/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv

DON'T RUN THIS ON A PRODUCTION DATABASE <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

20121120 mrr clean up, comment and deploy coding standard

20142401 rtt added wrapper to check for DB name like Production

20160331 rtt updated to add REV\_WEB\_FARM\_SERVER and to Support Post 10.03 Audit Trail Table names

\*/

-- @Debug = 0 causes COMMIT anything else ROLLBACK

DECLARE @DebugOn INT = 0

, @RvTrcSQLAlter INT = 0, @PrntPXPEmailUpdated INT = 0, @RPSSANUpdated INT = 0, @RPPhnUpdated INT = 0

, @RPEmailUpdated INT = 0, @REmailUpdated INT = 0, @RPPriPhnUpdated INT = 0, @LastNameChanges INT = 0

, @FirstNameChanges INT = 0, @RPCompKeysUpdated INT = 0, @dbname nvarchar (100)

-- Lets make sure that we are not running on production database

SET @dbname =(SELECT DB\_NAME() AS DataBaseName)

IF @dbname not like '%Prod%'

BEGIN

PRINT 'This does not appear to be a Production Database Process will continue ' + @dbname

BEGIN TRANSACTION

TRUNCATE TABLE rev.EPC\_GBW\_SCHL\_YR\_WS\_LOG

TRUNCATE TABLE rev.REV\_DATASET\_FILTER

TRUNCATE TABLE rev.REV\_AUDIT\_TRAIL\_PROP\_HIS

TRUNCATE TABLE rev.REV\_PROCESS\_QUEUE\_RESULT

TRUNCATE TABLE rev.REV\_TRACE\_SQL

TRUNCATE TABLE rev.REV\_PERSON\_PHOTO

TRUNCATE TABLE rev.REV\_EMAIL\_QUEUE

TRUNCATE TABLE rev.REV\_ERROR

TRUNCATE TABLE rev.REV\_WEB\_FARM\_SERVER

IF EXISTS

(

SELECT \*

FROM sys.foreign\_keys

WHERE object\_id = OBJECT\_ID(N'rev.REV\_AUDIT\_TRAIL\_PROP\_F1')

AND parent\_object\_id = OBJECT\_ID(N'rev.REV\_AUDIT\_TRAIL\_PROP\_HIS')

)

BEGIN

ALTER TABLE rev.REV\_AUDIT\_TRAIL\_PROP\_HIS DROP CONSTRAINT REV\_AUDIT\_TRAIL\_PROP\_F1

END

IF EXISTS

(

SELECT \* FROM sys.foreign\_keys

WHERE object\_id = OBJECT\_ID(N'rev.REV\_PROCESS\_QUEUE\_RESULT\_F1')

AND parent\_object\_id = OBJECT\_ID(N'rev.REV\_PROCESS\_QUEUE\_RESULT')

)

BEGIN

ALTER TABLE rev.REV\_PROCESS\_QUEUE\_RESULT DROP CONSTRAINT REV\_PROCESS\_QUEUE\_RESULT\_F1

END

IF EXISTS

(

SELECT \*

FROM sys.foreign\_keys

WHERE object\_id = OBJECT\_ID

(N'rev.REV\_TRACE\_SQL\_F1')

AND parent\_object\_id = OBJECT\_ID(N'rev.REV\_TRACE\_SQL')

)

BEGIN

ALTER TABLE rev.REV\_TRACE\_SQL DROP CONSTRAINT REV\_TRACE\_SQL\_F1

END

TRUNCATE TABLE rev.REV\_AUDIT\_TRAIL\_HIS

TRUNCATE TABLE rev.REV\_PROCESS\_QUEUE

TRUNCATE TABLE rev.REV\_TRACE

ALTER TABLE rev.REV\_AUDIT\_TRAIL\_PROP\_HIS

WITH CHECK ADD CONSTRAINT REV\_AUDIT\_TRAIL\_PROP\_F1

FOREIGN KEY(AUDIT\_TRAIL\_GU)

REFERENCES rev.REV\_AUDIT\_TRAIL\_HIS (AUDIT\_TRAIL\_GU)

ON UPDATE CASCADE

ON DELETE CASCADE

ALTER TABLE rev.REV\_PROCESS\_QUEUE\_RESULT

WITH CHECK ADD CONSTRAINT REV\_PROCESS\_QUEUE\_RESULT\_F1 FOREIGN KEY(PROCESS\_QUEUE\_GU)

REFERENCES rev.REV\_PROCESS\_QUEUE (PROCESS\_QUEUE\_GU)

ON UPDATE CASCADE

ON DELETE CASCADE

ALTER TABLE rev.REV\_TRACE\_SQL

WITH CHECK ADD CONSTRAINT REV\_TRACE\_SQL\_F1 FOREIGN KEY(TRACE\_GU)

REFERENCES rev.REV\_TRACE (TRACE\_GU)

ON UPDATE CASCADE

ON DELETE CASCADE

ALTER TABLE rev.REV\_AUDIT\_TRAIL\_PROP\_HIS

CHECK CONSTRAINT REV\_AUDIT\_TRAIL\_PROP\_F1

ALTER TABLE rev.REV\_PROCESS\_QUEUE\_RESULT

CHECK CONSTRAINT REV\_PROCESS\_QUEUE\_RESULT\_F1

ALTER TABLE rev.REV\_TRACE\_SQL

CHECK CONSTRAINT REV\_TRACE\_SQL\_F1

SET @RvTrcSQLAlter = @@ROWCOUNT

UPDATE rev.EPC\_PARENT\_PXP

SET EMAIL\_1 = 'email@edupoint.com',

EMAIL\_2 = 'email@edupoint.com',

EMAIL\_3 = 'email@edupoint.com',

EMAIL\_4 = 'email@edupoint.com',

EMAIL\_5 = 'email@edupoint.com'

SET @PrntPXPEmailUpdated = @@ROWCOUNT

UPDATE rev.REV\_PERSON

SET SOCIAL\_SECURITY\_NUMBER = '123456789'

WHERE SOCIAL\_SECURITY\_NUMBER is not null

SET @RPSSANUpdated = @@ROWCOUNT

UPDATE rev.REV\_PERSON\_PHONE

SET PHONE = SUBSTRING(PHONE,1,3) + '5551234'

WHERE LEN(PHONE) = 10

SET @RPPhnUpdated = @RPPhnUpdated + @@ROWCOUNT

UPDATE rev.REV\_PERSON\_PHONE

SET PHONE = '5551234'

WHERE LEN(PHONE) <> 10

SET @RPPhnUpdated = @RPPhnUpdated + @@ROWCOUNT

UPDATE rev.REV\_PERSON

SET PRIMARY\_PHONE = SUBSTRING(PRIMARY\_PHONE,1,3) + '5551234'

WHERE LEN(PRIMARY\_PHONE) = 10

SET @RPPriPhnUpdated = @RPPriPhnUpdated + @@ROWCOUNT

UPDATE rev.REV\_PERSON

SET PRIMARY\_PHONE = '5551234'

WHERE LEN(PRIMARY\_PHONE) <> 10

SET @RPPriPhnUpdated = @RPPriPhnUpdated + @@ROWCOUNT

UPDATE rev.REV\_PERSON

SET EMAIL = 'email@edupoint.com'

SET @RPEmailUpdated = @@ROWCOUNT

UPDATE rev.REV\_EMAIL

SET TO\_ADDRESS = 'email@edupoint.com',

FROM\_ADDRESS = 'email@edupoint.com'

SET @REmailUpdated = @@ROWCOUNT

-- handle all other states of @DebugOn with a rollback..... preventleaving the transaction open

IF @DebugOn = 0 COMMIT

ELSE ROLLBACK

PRINT ''

PRINT 'ParentPXP Updated: ' + CAST(@PrntPXPEmailUpdated AS VARCHAR(6)) + '.'

PRINT 'SSAN Updated: ' + CAST(@RPSSANUpdated AS VARCHAR(6)) + '.'

PRINT 'Phone # Updated: ' + CAST(@RPPhnUpdated AS VARCHAR(6)) + '.'

PRINT 'Primary Phone # Updated: ' + CAST(@RPPriPhnUpdated AS VARCHAR(6)) + '.'

PRINT 'Person Email Updated: ' + CAST(@RPEmailUpdated AS VARCHAR(6)) + '.'

PRINT 'RevEmail Updated: ' + CAST(@REmailUpdated AS VARCHAR(6)) + '.'

END

ELSE

PRINT 'This was a Production Database so the DBCleanse was skipped ' + @dbname