NULL

FROM app.Tgt\_ForeignExchange Tgt\_FX

GROUP BY PORTFOLIO,PID

IF EXISTS (SELECT DISTINCT Check\_MarketValue

FROM #Temp\_TargetTableMonetaryChecks

WHERE Check\_MarketValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','',a.portfolio+' | '+a.pid , 'MarketValue has opposite signs',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #Temp\_TargetTableMonetaryChecks A

where Check\_MarketValue = 'Fail'

IF EXISTS (SELECT DISTINCT Check\_MaturityValue

FROM #Temp\_TargetTableMonetaryChecks

WHERE Check\_MaturityValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','',a.portfolio+' | '+a.pid , 'MaturityValue has opposite signs',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #Temp\_TargetTableMonetaryChecks A

where Check\_MaturityValue= 'Fail'

IF EXISTS (SELECT DISTINCT Check\_LendableValue

FROM #Temp\_TargetTableMonetaryChecks

WHERE Check\_LendableValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','',a.portfolio+' | '+a.pid , 'LendableValue has opposite signs',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #Temp\_TargetTableMonetaryChecks A

where Check\_LendableValue = 'Fail'

IF EXISTS (SELECT DISTINCT Check\_ForwardValue

FROM #Temp\_TargetTableMonetaryChecks

WHERE Check\_ForwardValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','',a.portfolio+' | '+a.pid , 'ForwardValue has opposite signs',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #Temp\_TargetTableMonetaryChecks A

where Check\_ForwardValue = 'Fail'

IF EXISTS (SELECT DISTINCT Check\_CollateralValue

FROM #Temp\_TargetTableMonetaryChecks

WHERE Check\_CollateralValue = 'Fail')

INSERT INTO APP.issueTracer

SELECT distinct '','','','',a.portfolio+' | '+a.pid , 'CollateralValue has opposite signs',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Warning'

FROM #Temp\_TargetTableMonetaryChecks A

where Check\_CollateralValue = 'Fail'

SET NOCOUNT OFF

SET XACT\_ABORT OFF

--Call the next procedure i.e. to map data between source and target

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle issues

EXEC [dbo].[Proc\_InsertissueDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Map\_Tgt\_Agg\_BOC] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Map\_Tgt\_Agg\_BOC]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Map data between target and Agg\_BOC table

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [sp\_Map\_Tgt\_Agg\_BOC]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

-- DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.tgt\_unsecuredInflow;

--CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

-- ON app.tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.tgt\_assetInflow;

-- CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

-- ON app.tgt\_assetInflow(TGT\_assetInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.AGG\_BOC') AND NAME ='IX\_AGG\_BOC')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.AGG\_BOC;

-- CREATE NONCLUSTERED INDEX IX\_AGG\_BOC

-- ON app.AGG\_BOC(Agg\_BOC\_ID);

-- IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_FR2052A\_REF\_TableDetails') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON src.US\_FR2052A\_REF\_TableDetails;

-- CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

-- ON src.US\_FR2052A\_REF\_TableDetails(PID);

SET NOCOUNT ON

SET XACT\_ABORT ON

--get Total rows from soure to ensure we have received all the data and get failure/success counts

BEGIN TRY

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--insert data into map table

INSERT INTO [app].[Map\_Tgt\_Agg\_BOC]

(

TgtID

,AggBOCID

,[TargetTableName]

,[TargetFieldName]

,[AggregationReportName]

,[AggregationPortfolioName]

,[ProductID]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

Tgt\_AI.TGT\_assetInflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,tgt\_AI.Portfolio AS [AggregationPortfolioName]

,tgt\_AI.pid AS [ProductID]

,TGT\_ai.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].[TGT\_assetInflow] Tgt\_AI

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING(tgt\_AI.PID,5,100) = Agg.PID

AND ISNULL(TGT\_AI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_AI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_AI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_AI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_AI.maturityBucket,0, CHARINDEX('-', Tgt\_AI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_AI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(tgt\_AI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(tgt\_AI.treasuryControl,0) = ISNULL(AGG.treasuryControl,0)

AND ISNULL(tgt\_AI.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(tgt\_AI.SubProduct,0) = ISNULL(AGG.SubProduct,0)

AND ISNULL(Tgt\_AI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(tgt\_AI.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(tgt\_AI.CompanyName,0) = ISNULL( AGG.CompanyName,0)

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_ai.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_UI.TGT\_UnsecuredInflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_UI.Portfolio AS [AggregationPortfolioName]

,Tgt\_UI.pid AS [ProductID]

,Tgt\_UI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_UI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_UI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_UI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_UI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_UI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (TGT\_ui.maturityBucket,0, CHARINDEX('-', TGT\_ui.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_UI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_UI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(tgt\_UI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(tgt\_UI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(tgt\_UI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0) AND ISNULL(Tgt\_UI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_UI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_UI.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_UI.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_UI.CompanyName,0) = ISNULL( AGG.CompanyName,0)

---- get table details from reference table

left JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_ui.Portfolio

left JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl\_nb

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.portfolio = tgt\_UI.portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SI.Tgt\_securedInflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_SI.Portfolio AS [AggregationPortfolioName]

,Tgt\_SI.pid AS [ProductID]

,Tgt\_SI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedInflow Tgt\_SI

INNER JOIN app.AGG\_BOC agg

ON SUBSTRING( Tgt\_SI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SI.maturityBucket,0, CHARINDEX('-', Tgt\_SI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SI.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SI.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SI.EffectiveMaturityBucket,0) = ISNULL(AGG.EffectiveMaturityBucket,0)

AND ISNULL(Tgt\_SI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_SI.Unencumbered ,0) = ISNULL(AGG.Unencumbered,0)

AND ISNULL(Tgt\_SI.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SI.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

AND ISNULL(Tgt\_SI.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_SI.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_SI.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OI.Tgt\_OtherInflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_OI.Portfolio AS [AggregationPortfolioName]

,Tgt\_OI.pid AS [ProductID]

,Tgt\_OI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherInflow Tgt\_OI

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_OI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OI.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OI.TreasuryControl ,0) = ISNULL(AGG.TreasuryControl,0)

AND ISNULL(Tgt\_OI.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_OI.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_OI.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Outflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

AND ISNULL(Tgt\_SO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_SO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_SO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Outflows OS8

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

-- AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

--AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

--AND ISNULL(Tgt\_SO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_SO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_SO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND Tgt\_SO.Portfolio = 'Derivatives'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_WO.Tgt\_wholesaleOutflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_WO.Portfolio AS [AggregationPortfolioName]

,Tgt\_WO.pid AS [ProductID]

,Tgt\_WO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_WO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_WO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_WO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_WO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_WO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_WO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_WO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_WO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_WO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_WO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_WO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_WO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_WO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_WO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_WO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_WO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_WO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_WO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflow

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OO.Tgt\_OtherOutflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_OO.Portfolio AS [AggregationPortfolioName]

,Tgt\_OO.pid AS [ProductID]

,Tgt\_OO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherOutflow Tgt\_OO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_OO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_OO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_OO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_OO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_OO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_OO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Deposit outlflow

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_DO.Tgt\_depositsOutflowID AS TgtID

-- Get AggId into mapping table

,Agg.Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_DO.Portfolio AS [AggregationPortfolioName]

,Tgt\_DO.pid AS [ProductID]

,Tgt\_DO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_depositsOutflow Tgt\_DO

INNER JOIN app.[AGG\_BOC] agg

ON SUBSTRING( Tgt\_DO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_DO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_DO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_DO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_DO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_DO.maturityBucket,0, CHARINDEX('-', Tgt\_DO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_DO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_DO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_DO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_DO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_DO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_DO.insured,0) = ISNULL(AGG.Insured,0)

AND ISNULL(Tgt\_DO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_DO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_DO.Rehypothecated ,0) = ISNULL(AGG.Rehypothecated,0)

AND ISNULL(Tgt\_DO.[trigger] ,0) = ISNULL(AGG.[trigger],0)

AND ISNULL(Tgt\_DO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_DO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_DO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Non Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OI.Tgt\_OtherInflowID AS TgtID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_OI.Portfolio AS [AggregationPortfolioName]

,Tgt\_OI.pid AS [ProductID]

,Tgt\_OI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherInflow Tgt\_OI

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_OI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OI.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OI.TreasuryControl ,0) = ISNULL(AGG.TreasuryControl,0)

AND ISNULL(Tgt\_OI.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_OI.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_OI.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = tgt\_OI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*UNION

SELECT DISTINCT

Tgt\_UI.Tgt\_UnsecuredInflowID AS Tgt\_UIID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_UI.Portfolio AS [AggregationPortfolioName]

,Tgt\_UI.pid AS [ProductID]

,Tgt\_UI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_UnsecuredInflow Tgt\_UI

INNER JOIN [dbo].[AGG\_BOC] agg

ON SUBSTRING( Tgt\_UI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_UI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_UI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_UI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_UI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_UI.maturityBucket,0, CHARINDEX('-', Tgt\_UI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_UI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_UI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_UI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_UI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_UI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0) AND ISNULL(Tgt\_UI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_UI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.portfolio = tgt\_UI.portfolio

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: secured Outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS Tgt\_SOID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

AND ISNULL(Tgt\_SO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_SO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_SO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: other Outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OO.Tgt\_otherOutflowID AS Tgt\_OOID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_OO.Portfolio AS [AggregationPortfolioName]

,Tgt\_OO.pid AS [ProductID]

,Tgt\_OO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherOutflow Tgt\_OO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_OO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_OO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_OO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_OO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_OO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_OO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: wholesale Outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_WO.Tgt\_wholesaleOutflowID AS Tgt\_WOID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_WO.Portfolio AS [AggregationPortfolioName]

,Tgt\_WO.pid AS [ProductID]

,Tgt\_WO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN [app].AGG\_BOC agg

ON SUBSTRING( Tgt\_WO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_WO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_WO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_WO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_WO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_WO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_WO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_WO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_WO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_WO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_WO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_WO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_WO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_WO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_WO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_WO.DESC1,0) = ISNULL( AGG.DESC1,0)

AND ISNULL(Tgt\_WO.Portfolio,0) = ISNULL( AGG.Portfolio,0)

AND ISNULL(Tgt\_WO.CompanyName,0) = ISNULL( AGG.CompanyName,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Supplemental Informational - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_IS.Tgt\_informationalSupplementalID AS Tgt\_ISID

-- Get AggId into mapping table

,Agg\_BOC\_ID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'Agg BOC' AS [AggregationReportName]

,Tgt\_IS.Portfolio AS [AggregationPortfolioName]

,Tgt\_IS.pid AS [ProductID]

,Tgt\_IS.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM app.tgt\_informationalSupplemental Tgt\_IS

INNER JOIN app.[AGG\_BOC] agg

ON SUBSTRING( Tgt\_IS.PID,5,100) = agg.PID

AND ISNULL(Tgt\_IS.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_IS.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_IS.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_IS.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_IS.SUBproduct,0) = ISNULL(AGG.SUBproduct,0)

AND ISNULL(Tgt\_IS.[SID2],0) = ISNULL(AGG.[SID2],0)

AND ISNULL(Tgt\_IS.SUBproduct2,0) = ISNULL(AGG.SUBproduct2,0)

AND ISNULL(Tgt\_IS.product,0) = ISNULL(AGG.product,0)

AND ISNULL(Tgt\_IS.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_IS.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_IS.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_IS.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_IS.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_IS.Portfolio,0) = ISNULL(AGG.Portfolio,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_IS.PID

--EXEC dbo.sp\_Ctrl\_Tgt\_Agg\_BOC

-- EXEC dbo.sp\_Ctrl\_Tgt\_Agg\_BOC\_v2

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC [dbo].[Proc\_InsertIssueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Map\_Tgt\_Agg] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Map\_Tgt\_Agg]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Map data between target and Agg\_2052a table

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Map\_Tgt\_Agg]

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.tgt\_assetInflow;

-- CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

-- ON app.tgt\_assetInflow(TGT\_assetInflowID);

--IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('dbo.AGG\_FR2052a') AND NAME ='IX\_AGG\_2052a')

-- DROP INDEX IX\_Tgt\_AssetInflow ON dbo.AGG\_FR2052a;

-- CREATE NONCLUSTERED INDEX IX\_AGG\_2052a

-- ON dbo.AGG\_FR2052a(AGG\_FR2052aID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_FR2052A\_REF\_TableDetails') AND NAME ='IX\_Ref\_Tbl')

DROP INDEX IX\_Ref\_Tbl ON src.US\_FR2052A\_REF\_TableDetails;

CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

ON src.US\_FR2052A\_REF\_TableDetails(PID);

BEGIN TRY

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Asset Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--insert data into map table

INSERT INTO [app].[Map\_Tgt\_Agg]

(

TgtID

,Agg\_2052ID

,[TargetTableName]

,[TargetFieldName]

,[AggregationReportName]

,[AggregationPortfolioName]

,[ProductID]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

Tgt\_AI.TGT\_assetInflowID AS TgtID

--Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

-- Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,tgt\_AI.Portfolio AS [AggregationPortfolioName]

,tgt\_AI.pid AS [ProductID]

,TGT\_ai.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].[TGT\_assetInflow] Tgt\_AI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( tgt\_AI.PID,5,100) = agg.PID

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = Tgt\_AI.Portfolio

AND

ISNULL(TGT\_AI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_AI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_AI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_AI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_AI.maturityBucket,0, CHARINDEX('-', Tgt\_AI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_AI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(tgt\_AI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(tgt\_AI.treasuryControl,0) = ISNULL(AGG.treasuryControl,0)

AND ISNULL(tgt\_AI.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(tgt\_AI.SubProduct,0) = ISNULL(AGG.SubProduct,0)

AND ISNULL(Tgt\_AI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_UI.TGT\_UnsecuredInflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_UI.Portfolio AS [AggregationPortfolioName]

,Tgt\_UI.pid AS [ProductID]

,Tgt\_UI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_UI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_UI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_UI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_UI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_UI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_UI.maturityBucket,0, CHARINDEX('-', Tgt\_UI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_UI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_UI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(tgt\_UI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(tgt\_UI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(tgt\_UI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_UI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_UI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_UI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SI.Tgt\_securedInflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_SI.Portfolio AS [AggregationPortfolioName]

,Tgt\_SI.pid AS [ProductID]

,Tgt\_SI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedInflow Tgt\_SI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_SI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SI.maturityBucket,0, CHARINDEX('-', Tgt\_SI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SI.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SI.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SI.EffectiveMaturityBucket,0) = ISNULL(AGG.EffectiveMaturityBucket,0)

AND ISNULL(Tgt\_SI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_SI.Unencumbered ,0) = ISNULL(AGG.Unencumbered,0)

AND ISNULL(Tgt\_SI.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SI.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OI.Tgt\_OtherInflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_OI.Portfolio AS [AggregationPortfolioName]

,Tgt\_OI.pid AS [ProductID]

,Tgt\_OI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherInflow Tgt\_OI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_OI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OI.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OI.TreasuryControl ,0) = ISNULL(AGG.TreasuryControl,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Outflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

--AND ISNULL(Tgt\_SO.collateralValue ,0) = ISNULL(agg.collateralValue ,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Outflows OS8

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

--AND ISNULL(Tgt\_SO.collateralValue ,0) = ISNULL(agg.collateralValue ,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND Tgt\_SO.Portfolio = 'Derivatives'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflows

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_WO.Tgt\_wholesaleOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_WO.Portfolio AS [AggregationPortfolioName]

,Tgt\_WO.pid AS [ProductID]

,Tgt\_WO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_WO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_WO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_WO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_WO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_WO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_WO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_WO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_WO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_WO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_WO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_WO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_WO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_WO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_WO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_WO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Outflow

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OO.Tgt\_OtherOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_OO.Portfolio AS [AggregationPortfolioName]

,Tgt\_OO.pid AS [ProductID]

,Tgt\_OO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherOutflow Tgt\_OO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_OO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_OO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_OO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

--order by AGG\_FR2052aID

-- Eliminate NULL Values

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: derivatives outlflow OO1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT

Tgt\_OO.Tgt\_OtherOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_OO.Portfolio AS [AggregationPortfolioName]

,Tgt\_OO.pid AS [ProductID]

,Tgt\_OO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherOutflow Tgt\_OO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_OO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OO.product,0) = ISNULL(AGG.product,0)

---- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_OO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_OO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

WHERE Tgt\_OO.PID = 'O.O.1' AND Tgt\_OO.Portfolio = 'Derivatives'

-- Eliminate NULL Values

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Deposit outlflow

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_DO.Tgt\_depositsOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_DO.Portfolio AS [AggregationPortfolioName]

,Tgt\_DO.pid AS [ProductID]

,Tgt\_DO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_depositsOutflow Tgt\_DO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_DO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_DO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_DO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_DO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_DO.product,0) = ISNULL(AGG.product,0)

AND ISNULL(Tgt\_DO.maturityBucket,0) = ISNULL(AGG.maturityBucket,0)

AND ISNULL(Tgt\_DO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_DO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_DO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_DO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_DO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_DO.insured,0) = ISNULL(AGG.Insured,0)

AND ISNULL(Tgt\_DO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_DO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_DO.Rehypothecated ,0) = ISNULL(AGG.Rehypothecated,0)

AND ISNULL(Tgt\_DO.[trigger] ,0) = ISNULL(AGG.[trigger],0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Non Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Other Inflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OI.Tgt\_OtherInflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_OI.Portfolio AS [AggregationPortfolioName]

,Tgt\_OI.pid AS [ProductID]

,Tgt\_OI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherInflow Tgt\_OI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_OI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OI.maturityBucket,0, CHARINDEX('-', Tgt\_OI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OI.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OI.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OI.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OI.TreasuryControl ,0) = ISNULL(AGG.TreasuryControl,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND REF\_TBL.Portfolio = TGT\_OI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_UI.Tgt\_unsecuredInflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_UI.Portfolio AS [AggregationPortfolioName]

,Tgt\_UI.pid AS [ProductID]

,Tgt\_UI.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_UI.PID,5,100) = agg.PID

AND ISNULL(Tgt\_UI.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_UI.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_UI.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_UI.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_UI.maturityBucket,0, CHARINDEX('-', Tgt\_UI.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_UI.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_UI.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(tgt\_UI.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(tgt\_UI.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(tgt\_UI.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0) AND ISNULL(Tgt\_UI.forwardStartBucket,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_UI.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = tgt\_UI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: secured outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_SO.Tgt\_securedOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_SO.Portfolio AS [AggregationPortfolioName]

,Tgt\_SO.pid AS [ProductID]

,Tgt\_SO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_securedOutflow Tgt\_SO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_SO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_SO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_SO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_SO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_SO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_SO.maturityBucket,0, CHARINDEX('-', Tgt\_SO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_SO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_SO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_SO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_SO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_SO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_SO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_SO.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_SO.[SubProduct],0) = ISNULL(AGG.[SubProduct],0)

AND ISNULL(Tgt\_SO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_SO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

--AND ISNULL(Tgt\_SO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_SO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

AND ISNULL(Tgt\_SO.settlementMechanism ,0) = ISNULL(AGG.settlementMechanism,0)

--AND ISNULL(Tgt\_SO.collateralValue ,0) = ISNULL(agg.collateralValue ,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: other outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_OO.Tgt\_otherOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_OO.Portfolio AS [AggregationPortfolioName]

,Tgt\_OO.pid AS [ProductID]

,Tgt\_OO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_otherOutflow Tgt\_OO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_OO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_OO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_OO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_OO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_OO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_OO.maturityBucket,0, CHARINDEX('-', Tgt\_OO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_OO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_OO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_OO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_OO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_OO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_OO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_OO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_OO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_OO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_OO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

--order by AGG\_FR2052aID

-- Eliminate NULL Values

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: wholesale outflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_WO.Tgt\_wholesaleOutflowID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_WO.Portfolio AS [AggregationPortfolioName]

,Tgt\_WO.pid AS [ProductID]

,Tgt\_WO.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM [app].Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_WO.PID,5,100) = agg.PID

AND ISNULL(Tgt\_WO.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_WO.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_WO.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_WO.product,0) = ISNULL(AGG.product,0)

-- To ensure P1827D- (tgt) matches P1827D (agg)

AND ISNULL(SUBSTRING (Tgt\_WO.maturityBucket,0, CHARINDEX('-', Tgt\_WO.maturityBucket)),0) = ISNULL(AGG.MaturityStart,0)

AND ISNULL(Tgt\_WO.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_WO.CID ,0) = ISNULL(AGG.CID,0)

AND ISNULL(Tgt\_WO.[counterPartySector] ,0) = ISNULL(AGG.[counterPartySector],0)

AND ISNULL(Tgt\_WO.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_WO.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_WO.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_WO.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_WO.CollateralCurrency,0) = ISNULL(AGG.CollateralCurrency,0)

AND ISNULL(Tgt\_WO.forwardStartValue ,0) = ISNULL(AGG.forwardStartValue,0)

AND ISNULL(Tgt\_WO.forwardStartBucket ,0) = ISNULL(AGG.forwardStartBucket,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Supplemental Informational - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

UNION

SELECT DISTINCT

Tgt\_IS.Tgt\_informationalSupplementalID AS TgtID

-- Get AggId into mapping table

,AGG\_FR2052aID AS Agg\_2052ID

--Get target table details from temp table

,ref\_tbl.TargetTableName AS [TargetTableName]

-- Get target table details from temp table

,ref\_tbl.TargetTablePrimaryKeyName AS [TargetFieldName]

,'AggFR2052a' AS [AggregationReportName]

,Tgt\_IS.Portfolio AS [AggregationPortfolioName]

,Tgt\_IS.pid AS [ProductID]

,Tgt\_IS.ASOFDATE AS [AsofDate]

,getdate() AS [CreationDate]

FROM app.tgt\_informationalSupplemental Tgt\_IS

INNER JOIN [app].[AGG\_FR2052a] agg

ON SUBSTRING( Tgt\_IS.PID,5,100) = agg.PID

AND ISNULL(Tgt\_IS.reportScope,0) = ISNULL(AGG.reportScope,0)

AND ISNULL(Tgt\_IS.Currency,0) = ISNULL(AGG.Currency,0)

AND ISNULL(Tgt\_IS.Converted,0) = ISNULL(AGG.Converted,0)

AND ISNULL(Tgt\_IS.[SID],0) = ISNULL(AGG.[SID],0)

AND ISNULL(Tgt\_IS.SUBproduct,0) = ISNULL(AGG.SUBproduct,0)

AND ISNULL(Tgt\_IS.[SID2],0) = ISNULL(AGG.[SID2],0)

AND ISNULL(Tgt\_IS.SUBproduct2,0) = ISNULL(AGG.SUBproduct2,0)

AND ISNULL(Tgt\_IS.product,0) = ISNULL(AGG.product,0)

AND ISNULL(Tgt\_IS.reportType ,0) = ISNULL(AGG.reportType,0)

AND ISNULL(Tgt\_IS.internal ,0) = ISNULL(AGG.internal,0)

AND ISNULL(Tgt\_IS.internalCounterParty ,0) = ISNULL(AGG.internalCounterParty,0)

AND ISNULL(Tgt\_IS.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

AND ISNULL(Tgt\_IS.collateralClass,0) = ISNULL(AGG.collateralClass,0)

AND ISNULL(Tgt\_IS.primeBrokerage,0) = ISNULL(AGG.primeBrokerage,0)

------ get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_IS.PID

--Execute the next procedure to insert data into the control tables

-- EXEC [dbo].[sp\_Ctrl\_Tgt\_Agg\_FR2052a]

-- EXEC [dbo].[sp\_Ctrl\_Tgt\_Agg\_FR2052a\_v2]

END TRY

BEGIN CATCH

-- Execute error handling stroed procedure to get details about the errors

EXEC [dbo].[Proc\_InsertIssueDetails]

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_Map\_Src\_Tgt] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_Map\_Src\_Tgt]

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Map data between source and target tables

\*\* Author: Deloitte

\*\* Create Date: 3/7/2017

\*\* Name: Exec Procedure [dbo].[sp\_Map\_Asset\_Inflow]

\*\* Report Type: Asset Inflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

AS

BEGIN

BEGIN TRY

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_assetInflow') AND NAME ='IX\_Tgt\_AssetInflow')

-- DROP INDEX IX\_Tgt\_AssetInflow ON app.Tgt\_assetInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_AssetInflow

ON app.Tgt\_assetInflow(TGT\_assetInflowID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

--DROP INDEX IX\_Tgt\_UnsecuredInflow ON APP.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON APP.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('APP.Tgt\_securedInflow') AND NAME ='IX\_Tgt\_securedInflow')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_securedInflow

ON APP.Tgt\_securedInflow(TGT\_securedInflowID, srcid, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_T24\_MULTI\_LD\_BALANCE') AND NAME ='IX\_bal')

-- DROP INDEX IX\_bal ON SRC.US\_T24\_MULTI\_LD\_BALANCE;

CREATE NONCLUSTERED INDEX IX\_bal

ON src.US\_T24\_MULTI\_LD\_BALANCE(id, date, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.TMP\_MG\_Mortgage\_Schedule') AND NAME ='IX\_TMP')

-- DROP INDEX IX\_TMP ON SRC.TMP\_MG\_Mortgage\_Schedule;

CREATE NONCLUSTERED INDEX IX\_TMP

ON src.TMP\_MG\_Mortgage\_Schedule(CONTRACT\_NO, DUE\_DATE, asofdate);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

-- DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('src.US\_FR2052A\_REF\_TableDetails') AND NAME ='IX\_Ref\_Tbl')

-- DROP INDEX IX\_Ref\_Tbl ON src.US\_FR2052A\_REF\_TableDetails;

CREATE NONCLUSTERED INDEX IX\_Ref\_Tbl

ON src.US\_FR2052A\_REF\_TableDetails(PID);

SET NOCOUNT ON

SET XACT\_ABORT ON

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Asset inflows (Except Treasury Investment)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--insert data into target table

-- For Debuging purposes Expected Records - 80

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,Tgt\_AI.Tgt\_assetInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_AI.portfolio AS [TargetPortfolioName]

,Tgt\_AI.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_assetInflow Tgt\_AI

INNER JOIN SRC.[T\_FMDDATA] fmd

ON 'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR) = Tgt\_AI.SrcID

AND Tgt\_AI.Portfolio <> 'Treasury Investment'

and tgt\_ai.AsofDate = fmd.ASOFDATE

------ get TargetReportName from ReportType

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Tgt\_AI.PID = ref\_rep.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_AI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Asset inflows TREASURY INVESTMENT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50)))) AS VARCHAR) AS [SourceID]

,Tgt\_AI.Tgt\_assetInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_AI.portfolio AS [TargetPortfolioName]

,Tgt\_AI.pid AS [Product]

,Tgt\_AI.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_assetInflow Tgt\_AI

INNER JOIN

SRC.US\_OPICS\_TPOS TPOS

ON LTRIM(RTRIM('TPOS|'+LTRIM(RTRIM(BR)) + '|' + LTRIM(RTRIM(SECID)) + '|' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + '|' + LTRIM(RTRIM(PORT)) + '|' + LTRIM(RTRIM(INVTYPE)) + '|' + CAST(COST AS VARCHAR (50)))) = CAST(Tgt\_AI.SrcID AS VARCHAR(MAX))

------ get TargetReportName from ReportType

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Tgt\_AI.PID = ref\_rep.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_AI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Asset inflows TREASURY INVESTMENT

\*\* I.A.5 AND I.A.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(SPSH.BR+' - '+SPSH.DEALNO AS VARCHAR) AS [SourceID]

,Tgt\_AI.Tgt\_assetInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_AI.portfolio AS [TargetPortfolioName]

,Tgt\_AI.pid AS [Product]

,Tgt\_AI.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_assetInflow Tgt\_AI

INNER JOIN SRC.US\_OPICS\_SPSH SPSH

ON CAST(SPSH.BR+' - '+SPSH.DEALNO AS VARCHAR)= CAST(Tgt\_AI.SrcID AS VARCHAR)

------ get TargetReportName from ReportType

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON Tgt\_AI.PID = ref\_rep.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_AI.PID

AND ref\_tbl.Portfolio = tgt\_AI.Portfolio

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Unsecured inflows except WHOLESALE FUNDING TRANSACTIONS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,Tgt\_UI.TGT\_unsecuredInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_UI.portfolio AS [TargetPortfolioName]

,Tgt\_UI.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN SRC.[T\_FMDDATA] fmd

ON (Tgt\_UI.SrcID = 'T\_FMDDATA | '+CAST(fmd.n\_ID AS VARCHAR)

OR Tgt\_UI.SrcID = 'FMD | '+CAST(fmd.n\_ID AS VARCHAR)

OR Tgt\_UI.SrcID = 'MULTI-Account | '+CAST(fmd.n\_ID AS VARCHAR) )

and fmd.ASOFDATE = tgt\_ui.AsofDate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = Tgt\_UI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_UI.PID

--Get DESC1 value

AND( fmd.[desc1] = tgt\_UI.DESC1 COLLATE Latin1\_General\_CI\_AS

--I.U.4 doesnt have a sector code, hence the OR clause

OR fmd.Sector = tgt\_UI.Sector COLLATE Latin1\_General\_CI\_AS)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Unsecured inflows - WHOLESALE FUNDING TRANSACTIONS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'SCHD | '+ schd.DEALNO+schd.BR+CAST(INTENDDTE AS VARCHAR(50))+schd.SCHDSEQ AS [SourceID]

,Tgt\_UI.TGT\_unsecuredInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_UI.portfolio AS [TargetPortfolioName]

,Tgt\_UI.pid AS [Product]

,SCHD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN src.US\_OPICS\_SCHD SCHD

ON 'SCHD | '+ schd.DEALNO+' | '+schd.BR+' | '+CAST(INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ = Tgt\_UI.SrcID

and TGT\_ui.AsofDate = schd.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = Tgt\_UI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_UI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Unsecured inflows - WHOLESALE FUNDING TRANSACTIONS - IU2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR) AS [SourceID]

,Tgt\_UI.TGT\_unsecuredInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_UI.portfolio AS [TargetPortfolioName]

,Tgt\_UI.pid AS [Product]

,FMD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_unsecuredInflow Tgt\_UI

INNER JOIN src.T\_FMDDATA FMD

ON 'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR) = Tgt\_UI.SrcID

and TGT\_ui.AsofDate = FMD.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND ref\_tbl.Portfolio = Tgt\_UI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_UI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Unsecured inflows - Commercial and Retail Loans Loans Principal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join') IS NOT NULL DROP TABLE #Temp\_Join

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ BAL.ID+' | '+BAL.DATE AS SRCID

,CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

WHERE CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) > BAL.ASOFDATE

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

T.srcid AS [SourceID]

,TGT\_UI.Tgt\_unsecuredInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,TGT\_UI.portfolio AS [TargetPortfolioName]

,TGT\_UI.pid AS [Product]

,T.asofdate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_unsecuredInflow TGT\_UI

INNER JOIN #TEMP\_JOIN T

ON

(Tgt\_UI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_UI.ASOFDATE

and

T.asofdate = Tgt\_UI.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_UI.PID

AND REF\_TBL.PORTFOLIO = TGT\_ui.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_UI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Unsecured inflows - Commercial and Retail Loans Loans Interest

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join1') IS NOT NULL DROP TABLE #temp\_join1

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Interest | '+ BAL.ID+' | '+BAL.DATE AS SRCID

,CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join1

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

WHERE CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) > BAL.ASOFDATE

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

T.srcid AS [SourceID]

,TGT\_UI.Tgt\_unsecuredInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,TGT\_UI.portfolio AS [TargetPortfolioName]

,TGT\_UI.pid AS [Product]

,T.asofdate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_unsecuredInflow TGT\_UI

INNER JOIN #TEMP\_JOIN1 T

ON

(Tgt\_UI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_UI.ASOFDATE

and

T.asofdate = Tgt\_UI.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_UI.PID

AND REF\_TBL.PORTFOLIO = TGT\_ui.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_UI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Trade Finance

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow tgt\_SI

INNER JOIN src.[T\_FMDDATA] fmd

ON CAST(fmd.N\_ID AS VARCHAR) = tgt\_SI.SrcID

and tgt\_si.AsofDate = fmd.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_SI.PID

AND ref\_tbl.Portfolio = tgt\_SI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_SI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Wholesale Funding Transactions

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'RPDT | '+DEALNO+' | '+ rpdt.BR +' | '+ CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) +' | '+ CAST(RPDT.ASOFDATE AS VARCHAR(100)) AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,RPDT.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow tgt\_SI

INNER JOIN src.US\_OPICS\_RPDT RPDT

ON 'RPDT | '+DEALNO+' | '+ rpdt.BR +' | '+ CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) +' | '+ CAST(RPDT.ASOFDATE AS VARCHAR(100)) = tgt\_SI.SrcID

and tgt\_si.AsofDate = rpdt.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_SI.PID

AND ref\_tbl.Portfolio = tgt\_SI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_SI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Loans Principal- Commercial and Retail Loans

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join3') IS NOT NULL DROP TABLE #Temp\_Join3

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Principal | '+ BAL.ID+' | '+BAL.DATE AS SRCID

,CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join3

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

WHERE CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) > BAL.ASOFDATE

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('#temp\_join3') AND NAME ='IX\_temp')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_temp

ON #temp\_join3(srcid, asofdate);

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT

T.srcID AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,T.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow TGT\_sI

INNER JOIN #TEMP\_JOIN3 T

ON

(TGT\_sI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_sI.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_sI.PID

AND REF\_TBL.PORTFOLIO = TGT\_sI.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_sI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Loans Interest- Commercial and Retail Loans

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join4') IS NOT NULL DROP TABLE #Temp\_Join4

SELECT

'US\_T24\_MULTI\_LD\_BALANCE | Interest | '+ BAL.ID+' | '+BAL.DATE AS SRCID

,CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join4

FROM SRC.US\_T24\_MULTI\_LD\_BALANCE BAL

WHERE CAST(Convert(Varchar(20),CAST(BAL.DATE as DATE),101) as DATE) > BAL.ASOFDATE

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('#Temp\_Join4') AND NAME ='IX\_temp')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_temp

ON #Temp\_Join4(srcid, asofdate);

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT

T.srcID AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,T.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow TGT\_sI

INNER JOIN #TEMP\_JOIN4 T

ON

(TGT\_sI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_sI.ASOFDATE

and

T.asofdate = TGT\_sI.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_sI.PID

AND REF\_TBL.PORTFOLIO = TGT\_sI.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_sI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Mortgages Principal - Commercial and Retail Loans

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join5') IS NOT NULL DROP TABLE #temp\_join5

SELECT

'TMP\_MG\_Mortgage\_Schedule | Principal | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE AS SRCID

,tmp.[due\_date] AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join5

FROM SRC.TMP\_MG\_Mortgage\_Schedule tmp

WHERE tmp.[due\_date] > tmp.ASOFDATE

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('#temp\_join5') AND NAME ='IX\_temp')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_temp

ON #temp\_join5(srcid, asofdate);

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

T.srcid AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,T.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow TGT\_sI

INNER JOIN #TEMP\_JOIN5 T

ON

(TGT\_sI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_sI.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_sI.PID

AND REF\_TBL.PORTFOLIO = TGT\_sI.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_sI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Mortgages Interest - Commercial and Retail Loans

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID('tempdb..#Temp\_Join6') IS NOT NULL DROP TABLE #temp\_join6

SELECT

'TMP\_MG\_Mortgage\_Schedule | Interest | ' + TMP.CONTRACT\_NO+' | '+TMP.DUE\_DATE AS SRCID

,tmp.[due\_date] AS DATE\_CONV

,ASOFDATE AS ASOFDATE

INTO #temp\_join6

FROM SRC.TMP\_MG\_Mortgage\_Schedule tmp

WHERE tmp.[due\_date] > tmp.ASOFDATE

IF NOT EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('#temp\_join6') AND NAME ='IX\_temp')

-- DROP INDEX IX\_Tgt\_securedInflow ON APP.Tgt\_securedInflow;

CREATE NONCLUSTERED INDEX IX\_temp

ON #temp\_join6(srcid, asofdate);

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

T.srcid AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,T.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow TGT\_sI

INNER JOIN #TEMP\_JOIN6 T

ON

(TGT\_sI.SrcID = T.srcid)

AND T.ASOFDATE = TGT\_sI.ASOFDATE

and

T.asofdate = TGT\_sI.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = TGT\_sI.PID

AND REF\_TBL.PORTFOLIO = TGT\_sI.PORTFOLIO

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = TGT\_sI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Secured inflows - Wholesale Funding Transactions

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

cast('RPDT | '+DEALNO+' | '+ rpdt.BR + ' | '+CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100)) as varchar) AS [SourceID]

,tgt\_SI.Tgt\_securedInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_SI.portfolio AS [TargetPortfolioName]

,tgt\_SI.pid AS [Product]

,RPDT.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedInflow tgt\_SI

INNER JOIN SRC.US\_OPICS\_RPDT RPDT

ON cast('RPDT | '+DEALNO+' | '+ rpdt.BR + ' | '+CAST(DEALTIME AS VARCHAR(50))+' | '+CAST(YIELD\_8 AS VARCHAR(100))+' | '+CAST(CLPRICE\_8 AS VARCHAR(100))+' | '+CAST(DRPRICE\_8 AS VARCHAR(100)) + ' | '+CAST(RPDT.ASOFDATE AS VARCHAR(100)) as varchar) = CAST(tgt\_SI.[SrcID] AS VARCHAR)

and tgt\_si.asofdate = RPDT.asofdate

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_SI.PID

AND ref\_tbl.portfolio = tgt\_SI.portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_SI.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Other Outflows except WHOLESALE FUNDING TRANSACTIONS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OO.portfolio AS [TargetPortfolioName]

,tgt\_OO.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow tgt\_OO

INNER JOIN SRC.[T\_FMDDATA] fmd

ON CAST(fmd.N\_ID AS VARCHAR) = tgt\_OO.SrcID

AND tgt\_OO.AsofDate = FMD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_OO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Other Outflows - CREDIT FACILITY

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'FBNK\_LIMIT | '+LIMIT.[CREDIT\_LINE] AS [SourceID]

,tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OO.portfolio AS [TargetPortfolioName]

,tgt\_OO.pid AS [Product]

,LIMit.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow tgt\_OO

INNER JOIN src.FBNK\_LIMIT LIMit

ON ('FBNK\_LIMIT\_AVAIL\_TOTAL | '+LIMIT.[CREDIT\_LINE] = tgt\_OO.SrcID

OR 'FBNK\_LIMIT\_AVAIL\_4027 | '+LIMIT.[CREDIT\_LINE\_NO] = tgt\_OO.SrcID

OR 'FBNK\_LIMIT\_OS | '+LIMIT.[CREDIT\_LINE\_NO] = tgt\_OO.SrcID

OR 'FBNK\_LIMIT\_OS\_4027 | '+LIMIT.[CREDIT\_LINE\_NO] = tgt\_OO.SrcID)

AND tgt\_OO.AsofDate = LIMit.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_OO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Other Outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

SCHD.DEALNO+SCHD.BR+CAST(SCHD.INTENDDTE AS VARCHAR(50))+SCHD.SCHDSEQ AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,SCHD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow Tgt\_OO

INNER JOIN SRC.US\_OPICS\_SCHD SCHD

ON SCHD.DEALNO+SCHD.BR+CAST(SCHD.INTENDDTE AS VARCHAR(50))+SCHD.SCHDSEQ = Tgt\_OO.SrcID

AND TGT\_OO.AsofDate = SCHD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Other Outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR) AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,FMD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow Tgt\_OO

INNER JOIN SRC.T\_FMDDATA FMD

ON 'T\_FMDDATA | '+CAST(N\_ID AS VARCHAR) = Tgt\_OO.SrcID

AND TGT\_OO.AsofDate = FMD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Other Outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,TPOS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow Tgt\_OO

INNER JOIN SRC.US\_OPICS\_TPOS TPOS

ON 'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) = Tgt\_OO.SrcID

AND Tgt\_OO.AsofDate = TPOS.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Other Outflows - Derivatives

-- \*\* O.O.1

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,M\_LD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow Tgt\_OO

INNER JOIN src.MULTI\_LD M\_LD

ON M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID = Tgt\_OO.SrcID

AND TGT\_OO.AsofDate = M\_LD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND ref\_tbl.Portfolio = Tgt\_OO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows except WHOLESALE FUNDING TRANSACTIONS

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,tgt\_OI.Tgt\_otherinflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OI.portfolio AS [TargetPortfolioName]

,tgt\_OI.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherInflow tgt\_OI

INNER JOIN src.[T\_FMDDATA] fmd

ON CAST(fmd.N\_ID AS VARCHAR) = tgt\_OI.SrcID

AND TGT\_OI.ASOFDATE = FMD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_OI.PID

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows except WHOLESALE FUNDING TRANSACTIONS (I.O.1 - Derivatives)

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(LD.ID+'-'+LD.CUSTOMER\_ID AS VARCHAR) AS [SourceID]

,tgt\_OI.Tgt\_otherinflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OI.portfolio AS [TargetPortfolioName]

,tgt\_OI.pid AS [Product]

,LD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherInflow tgt\_OI

INNER JOIN src.MULTI\_LD LD

ON CAST(LD.ID+'-'+LD.CUSTOMER\_ID AS VARCHAR) = tgt\_OI.SrcID

AND Tgt\_oi.AsofDate = LD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_OI.PID

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows I.O.9

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

IF OBJECT\_ID(N'tempdb..#Temp') IS NOT NULL

BEGIN

DROP TABLE #Temp

END

IF OBJECT\_ID(N'tempdb..#Temp2') IS NOT NULL

BEGIN

DROP TABLE #Temp2

END

CREATE TABLE #TEMP

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,REVIEW\_FREQUENCY VARCHAR(100)

,STARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

CREATE TABLE #TEMP2

(

CREDIT\_LINE\_NO VARCHAR(100)

,CURRENCY VARCHAR(10)

,AMOUNT DECIMAL (30,15)

,STARTDATE DATE

,EFFECTIVESTARTDATE DATE

,EXPIRY\_DATE DATE

,FREQUENCY INTEGER

,CO\_CODE VARCHAR(20)

,ASOFDATE DATE

)

INSERT INTO #TEMP

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,REVIEW\_FREQUENCY

,STARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,LIMIT\_CURRENCY

--,REVIEW\_FREQUENCY

,ISNULL((CAST(AVAIL\_AMT AS DECIMAL(30,15)) \* (CAST(SUBSTRING(CUSIP\_NO,1,CHARINDEX('%',CUSIP\_NO)-1) AS DECIMAL(30,15))/100))/NULLIF((12/CAST(LEFT(SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) AS DECIMAL )),0),0) AS AMOUNT

,REVIEW\_FREQUENCY

,LEFT(REVIEW\_FREQUENCY, 8) AS STARTDATE

,EXPIRY\_DATE

,CAST(round((DATEDIFF(MONTH,cast(LEFT(REVIEW\_FREQUENCY,8) as DATE),CAST(EXPIRY\_DATE AS DATE)))/cast(LEFT(SUBSTRING (REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY)),2) as decimal)+0.4,0) AS INTEGER) AS FREQUENCY

,LIMIT.CO\_CODE

,ASOFDATE

FROM SRC.FBNK\_LIMIT LIMIT

WHERE AVAIL\_AMT NOT LIKE '%]%'

AND CUSIP\_NO IS NOT NULL

AND CUSIP\_NO <> '0'

AND AVAIL\_AMT <> '0'

AND AVAIL\_AMT IS NOT NULL

INSERT INTO #TEMP2

(

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

)

SELECT

CREDIT\_LINE\_NO

,CURRENCY

,AMOUNT

,STARTDATE

,DATEADD(MONTH, (B.RN-1)\* LEFT((SUBSTRING(REVIEW\_FREQUENCY, CHARINDEX('M',review\_frequency)+1,LEN(REVIEW\_FREQUENCY))),2),STARTDATE) AS EFFECTIVESTARTDATE

,EXPIRY\_DATE

,FREQUENCY

,CO\_CODE

,ASOFDATE

FROM #TEMP A

CROSS JOIN (SELECT TOP (SELECT ISNULL(MAX(FREQUENCY)+1,0) FROM #TEMP) rn = ROW\_NUMBER()

OVER (ORDER BY [object\_id])

FROM sys.all\_columns

ORDER BY [object\_id])B

WHERE B.RN < = A.FREQUENCY

ORDER BY CREDIT\_LINE\_NO

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

lim.CREDIT\_LINE\_NO + CAST(EFFECTIVESTARTDATE AS VARCHAR) AS [SourceID]

,tgt\_OI.Tgt\_otherinflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OI.portfolio AS [TargetPortfolioName]

,tgt\_OI.pid AS [Product]

,lim.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherInflow tgt\_OI

INNER JOIN #temp2 lim

ON lim.CREDIT\_LINE\_NO + CAST(EFFECTIVESTARTDATE AS VARCHAR) = tgt\_OI.SrcID

AND Tgt\_oi.AsofDate = lim.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_OI.PID

-- /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows WHOLESALE FUNDING TRANSACTIONS

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ AS [SourceID]

,Tgt\_OI.TGT\_otherInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OI.portfolio AS [TargetPortfolioName]

,Tgt\_OI.pid AS [Product]

,SCHD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_otherInflow Tgt\_OI

INNER JOIN src.US\_OPICS\_SCHD SCHD

ON 'SCHD | '+SCHD.DEALNO+' | '+SCHD.BR+' | '+CAST(SCHD.INTENDDTE AS VARCHAR(50))+' | '+SCHD.SCHDSEQ = Tgt\_OI.SrcID

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OI.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows Derivatives

-- \*\* I.O.1

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS VARCHAR) AS [SourceID]

,Tgt\_OI.TGT\_otherInflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OI.portfolio AS [TargetPortfolioName]

,Tgt\_OI.pid AS [Product]

,M\_LD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM [APP].Tgt\_otherInflow Tgt\_OI

INNER JOIN src.MULTI\_LD M\_LD

ON CAST(M\_LD.ID+'-'+M\_LD.CUSTOMER\_ID AS VARCHAR) = Tgt\_OI.SrcID

AND Tgt\_OI.ASOFDATE = M\_LD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND ref\_tbl.Portfolio = Tgt\_OI.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OI.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: secured outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'RPDT | '+RPDT.DEALNO+ ' | '+ RPDT.BR +' | '+ CAST(RPDT.DEALTIME AS VARCHAR(50))+' | '+ CAST(RPDT.YIELD\_8 AS VARCHAR(100))+' | '+ CAST(RPDT.CLPRICE\_8 AS VARCHAR(100))+' | '+ CAST(RPDT.DRPRICE\_8 AS VARCHAR(100)) + ' | '+ CAST(RPDT.ASOFDATE AS VARCHAR(100)) AS [SourceID]

,Tgt\_SO.Tgt\_securedOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_SO.portfolio AS [TargetPortfolioName]

,Tgt\_SO.pid AS [Product]

,RPDT.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedOutflow Tgt\_SO

INNER JOIN SRC.US\_OPICS\_RPDT RPDT

ON 'RPDT | '+RPDT.DEALNO+ ' | '+ RPDT.BR +' | '+ CAST(RPDT.DEALTIME AS VARCHAR(50))+' | '+ CAST(RPDT.YIELD\_8 AS VARCHAR(100))+' | '+ CAST(RPDT.CLPRICE\_8 AS VARCHAR(100))+' | '+ CAST(RPDT.DRPRICE\_8 AS VARCHAR(100)) + ' | '+ CAST(RPDT.ASOFDATE AS VARCHAR(100))= Tgt\_SO.SrcID

AND TGT\_SO.AsofDate = RPDT.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND ref\_tbl.Portfolio = Tgt\_SO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_SO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: secured outflows - Treasury Investment

-- \*\* O.S.8

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

SPSH.BR+' - '+SPSH.DEALNO AS [SourceID]

,Tgt\_SO.Tgt\_securedOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_SO.portfolio AS [TargetPortfolioName]

,Tgt\_SO.pid AS [Product]

,Tgt\_SO.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_securedOutflow Tgt\_SO

INNER JOIN SRC.US\_OPICS\_SPSH SPSH

ON SPSH.BR+' - '+SPSH.DEALNO = Tgt\_SO.SrcID

AND SPSH.SETTDATE > TGT\_SO.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_SO.PID

AND ref\_tbl.Portfolio = Tgt\_SO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_SO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Wholesale Outflows

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(fmd.N\_ID AS VARCHAR) AS [SourceID]

,tgt\_WO.Tgt\_wholesaleOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_WO.portfolio AS [TargetPortfolioName]

,tgt\_WO.pid AS [Product]

,fmd.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_wholesaleOutflow tgt\_WO

INNER JOIN SRC.[T\_FMDDATA] fmd

ON CAST(fmd.N\_ID AS VARCHAR) = tgt\_WO.SrcID

AND TGT\_WO.AsofDate = FMD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = tgt\_WO.PID

AND ref\_tbl.Portfolio = tgt\_WO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = tgt\_WO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: wholesale outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'SCHD | '+schd.DEALNO+' | '+schd.BR+' | '+CAST(schd.INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ AS [SourceID]

,Tgt\_WO.Tgt\_wholesaleOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_WO.portfolio AS [TargetPortfolioName]

,Tgt\_WO.pid AS [Product]

,SCHD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN SRC.US\_OPICS\_SCHD SCHD

ON 'SCHD | '+schd.DEALNO+' | '+schd.BR+' | '+CAST(schd.INTENDDTE AS VARCHAR(50))+' | '+schd.SCHDSEQ = Tgt\_WO.SrcID

AND TGT\_WO.AsofDate = SCHD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND ref\_tbl.Portfolio = Tgt\_WO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_WO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: wholesale outflows - Wholesale Funding Transactions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) AS [SourceID]

,Tgt\_WO.Tgt\_wholesaleOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_WO.portfolio AS [TargetPortfolioName]

,Tgt\_WO.pid AS [Product]

,TPOS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN SRC.US\_OPICS\_TPOS TPOS

ON 'TPOS | '+LTRIM(RTRIM(LTRIM(RTRIM(TPOS.BR)) + ' | ' + LTRIM(RTRIM(TPOS.SECID)) + ' | ' + CAST(LTRIM(RTRIM(TPOS.ASOFDATE)) AS VARCHAR(50)) + ' | ' + LTRIM(RTRIM(TPOS.PORT)) + ' | ' + LTRIM(RTRIM(TPOS.INVTYPE)) + ' | ' + CAST(TPOS.COST AS VARCHAR (50)))) = Tgt\_WO.SrcID

AND TGT\_WO.AsofDate = TPOS.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND ref\_tbl.Portfolio = Tgt\_WO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_WO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Derivative - Foreign Exchange

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

(F\_REVAL.BR+F\_REVAL.DEALNO) AS [SourceID]

,Tgt\_sfx.Tgt\_ForeignExchange\_ID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_sfx.portfolio AS [TargetPortfolioName]

,Tgt\_sfx.pid AS [Product]

,F\_REVAL.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_ForeignExchange Tgt\_sfx

INNER JOIN SRC.US\_OPICS\_FXDH\_REVAL F\_REVAL

ON (F\_REVAL.BR+F\_REVAL.DEALNO) = Tgt\_sfx.SrcID

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_sfx.PID

AND ref\_tbl.Portfolio = Tgt\_sfx.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_sfx.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Non Branch

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Report Type: Other Inflows - NB

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_IO | '+CAST(NB\_IO.ID AS VARCHAR) + ' | I.O.'+ CAST(NB\_IO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IO.ReportingEntity +' | '+CAST(NB\_IO.asofdate AS VARCHAR) AS [SourceID]

,tgt\_OI.Tgt\_otherinflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,tgt\_OI.portfolio AS [TargetPortfolioName]

,tgt\_OI.pid AS [Product]

,NB\_IO.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_IO NB\_IO

INNER JOIN APP.Tgt\_otherInflow tgt\_OI

ON Tgt\_OI.SrcID = 'NB\_IO | '+CAST(NB\_IO.ID AS VARCHAR) + ' | I.O.'+ CAST(NB\_IO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IO.ReportingEntity +' | '+CAST(NB\_IO.asofdate AS VARCHAR)

AND NB\_IO.ASOFDATE = tgt\_OI.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OI.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OI.PID

AND REF\_tbl.portfolio = Tgt\_OI.Portfolio

--AND ref\_tbl.pid = fmd.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Unsecured Inflows - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+CAST(NB\_IU.asofdate AS VARCHAR) AS [SourceID]

,Tgt\_UI.Tgt\_unsecuredinflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_UI.portfolio AS [TargetPortfolioName]

,Tgt\_UI.pid AS [Product]

,NB\_IU.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_IU NB\_IU

INNER JOIN APP.Tgt\_unsecuredInflow Tgt\_UI

ON Tgt\_UI.SrcID = 'NB\_IU | '+CAST(NB\_IU.ID AS VARCHAR) + ' | I.U.'+ CAST(NB\_IU.PID\_ProductID AS VARCHAR)+ ' | '+NB\_IU.ReportingEntity +' | '+CAST(NB\_IU.asofdate AS VARCHAR)

AND NB\_IU.ASOFDATE = TGT\_UI.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_UI.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_UI.PID

AND REF\_tbl.portfolio = Tgt\_UI.Portfolio

--AND ref\_tbl.pid = fmd.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Secured Outflow - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_OS | '+ CAST(NB\_OS.ID AS VARCHAR) + ' | O.S.'+ CAST(NB\_OS.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OS.ReportingEntity +' | '+CAST(NB\_OS.asofdate AS VARCHAR) AS [SourceID]

,Tgt\_OS.Tgt\_securedOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OS.portfolio AS [TargetPortfolioName]

,Tgt\_OS.pid AS [Product]

,NB\_OS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_OS NB\_OS

INNER JOIN APP.Tgt\_securedOutflow Tgt\_OS

ON Tgt\_OS.SrcID = 'NB\_OS | '+ CAST(NB\_OS.ID AS VARCHAR) + ' | O.S.'+ CAST(NB\_OS.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OS.ReportingEntity +' | '+CAST(NB\_OS.asofdate AS VARCHAR)

AND NB\_OS.ASOFDATE = Tgt\_OS.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OS.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OS.PID

AND REF\_tbl.portfolio = Tgt\_OS.Portfolio

--AND ref\_tbl.pid = fmd.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: other Outflow - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_OO | '+CAST(NB\_OO.ID AS VARCHAR) + ' | O.O.'+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OO.ReportingEntity +' | '+CAST(NB\_OO.asofdate AS VARCHAR) AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,NB\_OO.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_OO NB\_OO

INNER JOIN APP.Tgt\_otherOutflow Tgt\_OO

ON Tgt\_OO.SrcID = 'NB\_OO | '+CAST(NB\_OO.ID AS VARCHAR) + ' | O.O.'+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ CAST(NB\_OO.PID\_ProductID AS VARCHAR)+ ' | '+NB\_OO.ReportingEntity +' | '+CAST(NB\_OO.asofdate AS VARCHAR)

AND NB\_OO.ASOFDATE = Tgt\_OO.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND REF\_tbl.portfolio = Tgt\_OO.Portfolio

--AND ref\_tbl.pid = fmd.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Wholesale Outflow - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_OW | '+CAST(NB\_OW.ID AS VARCHAR) + ' | O.W.'+ CAST(NB\_OW.PID\_ProductID AS VARCHAR)+' | '+NB\_OW.ReportingEntity +' | '+CAST(NB\_OW.asofdate AS VARCHAR) AS [SourceID]

,Tgt\_OW.Tgt\_wholesaleOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OW.portfolio AS [TargetPortfolioName]

,Tgt\_OW.pid AS [Product]

,NB\_OW.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_OW NB\_OW

INNER JOIN APP.Tgt\_wholesaleOutflow Tgt\_OW

ON Tgt\_OW.SrcID = 'NB\_OW | '+CAST(NB\_OW.ID AS VARCHAR) + ' | O.W.'+ CAST(NB\_OW.PID\_ProductID AS VARCHAR)+' | '+NB\_OW.ReportingEntity +' | '+CAST(NB\_OW.asofdate AS VARCHAR)

AND NB\_OW.ASOFDATE = Tgt\_OW.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OW.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_OW.PID

AND REF\_tbl.portfolio = Tgt\_OW.Portfolio

--AND ref\_tbl.pid = fmd.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Report Type: Supplemental Informational - NB

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'NB\_SI | '+CAST(NB\_IS.ID AS VARCHAR) + ' | S.I.'+ CAST(NB\_IS.PID\_ProductID AS VARCHAR)+' | '+NB\_IS.ReportingEntity +' | '+CAST(NB\_IS.asofdate AS VARCHAR) AS [SourceID]

,Tgt\_IS.Tgt\_informationalSupplementalID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_IS.portfolio AS [TargetPortfolioName]

,Tgt\_IS.pid AS [Product]

,NB\_IS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM SRC.US\_FR2052A\_NB\_SI NB\_IS

INNER JOIN app.Tgt\_informationalSupplemental Tgt\_IS

ON Tgt\_IS.SrcID = 'NB\_SI | '+CAST(NB\_IS.ID AS VARCHAR) + ' | S.I.'+ CAST(NB\_IS.PID\_ProductID AS VARCHAR)+' | '+NB\_IS.ReportingEntity +' | '+CAST(NB\_IS.asofdate AS VARCHAR)

AND NB\_IS.ASOFDATE = Tgt\_IS.ASOFDATE

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_IS.PID

---- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails\_NB ref\_tbl

ON ref\_tbl.pid = Tgt\_IS.PID

AND REF\_tbl.portfolio = Tgt\_IS.Portfolio

--AND ref\_tbl.pid = fmd.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Operational Deposits - MIS\_DW Tables

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100))) AS [SourceID]

,Tgt\_DO.Tgt\_depositsOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_DO.portfolio AS [TargetPortfolioName]

,Tgt\_DO.pid AS [Product]

,TPOS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_depositsOutflow Tgt\_DO

INNER JOIN [SRC].[US\_OPICS\_TPOS] TPOS

ON CAST(Tgt\_DO.SrcID AS VARCHAR(4000)) = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100)))

AND Tgt\_DO.AsofDate = TPOS.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'TPOS'

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_DO.Portfolio))

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_DO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Operational Deposits - OD2,3,4,5 (Interest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT ALL MULTI\_MM TABLES

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100)) AS [SourceID]

,Tgt\_DO.Tgt\_depositsOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_DO.portfolio AS [TargetPortfolioName]

,Tgt\_DO.pid AS [Product]

,MM.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_depositsOutflow Tgt\_DO

INNER JOIN SRC.[Multi\_MM] MM

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'Multi\_mm|'+MM.ID+'|'+MM.CUSTOMER\_ID+'|'+MM.MATURITY\_DATE+'|'+MM.DATE\_TIME+'|'+CAST(MM.asofdate AS VARCHAR(100))

AND Tgt\_DO.AsofDate = MM.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'Multi\_MM'

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_DO.Portfolio))

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_DO.PID

WHERE Tgt\_DO.PID IN ('O.D.5','O.D.2','O.D.3','O.D.4') AND ref\_tbl.SourceTableName = 'Multi\_MM'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Operational Deposits - OD11

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT ALL MULTI\_MM TABLES

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'CDARS|'+CDARS.AccountId+'|'+CDARS.OriginalPrincipalAmount+'|'+CAST(CDARS.Asofdate as varchar(100)) AS [SourceID]

,Tgt\_DO.Tgt\_depositsOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_DO.portfolio AS [TargetPortfolioName]

,Tgt\_DO.pid AS [Product]

,CDARS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_depositsOutflow Tgt\_DO

INNER JOIN APP.US\_FR2052A\_REF\_CDARS CDARS

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = 'CDARS|'+CDARS.AccountId+'|'+CDARS.OriginalPrincipalAmount+'|'+CAST(CDARS.Asofdate as varchar(100))

AND Tgt\_DO.AsofDate = CDARS.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'US\_FR2052A\_REF\_CDARS'

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_DO.Portfolio))

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_DO.PID

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: Operational Deposits - OO19

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101)) AS [SourceID]

,Tgt\_OO.Tgt\_otherOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_OO.portfolio AS [TargetPortfolioName]

,Tgt\_OO.pid AS [Product]

,TPOS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_otherOutflow Tgt\_OO

INNER JOIN SRC.US\_OPICS\_TPOS TPOS

ON CAST(Tgt\_OO.SrcID AS VARCHAR(MAX)) = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CONVERT(VARCHAR(10),TPOS.ASOFDATE,101))

AND Tgt\_OO.AsofDate = TPOS.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_OO.PID

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_OO.Portfolio))

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_OO.PID

--/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* ReportType: Operational Deposits - OD1,2,3,4,5,6,7,9,10

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

CAST(FMD.N\_ID AS VARCHAR(4000)) AS [SourceID]

,Tgt\_DO.Tgt\_depositsOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_DO.portfolio AS [TargetPortfolioName]

,Tgt\_DO.pid AS [Product]

,FMD.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_depositsOutflow Tgt\_DO

INNER JOIN SRC.T\_FMDDATA FMD

ON CAST(Tgt\_DO.SrcID AS VARCHAR(MAX)) = CAST(FMD.N\_ID AS VARCHAR(MAX))

AND Tgt\_DO.AsofDate = FMD.ASOFDATE

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_DO.PID

AND ref\_tbl.SourceTableName = 'T\_FMDDATA'

AND LTRIM(RTRIM(ref\_tbl.Portfolio)) = LTRIM(RTRIM(Tgt\_DO.Portfolio))

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_DO.PID

WHERE Tgt\_DO.PID IN ('O.D.1'

,'O.D.10'

,'O.D.11'

,'O.D.12'

,'O.D.13'

,'O.D.2'

,'O.D.3'

,'O.D.4'

,'O.D.5'

,'O.D.6'

,'O.D.7'

,'O.D.8'

,'O.D.9')

AND ref\_tbl.SourceTableName = 'T\_FMDDATA'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* ReportType: wholesale outflows (OW16) - Wholesale Funding Transactions

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- INSERT ALL TPOS TABLES

INSERT INTO [APP].Map\_Src\_Tgt

(

[SourceID]

,[TargetID]

,[SourceDatabaseName]

,[SourceTableName]

,[SourceFieldName]

,[TargetReportName]

,[TargetPortfolioName]

,[Product]

,[AsofDate]

,[CreationDate]

)

SELECT DISTINCT

'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100))) AS [SourceID]

,Tgt\_WO.Tgt\_wholesaleOutflowID AS [TargetID]

,ref\_tbl.DatabaseName AS [SourceDatabaseName]

,ref\_tbl.SourceTableName AS [SourceTableName]

,ref\_tbl.TablePrimaryKeyName AS [SourceFieldName]

,ref\_rep.ReportType AS [TargetReportName]

,Tgt\_WO.portfolio AS [TargetPortfolioName]

,Tgt\_WO.pid AS [Product]

,TPOS.AsOfDate AS [AsofDate]

,CreationDate AS [CreationDate]

FROM APP.Tgt\_wholesaleOutflow Tgt\_WO

INNER JOIN [SRC].[US\_OPICS\_TPOS] TPOS

ON CAST(Tgt\_WO.SrcID AS VARCHAR(4000)) = 'TPOS|'+(LTRIM(RTRIM(TPOS.BR))+'|'+LTRIM(RTRIM(TPOS.COST))+'|'+LTRIM(RTRIM(TPOS.CCY))+'|'+LTRIM(RTRIM(TPOS.INVTYPE))+'|'+LTRIM(RTRIM(TPOS.PORT))+'|'+LTRIM(RTRIM(TPOS.SECID))+'|'+CAST(TPOS.ASOFDATE AS VARCHAR(100)))

-- get table details from reference table

INNER JOIN src.US\_FR2052A\_REF\_TableDetails ref\_tbl

ON ref\_tbl.pid = Tgt\_WO.PID

AND ref\_tbl.Portfolio = Tgt\_WO.Portfolio

-- get TargetReportName from ReportType

LEFT JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.pid = Tgt\_WO.PID

--Execute the next procedure to insert data into the control tables

END TRY

BEGIN CATCH

-- Execute issue handling stroed procedure to get details about the issues

EXEC Proc\_InsertissueDetails

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_InformationalSupplemental] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_InformationalSupplemental]

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Migrate data from source tables to target tables

\*\* Author: Deloitte

\*\* Create Date: 5/16/2017

\*\* Name: Exec Procedure [dbo].[sp\_InformationalSupplemental]

\*\* Report Type: Wholesale Outflow

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

-- for testing purposes: expect 24 records

BEGIN

SET NOCOUNT ON

SET XACT\_ABORT ON

-- Add nonclustered indexes to improve performance

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.Tgt\_unsecuredInflow') AND NAME ='IX\_Tgt\_UnsecuredInflow')

DROP INDEX IX\_Tgt\_UnsecuredInflow ON app.Tgt\_unsecuredInflow;

CREATE NONCLUSTERED INDEX IX\_Tgt\_UnsecuredInflow

ON app.Tgt\_unsecuredInflow(TGT\_unsecuredInflowID);

IF EXISTS(SELECT \* FROM sys.indexes WHERE object\_id = object\_id('app.US\_FR2052A\_REF\_Collateral\_MISDW') AND NAME ='IX\_Ref\_Rep')

DROP INDEX IX\_Ref\_Rep ON app.US\_FR2052A\_REF\_Collateral\_MISDW;

CREATE NONCLUSTERED INDEX IX\_Ref\_Rep

ON app.US\_FR2052A\_REF\_Collateral\_MISDW(PID);

BEGIN TRY

--Declare to enter PID and Portfolio dynamically

DECLARE @PID VARCHAR(10)

DECLARE @PID1 VARCHAR(10)

DECLARE @SubProduct VARCHAR (100)

DECLARE @Portfolio VARCHAR(100)

DECLARE @DESC1 VARCHAR(10)

DECLARE @Sector VARCHAR(10)

DECLARE @Currency VARCHAR(3)

DECLARE @MaturityBucket VARCHAR(10)

DECLARE @TreasuryControl VARCHAR (1)

DECLARE @SID INT

DECLARE @PrimeBrokerage VARCHAR(10)

DECLARE @Contac\_ID VARCHAR (5)

DECLARE @Currencyfilter VARCHAR(3)

DECLARE @AsofDate\_1 Date

DECLARE @Category VARCHAR (50)

DECLARE @Category1 VARCHAR (150)

DECLARE @Category2 VARCHAR (50)

DECLARE @Officer VARCHAR (50)

DECLARE @officer2 VARCHAR (50)

DECLARE @Unencumbered VARCHAR(10)

DECLARE @Settlement VARCHar(20)

DECLARE @CompanyName VARCHAR (50) -- NEW Variable 20170327 KBLack O.W.9

DECLARE @CompanyName1 VARCHAR (50) -- NEW Variable 20170327 KBLack O.W.9

DECLARE @AL VARCHAR(5)

DECLARE @BR VARCHAR(5)

DECLARE @CountryCode VARCHAR(50)

DECLARE @Rehypothecated VARCHAR(50)

DECLARE @CreationDate DATETIME

DECLARE @REPORTENTITY VARCHAR(50)

SET @CreationDate = (SELECT CreationDate FROM APP.GlobalVariable)

DECLARE @ASOFDATE DATE

SET @ASOFDATE = (SELECT ASOFDATE FROM APP.GlobalVariable)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Non - Branch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.2'

SET @Portfolio = 'Other Payables and Margin Collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'Y'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NB\_SI.[SID] AS [SID]

,NB\_SI.[Sub\_Product] AS [SubProduct]

,NB\_SI.[SID2] AS [SID2]

,NB\_SI.Sub\_Product2 AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,null AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency -- Reasonableness Test

,null AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.5'

SET @Portfolio = 'Other Payables and margin collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'Y'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NB\_SI.[SID] AS [SID]

,NB\_SI.[Sub\_Product] AS [SubProduct]

,NB\_SI.[SID2] AS [SID2]

,NB\_SI.Sub\_Product2 AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,null AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency -- Reasonableness Test

,null AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.15

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.15'

SET @Portfolio = 'Other Payables and margin collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'Y'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,NULL AS [SID2]

,NULL AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue] -- Calcuation performed with Foreign Currency Exchange function

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer] -- Reasonableness Test

,'N/A' AS reasonableness\_Sector -- Reasonableness Test

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company] -- Reasonableness Test

,'N/A' AS reasonableness\_Contract\_DealNo -- Reasonableness Test

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,null AS customerNumber -- Reasonableness Test

,NULL AS orig\_marketValue -- Reasonableness Test

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue -- Reasonableness Test

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency -- Reasonableness Test

,null AS maturityDate -- Reasonableness Test

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.16

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.16'

SET @Portfolio = 'Other Payables and margin collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter -- PrimeBrokerage = 'Y'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,NULL AS [SID2]

,NULL AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue]

,ref\_col.collateralclassFR2052a AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N/A' AS reasonableness\_Contract\_DealNo

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

) AS [EVENT]

,null AS customerNumber

,NULL AS orig\_marketValue

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency

,null AS maturityDate

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.17

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.17'

SET @Portfolio = 'Other Payables and margin collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,NULL AS [SID2]

,NULL AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue]

,NULL AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N/A' AS reasonableness\_Contract\_DealNo

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,null AS customerNumber

,NULL AS orig\_marketValue

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency

,null AS maturityDate

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \*\* Portfolio: Other Payables and Margin Collateral

-- \*\* PID: S.I.18

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

SET @PID = 'S.I.18'

SET @Portfolio = 'Other Payables and margin collaterals'

SET @CURRENCY = 'USD'

SET @PrimeBrokerage = (SELECT DISTINCT Filter

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE PID = @PID

AND Portfolio = @Portfolio

AND FilterName = 'PrimeBrokerage'

)

SET @REPORTENTITY = 'NbEntity'

IF NOT EXISTS (SELECT \* FROM app.ISSUETracer

WHERE ISSUEProc = @PID+' | '+@Portfolio

AND ExecutionStatus = 'Fail'

)

INSERT INTO app.Tgt\_informationalSupplemental

(

SrcID

,reportScope

,Currency

,Converted

,PID

,Product

,[sid]

,subProduct

,SID2

,subProduct2

,marketValue

,collateralClass

,internal

,internalCounterParty

,primeBrokerage

,Portfolio

,CompanyName

,DESC1

,reasonableness\_Customer

,reasonableness\_Sector

,reasonableness\_Company

,reasonableness\_Contract\_DealNo

,reportType

,AsofDate

,[EVENT]

,customerNumber

,orig\_marketValue

,orig\_maturityValue

,orig\_currency

,maturityDate

,creationDate

)

SELECT

'NB\_SI | '+CAST(NB\_SI.ID AS VARCHAR) + ' | S.I.' + CAST(NB\_SI.PID\_ProductID AS VARCHAR)+ ' | '+NB\_SI.ReportingEntity +' | '+CAST(NB\_SI.asofdate AS VARCHAR) AS [SrcID]

,ref\_en.reportingentityDescription AS ReportScope

,@Currency AS Currency

,CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue = @Currency

THEN 'False'

ELSE 'True'

END AS [Converted]

, @PID AS [PID]

,ref\_rep.Product AS Product

,NULL AS [SID]

,NULL AS [SubProduct]

,NULL AS [SID2]

,NULL AS [SubProduct2]

,dbo.fn\_CurrencyExchange

(NB\_SI.OriginalCurrencyOfMarketValue,(NB\_SI.MarketValueOriginalCurrency), @AsofDate)

AS [maturityValue]

,NULL AS [CollateralClass]

,CASE WHEN ref\_Internal\_cp.INTERNAL = 'Y' THEN 'True'

ELSE 'False' END AS [Internal]

,ref\_Internal\_cp.InternalCounterParty AS [InternalCounterPary]

,CASE WHEN @PrimeBrokerage = 'y' THEN 'True'

ELSE 'False' END AS [PrimeBrokerage]

,@Portfolio AS Portfolio

,ref\_en.Company\_Name AS [CompanyName]

,'N/A' AS [DESC1]

,'N/A' AS [reasonableness\_Customer]

,'N/A' AS reasonableness\_Sector

,CASE WHEN NB\_SI.Reportingentity IS NOT NULL THEN 'Y' ELSE 'N' END AS [reasonableness\_Company]

,'N/A' AS reasonableness\_Contract\_DealNo

,ref\_rep.ReportType AS [ReportType]

,@AsofDate AS [AsofDate]

,(

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue IS NULL OR NB\_SI.OriginalCurrencyOfMarketValue = ''

THEN 'Null or blank currency code identified |' ELSE '' END )

+

(SELECT

CASE WHEN NB\_SI.OriginalCurrencyOfMarketValue NOT IN (SELECT DISTINCT Currency\_code

FROM src.fbnk\_currency

WHERE ASOFDATE = @ASOFDATE)

THEN 'Currency code unavailable in ref tables identified |'

ELSE '' END )

)

AS [EVENT]

,null AS customerNumber

,NULL AS orig\_marketValue

,NB\_SI.MarketValueOriginalCurrency AS orig\_maturityValue

,NB\_SI.OriginalCurrencyOfMarketValue AS orig\_currency

,null AS maturityDate

,@CreationDate AS CreationDate

FROM SRC.US\_FR2052A\_NB\_SI NB\_SI

INNER JOIN app.US\_FR2052A\_REF\_ReportType ref\_rep

ON ref\_rep.PID = @PID

LEFT JOIN app.US\_FR2052A\_REF\_Entity ref\_en

ON ref\_en.ReportingEntity = NB\_SI.Reportingentity

AND NB\_SI.EntityCode = REPLACE(ref\_en.Company\_code, 'BOCA', '')

--and (

-- (ref\_en.Company\_code = 'BOCA'+NB\_IU.EntityCode

-- AND NB\_IU.Reportingentity = 'BOCA'

-- )

-- OR

-- ref\_en.Company\_code = NB\_IU.EntityCode

-- )

LEFT JOIN app.US\_FR2052A\_REF\_Collateral ref\_col

ON ref\_col.collateralType = NB\_SI.Collateralcode

and ref\_col.reportingentity = NB\_SI.Reportingentity

LEFT JOIN app.US\_FR2052A\_REF\_InternalCounterParty ref\_Internal\_cp

ON ref\_Internal\_cp.customer\_id = NB\_SI.CounterpartyCode

WHERE

ISNULL(ref\_Internal\_cp.InternalCounterParty,'') <> ref\_en.reportingentityDescription

AND NB\_SI.MarketValueOriginalCurrency IS NOT NULL

AND NB\_SI.MarketValueOriginalCurrency <> 0

AND NB\_SI.ASOFDATE = @ASOFDATE

AND 'S.I.'+CAST(NB\_SI.PID\_ProductID AS VARCHAR) = @PID

and NB\_SI.PortfolioName = @Portfolio

AND NB\_SI.Reportingentity IN (SELECT DISTINCT Filter -- PrimeBrokerage = 'N'

FROM src.US\_FR2052A\_REF\_TableFilters\_NB

WHERE FilterName = @ReportEntity

)

SET NOCOUNT OFF

SET XACT\_ABORT OFF

END TRY

BEGIN CATCH

BEGIN

--Call procedure to handle ISSUEs

EXEC [dbo].[Proc\_InsertISSUEDetails]

END

END CATCH

END

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_FR2052A\_Global] Script Date: 07/11/2017 22:35:58 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_FR2052A\_Global]

@Asofdate DATE

AS

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Desc: Execute the FR2052A Application

\*\* Author: Deloitte

\*\* Create Date: 6/16/2017

\*\* Name: Exec all procedures

\*\* Report Type: ALL Report Types

\*\* Products: All PIDs

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Reviewed By

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Change History

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Date Author Description

\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BEGIN

BEGIN TRY

/\* Truncate APP tables\*/

TRUNCATE TABLE APP.RPT\_FR2052a

TRUNCATE TABLE APP.AGG\_BOC

TRUNCATE TABLE APP.AGG\_FR2052a

TRUNCATE TABLE APP.Ctrl\_Src\_Tgt

TRUNCATE TABLE APP.Ctrl\_Tgt\_Agg

TRUNCATE TABLE APP.Ctrl\_Tgt\_Agg\_BOC

TRUNCATE TABLE APP.issueTracer

TRUNCATE TABLE APP.FRValidationTracer

TRUNCATE TABLE APP.Map\_Src\_Tgt

TRUNCATE TABLE APP.Map\_Tgt\_Agg

TRUNCATE TABLE APP.Map\_Tgt\_Agg\_BOC

TRUNCATE TABLE APP.Tgt\_assetInflow

TRUNCATE TABLE APP.Tgt\_unsecuredInflow

TRUNCATE TABLE APP.Tgt\_otherInflow

TRUNCATE TABLE APP.Tgt\_securedInflow

TRUNCATE TABLE APP.Tgt\_wholesaleOutflow

TRUNCATE TABLE APP.Tgt\_otherOutflow

TRUNCATE TABLE APP.Tgt\_securedOutflow

TRUNCATE TABLE APP.Tgt\_depositsOutflow

TRUNCATE TABLE APP.Tgt\_ForeignExchange

TRUNCATE TABLE APP.Tgt\_informationalSupplemental

/\* Run a check to see if tables used in the application exists in the system and have records\*/

EXEC dbo.sp\_RequiredTableCheck

IF @AsOfDate not in (SELECT DISTINCT asofdate

FROM SRC.FBNK\_CURRENCY

WHERE ASOFDATE = @ASOFDATE)

INSERT INTO APP.issueTracer

values( '','','','','', 'ASOFDATE does not exist in the Foreign Exchange Table',SUSER\_SNAME() ,Host\_NAME(), GETDATE(), 'Fail')

ELSE

BEGIN

/\*Truncate and populate reporting date and creation date in global varibale table\*/

TRUNCATE TABLE APP.GlobalVariable

INSERT INTO APP.GlobalVariable

SELECT @Asofdate, GETDATE()

/\* Execute Source to Target ETL Scripts \*/

EXEC dbo.sp\_CurrentReference

EXEC dbo.sp\_Asset\_Inflow

EXEC dbo.sp\_ForeignExchange

EXEC dbo.sp\_Unsecured\_Inflow

EXEC dbo.sp\_secured\_Inflow

EXEC dbo.sp\_other\_Inflow

EXEC dbo.sp\_wholesale\_outflow

EXEC dbo.sp\_secured\_outflow

EXEC dbo.sp\_deposit\_outflow

EXEC dbo.sp\_other\_outflow

EXEC dbo.sp\_InformationalSupplemental

/\* Execute Source to Target Control Scripts \*/

EXEC dbo.sp\_Ctrl\_Asset\_Inflow

EXEC dbo.sp\_Ctrl\_ForeignExchange

EXEC dbo.sp\_Ctrl\_Unsecured\_Inflow

EXEC dbo.sp\_Ctrl\_Secured\_Inflow

EXEC dbo.sp\_Ctrl\_wholesale\_outflow

EXEC dbo.sp\_Ctrl\_Secured\_Outflow

EXEC dbo.sp\_Ctrl\_deposits\_outflow

EXEC dbo.sp\_Ctrl\_Other\_outflow

EXEC dbo.sp\_Ctrl\_Informational\_Supplemental

/\* Execute Source to Target Control-logic-check to records any FAILED validations \*/

EXEC dbo.sp\_Ctrl\_src\_tgt\_logic\_check

/\* Execute event handler to record any data quality issues \*/

EXEC dbo.sp\_eventHandler

/\* Execute Source to Target map Scripts \*/

EXEC dbo.sp\_MAP\_SRC\_TGT

/\* Execute target to aggregation ETL Scripts \*/

EXEC dbo.sp\_AggFR2052a

EXEC dbo.sp\_Agg\_BOC

/\* Execute target to aggregate map Scripts \*/

EXEC dbo.sp\_Map\_Tgt\_Agg

EXEC dbo.sp\_Map\_Tgt\_Agg\_BOC

/\* Execute target to aggregate Control Scripts \*/

EXEC dbo.sp\_Ctrl\_Tgt\_Agg\_FR2052a

EXEC dbo.sp\_Ctrl\_Tgt\_Agg\_BOC

/\* Execute target to aggregate Control-logic-check to records any FAILED validations \*/

EXEC dbo.sp\_Ctrl\_tgt\_agg\_logic\_check

EXEC dbo.sp\_Ctrl\_tgt\_agg\_BOC\_logic\_check

/\* Execute checks to ensure all mandatory fields are populated\*/

EXEC dbo.sp\_RequiredFieldsCheck

/\* Execute monetary value check to ensure that, within a PID, monetary values are of the same sign \*/

EXEC dbo.sp\_MonetaryValueCheck

/\* Execute FR Validation checks on aggregated data \*/

EXEC [dbo].[sp\_FR\_Validation]

/\* Execute scripts to convert aggregation FR2052a data into XML format \*/

EXEC dbo.sp\_RPT\_FR2052a

/\*Execute scripts to generate XML\*/

EXEC dbo.sp\_XML\_GENERATOR

end

END TRY

BEGIN CATCH

EXEC [dbo].[Proc\_InsertIssueDetails]

END CATCH

SELECT \* FROM APP.[ISSUETracer] -- To see system and custom errors

SELECT \* FROM APP.FRValidationTracer -- To see FR2052A Validation errors

END

GO