













Task List

PF LOAN ENTRY INTERFACE

----------------------------------------------------------------------

1. Select Loan code

Fetch Loan Details data

----------------------------------------------------------------------

2. Select PF No or Token No

Fetch Employee Details

----------------------------------------------------------------------

3. Put Loan Date

After Loan Entered do the following task....

i. Check Applied Loan Date already entered on this date

ii. Load Component Share Grid

iii. Fetch Previous Loan Balance

iV.

----------------------------------------------------------------------

4. Put Claim Amount

After Loan Claim Amount given

i. Check Sanction Amount

----------------------------------------------------------------------

5. Put Loan Sanction Year

After putting Sanction Year

i. Change No. of Installment

(WPS ---> 1 year = 25 installments)

(PIS ---> 1 year = 12 installments)

Calculation EMI amount

----------------------------------------------------------------------

----------------------------------------------------------------------

----------------------------------------------------------------------

CREATE OR REPLACE PROCEDURE GLOSTER\_WEB.PRC\_PF\_COMPONENT\_SHARE

(

p\_companycode varchar2,

p\_divisioncode varchar2,

p\_yearcode varchar2,

p\_workerserial varchar2,

p\_loancode varchar2,

p\_loandate varchar2,

OPTMODE VARCHAR2

)

as

lv\_output VARCHAR2(4000);

lv\_query VARCHAR2(4000);

lv\_query2 VARCHAR2(4000);

lv\_PFNO VARCHAR2(10);

lv\_FUNCTION VARCHAR2(4000);

lv\_TOKENNO VARCHAR2(10);

lv\_FNSTDT VARCHAR2(11);

lv\_FNENDT VARCHAR(11);

lv\_EMP\_COMCODE VARCHAR2(10);

lv\_EMP\_DIVCODE VARCHAR2(10);

BEGIN

SELECT TOKENNO,PFNO,CURRENTCOMPANYCODE, CURRENTDIVISIONCODE

INTO lv\_TOKENNO,lv\_PFNO,lv\_EMP\_COMCODE,lv\_EMP\_DIVCODE

FROM PFEMPLOYEEMASTER

WHERE WORKERSERIAL=p\_workerserial;

DELETE FROM GTT\_LOANSANCTION\_PRAM;

SELECT FN\_GETFORTNIGHTSTARTENDDATE(p\_loandate,'START') INTO lv\_FNSTDT FROM DUAL;

SELECT FN\_GETFORTNIGHTSTARTENDDATE(p\_loandate,'END') INTO lv\_FNENDT FROM DUAL;

PROC\_WPSVIEWCREATION ( lv\_EMP\_COMCODE,lv\_EMP\_DIVCODE,'MAST',0,lv\_FNSTDT,lv\_FNENDT);

PRC\_PISVIEWCREATION(lv\_EMP\_COMCODE,lv\_EMP\_DIVCODE,'PISASSIGN',0,TO\_CHAR(TO\_DATE(p\_loandate,'DD/MM/YYYY'),'YYYYMM'),TO\_CHAR(TO\_DATE(p\_loandate,'DD/MM/YYYY'),'YYYYMM'));

lv\_query:='INSERT INTO GTT\_LOANSANCTION\_PRAM'||chr(10)

||'(COMPANYCODE, DIVISIONCODE, YEARCODE,PFNO,TOKENNO , WORKERSERIAL, LOANDATE, LOANCODE,PARAM\_NAME,PF\_E,PF\_C , VPF,OTHER )'||chr(10)

||'SELECT '''||p\_companycode||''' COMPANYCODE,'''||p\_divisioncode||''' DIVISIONCODE ,'||CHR(10)

||''''||p\_yearcode||''' YEARCODE,'''||lv\_PFNO||''' PFNO,'''||lv\_TOKENNO||''' TOKENNO ,'||CHR(10)

||''''||p\_workerserial||''' WORKERSERIAL,TO\_DATE('''||p\_loandate||''',''DD/MM/YYYY'') LOANDATE,'''||p\_loancode||''' LOANCODE,PARAM\_NAME,'||CHR(10)

||'NULL PF\_E,NULL PF\_C ,NULL VPF,NULL OTHER '||CHR(10);

lv\_query:=lv\_query ||' FROM SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM'||CHR(10)

||'WHERE LOANCODE='''||p\_loancode||''''||CHR(10)

--||'GROUP BY PARAM\_NAME'||CHR(10)

||'order BY PIORITY\_INDEX';

--DBMS\_OUTPUT.PUT\_LINE(lv\_query);

execute IMMEDIATE lv\_query;

for c2 in(SELECT \* FROM SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM WHERE COMPANYCODE=p\_companycode /\*AND DIVISIONCODE=p\_divisioncode\*/ AND LOANCODE = p\_loancode order BY PIORITY\_INDEX /\*WHERE APPLICABLE\_COMPONENT='PF\_E'\*/)

loop

IF c2.HELP\_QRY IS NOT NULL THEN

lv\_query2:=REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(c2.HELP\_QRY,'<<COMPANYCODE>>',p\_companycode),'<<DIVISIONCODE>>',p\_divisioncode),'<<YEARCODE>>',p\_yearcode), '<<PFNO>>',lv\_PFNO),'<<LOANCODE>>',p\_loancode),'<<LOANDATE>>',p\_loandate);

DBMS\_OUTPUT.PUT\_LINE(lv\_query2);

IF c2.APPLICABLE\_COMPONENT='COMMON' THEN

EXECUTE IMMEDIATE lv\_query2 INTO lv\_output;

EXECUTE IMMEDIATE lv\_output;

elsif c2.APPLICABLE\_COMPONENT='COMMON2' THEN

EXECUTE IMMEDIATE lv\_query2;

ELSE

IF c2.HELP\_QRY IS NOT NULL THEN

EXECUTE IMMEDIATE lv\_query2 INTO lv\_output;

lv\_query:='UPDATE GTT\_LOANSANCTION\_PRAM SET '||c2.APPLICABLE\_COMPONENT||' = '''||lv\_output||''' WHERE PARAM\_NAME = '''||c2.PARAM\_NAME||'''';

EXECUTE IMMEDIATE lv\_query;

END IF;

END IF;

UPDATE GTT\_LOANSANCTION\_PRAM SET TABINDEX=C2.PIORITY\_INDEX WHERE PARAM\_NAME=c2.PARAM\_NAME;

ELSE

IF OPTMODE='A' THEN

lv\_FUNCTION:='SELECT '||REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(c2.PROC\_NAME,'<<COMPANYCODE>>',p\_companycode),'<<DIVISIONCODE>>',p\_divisioncode),'<<YEARCODE>>',p\_yearcode), '<<PFNO>>',lv\_PFNO),'<<LOANCODE>>',p\_loancode),'<<LOANDATE>>',p\_loandate)||' FROM DUAL';

DBMS\_OUTPUT.PUT\_LINE(lv\_FUNCTION);

EXECUTE IMMEDIATE lv\_FUNCTION INTO lv\_output;

IF lv\_output='Y' THEN

lv\_query:='UPDATE GTT\_LOANSANCTION\_PRAM SET ISELLIGIBLE = '''||lv\_output||''' WHERE PARAM\_NAME = '''||c2.PARAM\_NAME||'''';

EXECUTE IMMEDIATE lv\_query;

ELSE

lv\_query:='UPDATE GTT\_LOANSANCTION\_PRAM SET ISELLIGIBLE = ''N'',REMARKS='''||lv\_output||''' WHERE PARAM\_NAME = '''||c2.PARAM\_NAME||'''';

EXECUTE IMMEDIATE lv\_query;

RETURN;

END IF;

UPDATE GTT\_LOANSANCTION\_PRAM SET TABINDEX=C2.PIORITY\_INDEX WHERE PARAM\_NAME=c2.PARAM\_NAME;

END IF;

END IF;

end loop;

END;

/

CREATE OR REPLACE FUNCTION FN\_GET\_QRYSTRING

(

P\_PARAMS VARCHAR2,

P\_QUERY VARCHAR2

)

RETURN VARCHAR2

AS

LV\_RESULT VARCHAR2(4000);

BEGIN

LV\_RESULT := P\_QUERY;

--LV\_PARAMS := 'COMPANYCODE:0001~DIVISIONCODE:0002';

--LV\_QRY := 'SELECT <<COMPANYCODE>>,<<DIVISIONCODE>> FROM DUAL';

--SELECT FN\_GET\_QRYSTRING('COMPANYCODE:0001~DIVISIONCODE:0002','SELECT <<COMPANYCODE>>,<<DIVISIONCODE>> FROM DUAL') FROM DUAL

FOR C1 IN

(

SELECT REC1,SUBSTR(REC1,0,INSTR(REC1,':')-1) PARAMNAME,

SUBSTR(REC1,INSTR(REC1,':')+1, LENGTH(REC1)) PARAMVALUE

FROM

(

SELECT REGEXP\_SUBSTR(P\_PARAMS,'[^~]+', 1, LEVEL) REC1

FROM DUAL

CONNECT BY REGEXP\_SUBSTR(P\_PARAMS, '[^~]+', 1, LEVEL)

IS NOT NULL

)

)

LOOP

LV\_RESULT := REPLACE(LV\_RESULT,'<<'||C1.PARAMNAME||'>>',C1.PARAMVALUE);

END LOOP;

RETURN LV\_RESULT;

END;

/

CREATE OR REPLACE FUNCTION FN\_GET\_PARAMVALUE

(

P\_PARAMS VARCHAR2,

P\_PARAMNAME VARCHAR2

)

RETURN VARCHAR2

AS

LV\_RESULT VARCHAR2(4000);

BEGIN

LV\_RESULT := NULL;

--SELECT FN\_GET\_PARAMVALUE('COMPANYCODE:0001~DIVISIONCODE:0002','COMPANYCODE') FROM DUAL

SELECT PARAMVALUE INTO LV\_RESULT FROM (

SELECT REC1,SUBSTR(REC1,0,INSTR(REC1,':')-1) PARAMNAME,

SUBSTR(REC1,INSTR(REC1,':')+1, LENGTH(REC1)) PARAMVALUE

FROM

(

select regexp\_substr(P\_PARAMS,'[^~]+', 1, level) REC1

from dual

connect BY regexp\_substr(P\_PARAMS, '[^~]+', 1, level)

is not null

)

)

WHERE PARAMNAME=P\_PARAMNAME;

RETURN LV\_RESULT;

EXCEPTION WHEN OTHERS THEN

LV\_RESULT := NULL;

RETURN LV\_RESULT;

END;

/

DROP TABLE SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM CASCADE CONSTRAINTS;

CREATE TABLE SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(

COMPANYCODE VARCHAR2(10 BYTE),

DIVISIONCODE VARCHAR2(10 BYTE),

LOANCODE VARCHAR2(10 BYTE),

PARAM\_NAME VARCHAR2(100 BYTE),

PARAM\_DESC VARCHAR2(100 BYTE),

APPLICABLE\_COMPONENT VARCHAR2(10 BYTE),

PROC\_NAME VARCHAR2(200 BYTE),

HELP\_QRY VARCHAR2(4000 BYTE),

PIORITY\_INDEX NUMBER(3)

)

--INSERT SCRIPT

SET DEFINE OFF;

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'PF GROSS', 'PF GROSS',

'OTHER', NULL, 'SELECT FN\_PFGROSS\_FOR\_PFLOAN(''<<COMPANYCODE>>'',''<<DIVIISIONCODE>>'',''<<PFNO>>'',''<<LOANDATE>>'')\*12 FROM DUAL ', 3);

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'MINIMUM BALANCE', 'MINIMUM BALANCE',

'PF\_E', NULL, ' SELECT DISTINCT MINBALANCE'||CHR(10)||' FROM PFLOANMASTER'||CHR(10)||' WHERE COMPANYCODE = ''<<COMPANYCODE>>'''||CHR(10)||' --AND DIVISIONCODE = ''<<DIVISIONCODE>>'''||CHR(10)||' AND LOANCODE = ''<<LOANCODE>>'''||CHR(10)||' AND AGAINSTCOMPONENT=''PF\_E''', 5);

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'RATIO', 'RATIO',

'COMMON', NULL, 'select decode(rtrim(xmlagg(xmlelement(e,x.COMPOMENT||'' = ''||x.PERCENTAGE||'','') ).extract(''//text()''),'',''),'''',null, ''UPDATE GTT\_LOANSANCTION\_PRAM SET ''||rtrim(xmlagg(xmlelement(e,x.COMPOMENT||'' = ''||x.PERCENTAGE||'','') ).extract(''//text()''),'','')|| '' WHERE PARAM\_NAME = ''''RATIO'''' '')'||CHR(10)||'FROM '||CHR(10)||'( select PF\_E\_PERCENTAGE PERCENTAGE,''PF\_E'' COMPOMENT from PFLOANCRITERIA where COMPANAYCODE=''<<COMPANYCODE>>'' AND DIVISIONCODE=''<<DIVISIONCODE>>'' AND LOANCODE=''<<LOANCODE>>'''||CHR(10)||' union'||CHR(10)||' select PF\_C\_PERCENTAGE PERCENTAGE,''PF\_C'' COMPOMENT from PFLOANCRITERIA where COMPANAYCODE=''<<COMPANYCODE>>'' AND DIVISIONCODE=''<<DIVISIONCODE>>'' AND LOANCODE=''<<LOANCODE>>'''||CHR(10)||' union'||CHR(10)||' select VPF\_PERCENTAGE PERCENTAGE,''VPF'' COMPOMENT from PFLOANCRITERIA where COMPANAYCODE=''<<COMPANYCODE>>'' AND DIVISIONCODE=''<<DIVISIONCODE>>'' AND LOANCODE=''<<LOANCODE>>'''||CHR(10)||') X ', 6);

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'ACTUAL BALANCE', 'ACTUAL BALANCE',

'COMMON', NULL, 'select decode(rtrim(xmlagg(xmlelement(e,x.componentcode||'' = ''||x.COMPONENTAMOUNT||'','') ).extract(''//text()''),'',''),'''',null, ''UPDATE GTT\_LOANSANCTION\_PRAM SET ''||rtrim(xmlagg(xmlelement(e,x.componentcode||'' = ''||x.COMPONENTAMOUNT||'','') ).extract(''//text()''),'','')|| '' WHERE PARAM\_NAME = ''''ACTUAL BALANCE'''' '')'||CHR(10)||'FROM '||CHR(10)||'('||CHR(10)||' SELECT COMPONENTCODE, SUM(COMPONENTAMOUNT) COMPONENTAMOUNT '||CHR(10)||' FROM'||CHR(10)||' ('||CHR(10)||' SELECT COMPONENTCODE, COMPONENTAMOUNT FROM PFTRANSACTIONDETAILS'||CHR(10)||' WHERE 1=1'||CHR(13)||CHR(10)||' --COMPANYCODE=''<<COMPANYCODE>>'''||CHR(10)||' --AND DIVISIONCODE=''<<DIVISIONCODE>>'' '||CHR(10)||' AND YEARCODE=''<<YEARCODE>>'''||CHR(10)||' AND YEARMONTH <= TO\_char(to\_date(''<<LOANDATE>>'',''DD/MM/YYYY''),''YYYYMM'') '||CHR(10)||' AND PFNO=''<<PFNO>>'''||CHR(10)||' AND COMPONENTCODE<>''EPF'''||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''PF\_E'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL'||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''PF\_C'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL'||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''VPF'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL'||CHR(10)||' )'||CHR(10)||' GROUP BY COMPONENTCODE '||CHR(10)||') X ', 4);

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'OPENING BALANCE', 'OPENING BALANCE',

'COMMON', NULL, 'select decode(rtrim(xmlagg(xmlelement(e,x.componentcode||'' = ''||x.COMPONENTAMOUNT||'','') ).extract(''//text()''),'',''),'''',null, ''UPDATE GTT\_LOANSANCTION\_PRAM SET ''||rtrim(xmlagg(xmlelement(e,x.componentcode||'' = ''||x.COMPONENTAMOUNT||'','') ).extract(''//text()''),'','')|| '' WHERE PARAM\_NAME = ''''OPENING BALANCE'''' '')'||CHR(10)||'FROM '||CHR(10)||'('||CHR(10)||' SELECT COMPONENTCODE, SUM(COMPONENTAMOUNT) COMPONENTAMOUNT'||CHR(10)||' FROM'||CHR(10)||' ( '||CHR(10)||' SELECT COMPONENTCODE, COMPONENTAMOUNT '||CHR(10)||' FROM PFTRANSACTIONDETAILS '||CHR(10)||' WHERE 1=1'||CHR(13)||CHR(10)||' --COMPANYCODE=''<<COMPANYCODE>>'' '||CHR(10)||' --AND DIVISIONCODE=''<<DIVISIONCODE>>'' '||CHR(10)||' AND YEARCODE=''<<YEARCODE>>'' '||CHR(10)||' AND TRANSACTIONTYPE=''PF INTEREST OPENING BALANCE'''||CHR(10)||' AND PFNO=''<<PFNO>>'''||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''PF\_E'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL'||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''PF\_C'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL'||CHR(10)||' UNION ALL'||CHR(10)||' SELECT ''VPF'' COMPONENTCODE, 0 COMPONENTAMOUNT FROM DUAL '||CHR(10)||' )'||CHR(10)||' GROUP BY COMPONENTCODE '||CHR(10)||') X ', 2);

Insert into SYS\_SWT\_PF\_LOAN\_SANCTION\_PARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, PARAM\_NAME, PARAM\_DESC,

APPLICABLE\_COMPONENT, PROC\_NAME, HELP\_QRY, PIORITY\_INDEX)

Values

('0002', '0010', 'PFL', 'LOAN APPLICABLE', 'LOAN APPLICABLE',

'COMMON', 'FN\_ELIGIBLEFORPFL(''<<COMPANYCODE>>'',''<<DIVISIONCODE>>'',''<<PFNO>>'',''<<LOANCODE>>'',''<<LOANDATE>>'')', NULL, 1);

COMMIT;

DROP TABLE GTT\_LOANSANCTION\_PRAM CASCADE CONSTRAINTS;

CREATE GLOBAL TEMPORARY TABLE GTT\_LOANSANCTION\_PRAM

(

COMPANYCODE VARCHAR2(10 BYTE),

DIVISIONCODE VARCHAR2(10 BYTE),

YEARCODE VARCHAR2(20 BYTE),

PFNO VARCHAR2(10 BYTE),

TOKENNO VARCHAR2(10 BYTE),

WORKERSERIAL VARCHAR2(10 BYTE),

LOANDATE DATE,

LOANCODE VARCHAR2(10 BYTE),

PARAM\_NAME VARCHAR2(100 BYTE),

PF\_E NUMBER(18,2),

PF\_C NUMBER(18,2),

VPF NUMBER(18,2),

OTHER NUMBER(18,2),

ISELLIGIBLE CHAR(1 BYTE),

REMARKS VARCHAR2(500 BYTE),

TABINDEX VARCHAR2(2 BYTE)

)

ON COMMIT PRESERVE ROWS

NOCACHE;

CREATE OR REPLACE PROCEDURE PROC\_PFLOANSANCTION

(

P\_COMPANYCODE VARCHAR2,

P\_DIVISIONCODE VARCHAR2,

P\_PARAMS VARCHAR2,

P\_PARAMTAG VARCHAR2 DEFAULT NULL

)

AS

LV\_FIRST VARCHAR2(50);

LV\_SQLSTR VARCHAR2(20000);

LV\_COMPANY VARCHAR2(150);

LV\_DIVISION VARCHAR2(150);

LV\_RETMSG VARCHAR2(1000);

LV\_LOANCODE VARCHAR2(10);

LV\_LOANDATE VARCHAR2(10);

LV\_PFNO VARCHAR2(30);

LV\_LOANAMOUNT NUMBER(10,2);

LV\_QRYVAL VARCHAR2(100);

BEGIN

LV\_RETMSG := '#SUCCESS';

LV\_SQLSTR:=NULL;

LV\_LOANCODE := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANCODE');

LV\_PFNO := FN\_GET\_PARAMVALUE(P\_PARAMS, 'PFNO');

LV\_LOANDATE := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANDATE');

LV\_LOANAMOUNT := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANAMOUNT');

-- DBMS\_OUTPUT.PUT\_LINE(LV\_LOANCODE);

-- DBMS\_OUTPUT.PUT\_LINE(LV\_PFNO);

-- DBMS\_OUTPUT.PUT\_LINE(LV\_LOANDATE);

FOR C1 IN (

SELECT \* FROM PFLOANSANCTIONPARAM

WHERE COMPANYCODE=P\_COMPANYCODE

AND DIVISIONCODE=P\_DIVISIONCODE

AND LOANCODE=LV\_LOANCODE

ORDER BY QUERYINDEX

)

LOOP

IF C1.PROC\_NAME IS NOT NULL THEN

LV\_SQLSTR := 'BEGIN ' || FN\_GET\_QRYSTRING(P\_PARAMS, C1.PROC\_NAME) ||' END;';

DBMS\_OUTPUT.PUT\_LINE(LV\_SQLSTR);

EXECUTE IMMEDIATE LV\_SQLSTR;

END IF;

IF C1.QUERYSTRING IS NOT NULL THEN

LV\_SQLSTR := FN\_GET\_QRYSTRING(P\_PARAMS, C1.QUERYSTRING);

DBMS\_OUTPUT.PUT\_LINE(LV\_SQLSTR);

EXECUTE IMMEDIATE LV\_SQLSTR INTO LV\_QRYVAL;

DBMS\_OUTPUT.PUT\_LINE(LV\_QRYVAL);

EXIT WHEN LV\_QRYVAL <> '#SUCCESS';

END IF;

END LOOP;

-- INSERT INTO SYS\_GBL\_PROCOUTPUT\_INFO(SYS\_SAVE\_INFO) VALUES(LV\_QRYVAL);

INSERT INTO SYS\_GBL\_PROCOUTPUT\_INFO SELECT LV\_QRYVAL FROM DUAL;

-- dbms\_output.put\_line(lv\_sqlstr);

-- EXECUTE IMMEDIATE LV\_SQLSTR;

EXCEPTION WHEN OTHERS THEN NULL;

END;

/

CREATE OR REPLACE PROCEDURE PRCWPS\_PFLOAN\_AFTERSAVE

IS

LV\_CNT NUMBER;

LV\_RESULT VARCHAR2(10);

LV\_ERROR\_REMARK VARCHAR2(4000) := '' ;

LV\_MASTER GBL\_PFLOANAPPLICATION%ROWTYPE;

LV\_PF\_C NUMBER;

LV\_PF\_E NUMBER;

LV\_VPF NUMBER;

LV\_FORTNIGHTSTARTDATE VARCHAR2(10);

LV\_FORTNIGHTENDDATE VARCHAR2(10);

LV\_LOANADJTYPE VARCHAR2(1);

BEGIN

lv\_result:='#SUCCESS#';

select \*

into lv\_master

from gbl\_pfloanapplication

where rownum<=1;

select count(\*)

into lv\_cnt

from gbl\_pfloanapplication;

if nvl(lv\_cnt,0)=0 then

lv\_error\_remark := 'Validation Failure : [Blank data not allowded to save!]';

raise\_application\_error(to\_number(fn\_display\_error( 'COMMON')),fn\_display\_error( 'COMMON',5,lv\_error\_remark));

end if;

if lv\_master.operationmode is null then

lv\_error\_remark := 'Validation Failure : [Which kind of Activity you want to accomplish ADD / EDIT ? ]';

raise\_application\_error(to\_number(fn\_display\_error( 'COMMON')),fn\_display\_error( 'COMMON',6,lv\_error\_remark));

end if;

INSERT INTO PFLOANTRANSACTION

SELECT \*

FROM PFLOANAPPLICATION

WHERE PFNO = LV\_MASTER.PFNO

AND LOANCODE = LV\_MASTER.LOANCODE

AND LOANDATE = LV\_MASTER.LOANDATE;

SELECT LOANADJTYPE INTO LV\_LOANADJTYPE

FROM PFLOANMASTER

WHERE COMPANYCODE = LV\_MASTER.COMPANYCODE

AND DIVISIONCODE = LV\_MASTER.DIVISIONCODE

AND LOANCODE = LV\_MASTER.LOANCODE;

IF LV\_LOANADJTYPE = 'D' THEN

INSERT INTO PFLOANBREAKUP

(

COMPANYCODE, DIVISIONCODE, MODULE, YEARCODE, YEARMONTH, CATEGORYCODE, GRADECODE, WORKERSERIAL, TOKENNO,PFNO,

LOANCODE, LOANDATE, EFFECTYEARMONTH, INTERESTPERCENTAGE, AMOUNT, REPAYAMOUNT, REPAYCAPITAL, REPAYINTEREST,REMARKS,

TRANSACTIONTYPE, EFFECTFORTNIGHT, PAIDON, DEDUCTEDAMT, SYSROWID, USERNAME

)

SELECT COMPANYCODE, DIVISIONCODE, MODULE, YEARCODE, YEARMONTH, CATEGORYCODE, GRADECODE, WORKERSERIAL, TOKENNO,PFNO,

AGAINSTLOANCODE, AGAINSTLOANDATE, YEARMONTH, INTERESTPERCENTAGE, LOANAMOUNTADJUSTED + LOANINTAMOUNTADJUSTED,

LOANAMOUNTADJUSTED + LOANINTAMOUNTADJUSTED, LOANAMOUNTADJUSTED, LOANINTAMOUNTADJUSTED,

'ADJUSTED WITH THE LOAN CODE '|| LOANCODE || ' DATED ' || TO\_DATE(LOANDATE,'DD/MM/YYYY'),

'REPAY', DEDUCTIONSTARTDATE, LOANDATE, 0, SYSROWID, USERNAME

FROM PFLOANAPPLICATION

WHERE PFNO = LV\_MASTER.PFNO

AND LOANCODE = LV\_MASTER.LOANCODE

AND LOANDATE = LV\_MASTER.LOANDATE;

END IF;

if nvl(lv\_master.loantype,'XX')='NONREFUNDABLE' then

if nvl(lv\_master.module,'XX')='WPS' then

select to\_char(fortnightstartdate,'DD/MM/RRRR'),to\_char(fortnightenddate,'DD/MM/RRRR')

into lv\_fortnightstartdate,lv\_fortnightenddate

from wpswagedperioddeclaration

where companycode=lv\_master.companycode and divisioncode=lv\_master.divisioncode and to\_date(lv\_master.loandate,'DD/MM/RRRR') between fortnightstartdate and fortnightenddate;

else

select '01/'||substr(lv\_master.loandate,4),TO\_CHAR(last\_day(to\_date(lv\_master.loandate,'DD/MM/RRRR')),'DD/MM/YYYY')

into lv\_fortnightstartdate,lv\_fortnightenddate

from dual;

end if;

SELECT PF\_E,PF\_C,VPF INTO LV\_PF\_E ,LV\_PF\_C ,LV\_VPF

FROM GBL\_LOANSANCTION\_DETAILS

WHERE COMPANYCODE = lv\_master.companycode

AND DIVISIONCODE = lv\_master.divisioncode

AND PFNO = LV\_MASTER.PFNO

AND LOANCODE = LV\_MASTER.LOANCODE

AND LOANDATE = LV\_MASTER.LOANDATE

AND PARAM\_NAME = 'SANCTION AMOUNT';

delete from pftransactiondetails

where companycode=lv\_master.companycode

and divisioncode=lv\_master.divisioncode

and yearmonth=lv\_master.yearmonth

and startdate=to\_date(lv\_fortnightstartdate,'dd/mm/yyyy')

and enddate=to\_date(lv\_fortnightenddate,'dd/mm/yyyy')

and workerserial=lv\_master.workerserial

and pfno=lv\_master.pfno

and transactiontype='LOAN';

if LV\_PF\_E >0 then

insert into pftransactiondetails

select companycode, divisioncode, yearcode, yearmonth, to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') startdate,

to\_date(lv\_fortnightenddate,'dd/mm/yyyy') enddate, pfno, workerserial, tokenno,

'PF\_E' componentcode, -1\*pf\_e componentamount, 'LOAN' transactiontype, lv\_master.module postedfrom,

'WORKER' employmenttype, 'LESS' addless, null pftrustcode,

0 int\_per, 0 int\_amt, pf\_e total\_amt, null pensionno, 0 averagebalance, employeecompanycode, employeedivisioncode,

module, 0 adj\_amt, remarks, voucherno, voucherdate, systemvoucherno, systemvoucherdate,

loandate applicationdate, applicationno,SYSDATE LASTMODIFIEDDATE,username,SYSROWID

from pfloantransaction

where companycode=lv\_master.companycode

and divisioncode=lv\_master.divisioncode

and yearmonth=lv\_master.yearmonth

and loandate between to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') and to\_date(lv\_fortnightenddate,'dd/mm/yyyy')

and workerserial=lv\_master.workerserial

and pfno=lv\_master.pfno;

end if;

if LV\_PF\_C >0 then

insert into pftransactiondetails

select companycode, divisioncode, yearcode, yearmonth, to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') startdate,

to\_date(lv\_fortnightenddate,'dd/mm/yyyy') enddate, pfno, workerserial, tokenno,

'PF\_C' componentcode, -1\*pf\_c componentamount, 'LOAN' transactiontype, lv\_master.module postedfrom,

'WORKER' employmenttype, 'LESS' addless, null pftrustcode,

0 int\_per, 0 int\_amt, pf\_c total\_amt, null pensionno, 0 averagebalance, employeecompanycode, employeedivisioncode,

module, 0 adj\_amt, remarks, voucherno, voucherdate, systemvoucherno, systemvoucherdate,

loandate applicationdate, applicationno,SYSDATE LASTMODIFIEDDATE,username,SYSROWID

from pfloantransaction

where companycode=lv\_master.companycode

and divisioncode=lv\_master.divisioncode

and yearmonth=lv\_master.yearmonth

and loandate between to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') and to\_date(lv\_fortnightenddate,'dd/mm/yyyy')

and workerserial=lv\_master.workerserial

and pfno=lv\_master.pfno;

end if;

if LV\_VPF>0 then

insert into pftransactiondetails

select companycode, divisioncode, yearcode, yearmonth, to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') startdate,

to\_date(lv\_fortnightenddate,'dd/mm/yyyy') enddate, pfno, workerserial, tokenno,

'VPF' componentcode, -1\*vpf componentamount, 'LOAN' transactiontype, lv\_master.module postedfrom,

'WORKER' employmenttype, 'LESS' addless, null pftrustcode,

0 int\_per, 0 int\_amt, vpf total\_amt, null pensionno, 0 averagebalance, employeecompanycode, employeedivisioncode,

module, 0 adj\_amt, remarks, voucherno, voucherdate, systemvoucherno, systemvoucherdate,

loandate applicationdate, applicationno,SYSDATE LASTMODIFIEDDATE,username,SYSROWID

from pfloantransaction

where companycode=lv\_master.companycode

and divisioncode=lv\_master.divisioncode

and yearmonth=lv\_master.yearmonth

and loandate between to\_date(lv\_fortnightstartdate,'dd/mm/yyyy') and to\_date(lv\_fortnightenddate,'dd/mm/yyyy')

and workerserial=lv\_master.workerserial

and pfno=lv\_master.pfno;

end if;

ELSE

IF lv\_Master.OPERATIONMODE <>'D' THEN

FOR C1 IN (SELECT PFNO,LOANCODE,LOANDATE FROM GBL\_PFLOANAPPLICATION)

LOOP

DELETE FROM PFLOANINTEREST

WHERE PFNO = C1.PFNO

AND LOANCODE = C1.LOANCODE

AND YEARMONTH=TO\_CHAR(C1.LOANDATE,'YYYYMM');

END LOOP;

INSERT INTO PFLOANINTEREST

(COMPANYCODE, DIVISIONCODE, YEARCODE, YEARMONTH, FORTNIGHTSTARTDATE, FORTNIGHTENDDATE,LOANCODE, WORKERSERIAL, TOKENNO, PFNO,

LOANDATE, LOANAMOUNT, INTERESTAPPLICABLEON, INTERESTPERCENTAGE, INTERESTAMOUNT, MODULE, TRANSACTIONTYPE, REMARKS,

USERNAME, SYSROWID

)

SELECT COMPANYCODE, DIVISIONCODE, YEARCODE, YEARMONTH, TO\_DATE(LV\_FORTNIGHTSTARTDATE,'DD/MM/RRRR'), TO\_DATE(LV\_FORTNIGHTENDDATE,'DD/MM/RRRR'), LOANCODE, WORKERSERIAL, TOKENNO, PFNO,

LOANDATE, ACTUALLOANAMOUNT, ACTUALLOANAMOUNT, INTERESTPERCENTAGE, INTERESTAMOUNT, MODULE,'ADD',NULL,

USERNAME, 1 SYSROWID FROM GBL\_PFLOANAPPLICATION;

--ADDED ON 29/07/2020

IF LV\_MASTER.LOANINTAMOUNTADJUSTED > 0 AND LV\_LOANADJTYPE = 'D' THEN

INSERT INTO PFLOANINTEREST

(COMPANYCODE, DIVISIONCODE, YEARCODE, YEARMONTH, FORTNIGHTSTARTDATE, FORTNIGHTENDDATE,LOANCODE, WORKERSERIAL, TOKENNO, PFNO,

LOANDATE, LOANAMOUNT, INTERESTAPPLICABLEON, INTERESTPERCENTAGE, INTERESTAMOUNT, MODULE, TRANSACTIONTYPE, REMARKS,

USERNAME, SYSROWID

)

SELECT COMPANYCODE, DIVISIONCODE, YEARCODE, YEARMONTH, TO\_DATE(LV\_FORTNIGHTSTARTDATE,'DD/MM/RRRR'), TO\_DATE(LV\_FORTNIGHTENDDATE,'DD/MM/RRRR'), AGAINSTLOANCODE, WORKERSERIAL, TOKENNO, PFNO,

AGAINSTLOANDATE, ACTUALLOANAMOUNT, LOANAMOUNTADJUSTED, INTERESTPERCENTAGE, LOANINTAMOUNTADJUSTED, MODULE,'ADD',NULL,

USERNAME, 1 SYSROWID FROM GBL\_PFLOANAPPLICATION;

END IF;

--ENDED ON 29/07/2020

END IF;

end if;

end;

/

DROP TABLE PFLOANSANCTIONPARAM CASCADE CONSTRAINTS;

CREATE TABLE PFLOANSANCTIONPARAM

(

COMPANYCODE VARCHAR2(10 BYTE),

DIVISIONCODE VARCHAR2(10 BYTE),

LOANCODE VARCHAR2(10 BYTE),

QUERYINDEX NUMBER(5,2),

PARAMCODE VARCHAR2(30 BYTE),

PARAMNAME VARCHAR2(100 BYTE),

PARAMVALUE NUMBER(11,2),

PARAMVALUEFROM NUMBER(11,2),

PARAMVALUETO NUMBER(11,2),

QUERYSTRING VARCHAR2(3000 BYTE),

PROC\_NAME VARCHAR2(50 BYTE),

USERNAME VARCHAR2(50 BYTE),

SYSROWID VARCHAR2(50 BYTE),

LASTMODIFIED DATE,

ERROR\_MSG VARCHAR2(1000 BYTE)

)

--INSERT SCRIPT--

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 1, 'MAXLIMIT',

'MAXIMUM LIMIT', 100000, NULL, NULL, 'SELECT CASE WHEN <<LOANCLAIMAMOUNT>> < PARAMVALUE '||CHR(13)||CHR(10)||'THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''LOAN AMOUNT <<LOANCLAIMAMOUNT>> CAN NOT BE GREATER THAN MAXIMUM AMOUNT ''||PARAMVALUE '||CHR(13)||CHR(10)||'END CHK FROM PFLOANSANCTIONPARAM'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''MAXLIMIT'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 2, 'PREVLOAN\_PERCENT',

'PREVIOUS LOAN OUTSTANDING', 50, NULL, NULL, 'SELECT CASE WHEN (100 - (<<LOANAMOUNTADJUSTED>>/B.AMOUNT)\*100) > PARAMVALUE THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''PREVIOUS LOAN OUTSTANDING BALANCE (''||ROUND(((<<LOANAMOUNTADJUSTED>>/B.AMOUNT)\*100),2)||''%) MUST BE LESS THAN ''||PARAMVALUE||''%'' END CHK'||CHR(13)||CHR(10)||'FROM PFLOANSANCTIONPARAM A,'||CHR(13)||CHR(10)||'('||CHR(13)||CHR(10)||' SELECT AMOUNT FROM PFLOANTRANSACTION'||CHR(13)||CHR(10)||' WHERE PFNO=''<<PFNO>>'''||CHR(13)||CHR(10)||' AND LOANCODE=''<<LOANCODE>>'''||CHR(13)||CHR(10)||' AND LOANDATE=TO\_DATE(''<<AGAINSTLOANDATE>>'',''DD/MM/YYYY'')'||CHR(13)||CHR(10)||' AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||' AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||') B'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''PREVLOAN\_PERCENT'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 3, 'ELIGIBLE\_YEARS',

'ELIGIBLE YEARS', 0, 2, 50, 'SELECT CASE WHEN B.SERVICE\_YEAR BETWEEN PARAMVALUEFROM AND PARAMVALUETO THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''SERVICE YEAR MUST BE BETWEEN ''||PARAMVALUEFROM|| '' TO ''||PARAMVALUETO||'' YEARS'' END '||CHR(13)||CHR(10)||'FROM PFLOANSANCTIONPARAM A,'||CHR(13)||CHR(10)||'('||CHR(13)||CHR(10)||' SELECT (TO\_NUMBER(SUBSTR(''<<LOANDATE>>'',-4)) -TO\_NUMBER(TO\_CHAR(PFJOINDATE,''YYYY''))) SERVICE\_YEAR'||CHR(13)||CHR(10)||' FROM PFEMPLOYEEMASTER'||CHR(13)||CHR(10)||' WHERE PFNO = ''<<PFNO>>'''||CHR(13)||CHR(10)||') B'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''ELIGIBLE\_YEARS'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 14, 'MINIMUMPFBALANCE',

'MINIMUM PF BALANCE', 1000, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'MPL', 15, 'LOANTAKEN\_NOOFTIMES',

'LOANTAKEN NOOFTIMES', 5, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 3, 'VPF\_PERCENTAGE',

'VPF PERCENTAGE', 0, 0, 0, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 4, 'PF\_C\_PERCENTAGE',

'PF\_C PERCENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 5, 'PF\_E\_PERCENTAGE',

'PF\_E PERCENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 6, 'PREV\_LOAN\_DUE\_PERCENTAGE',

'PREVIOUS LOAN DUE PERCENTAGE', 50, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 7, 'MAXIMUMLOANAMOUNT',

'MAXIMUMLOANAMOUNT', 30000, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 8, 'MINIMUMLOANAMOUNT',

'MINIMUM LOAN AMOUNT', 0, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 9, 'ROUNDOFFLOANAMOUNT',

'ROUND OFF LOAN AMOUNT', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'MPL', 10, 'PFGROSSMONTH',

'PF GROSS MONTHLY', 0, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 11, 'ELIGIBLEPECENTAGE',

'ELIGIBLE PECENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'MPL', 12, 'ELIGIBLEAGE',

'ELIGIBLE AGE', 45, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 3, 'VPF\_PERCENTAGE',

'VPF PERCENTAGE', 0, 0, 0, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 4, 'PF\_C\_PERCENTAGE',

'PF\_C PERCENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 5, 'PF\_E\_PERCENTAGE',

'PF\_E PERCENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 6, 'PREV\_LOAN\_DUE\_PERCENTAGE',

'PREVIOUS LOAN DUE PERCENTAGE', 50, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 7, 'MAXIMUMLOANAMOUNT',

'MAXIMUMLOANAMOUNT', 30000, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 8, 'MINIMUMLOANAMOUNT',

'MINIMUM LOAN AMOUNT', 0, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 9, 'ROUNDOFFLOANAMOUNT',

'ROUND OFF LOAN AMOUNT', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'PFL', 10, 'PFGROSSMONTH',

'PF GROSS MONTHLY', 0, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 11, 'ELIGIBLEPECENTAGE',

'ELIGIBLE PECENTAGE', 100, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'PFL', 12, 'ELIGIBLEAGE',

'ELIGIBLE AGE', 45, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'PFL', 14, 'MINIMUMPFBALANCE',

'MINIMUM PF BALANCE', 1000, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010XXX', 'PFL', 15, 'LOANTAKEN\_NOOFTIMES',

'LOANTAKEN NOOFTIMES', 5, NULL, NULL, NULL,

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 1, 'MAXLIMIT',

'MAXIMUM LIMIT', 30000, NULL, NULL, 'SELECT CASE WHEN <<LOANCLAIMAMOUNT>> < PARAMVALUE '||CHR(13)||CHR(10)||'THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''LOAN AMOUNT <<LOANCLAIMAMOUNT>> CAN NOT BE GREATER THAN MAXIMUM AMOUNT ''||PARAMVALUE '||CHR(13)||CHR(10)||'END CHK FROM PFLOANSANCTIONPARAM'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''MAXLIMIT'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010', 'MPL', 2, 'PREVLOAN\_PERCENT',

'PREVIOUS LOAN OUTSTANDING', 100, NULL, NULL, 'SELECT CASE WHEN (100 - (<<LOANAMOUNTADJUSTED>>/B.AMOUNT)\*100) > PARAMVALUE THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''PREVIOUS LOAN OUTSTANDING BALANCE (''||ROUND(((<<LOANAMOUNTADJUSTED>>/B.AMOUNT)\*100),2)||''%) MUST BE LESS THAN ''||PARAMVALUE||''%'' END CHK'||CHR(13)||CHR(10)||'FROM PFLOANSANCTIONPARAM A,'||CHR(13)||CHR(10)||'('||CHR(13)||CHR(10)||' SELECT AMOUNT FROM PFLOANTRANSACTION'||CHR(13)||CHR(10)||' WHERE PFNO=''<<PFNO>>'''||CHR(13)||CHR(10)||' AND LOANCODE=''<<LOANCODE>>'''||CHR(13)||CHR(10)||' AND LOANDATE=TO\_DATE(''<<AGAINSTLOANDATE>>'',''DD/MM/YYYY'')'||CHR(13)||CHR(10)||' AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||' AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||') B'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''PREVLOAN\_PERCENT'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

Insert into PFLOANSANCTIONPARAM

(COMPANYCODE, DIVISIONCODE, LOANCODE, QUERYINDEX, PARAMCODE,

PARAMNAME, PARAMVALUE, PARAMVALUEFROM, PARAMVALUETO, QUERYSTRING,

PROC\_NAME, USERNAME, SYSROWID, LASTMODIFIED, ERROR\_MSG)

Values

('NJ0002', '0010xx', 'MPL', 13, 'ELIGIBLE\_YEARS',

'ELIGIBLE YEARS', 0, 2, 50, 'SELECT CASE WHEN B.SERVICE\_YEAR BETWEEN PARAMVALUEFROM AND PARAMVALUETO THEN ''#SUCCESS'' '||CHR(13)||CHR(10)||'ELSE ''SERVICE YEAR MUST BE BETWEEN ''||PARAMVALUEFROM|| '' TO ''||PARAMVALUETO||'' YEARS'' END '||CHR(13)||CHR(10)||'FROM PFLOANSANCTIONPARAM A,'||CHR(13)||CHR(10)||'('||CHR(13)||CHR(10)||' SELECT (TO\_NUMBER(SUBSTR(''<<LOANDATE>>'',-4)) -TO\_NUMBER(TO\_CHAR(PFJOINDATE,''YYYY''))) SERVICE\_YEAR'||CHR(13)||CHR(10)||' FROM PFEMPLOYEEMASTER'||CHR(13)||CHR(10)||' WHERE PFNO = ''<<PFNO>>'''||CHR(13)||CHR(10)||') B'||CHR(13)||CHR(10)||'WHERE 1=1'||CHR(13)||CHR(10)||'AND PARAMCODE=''ELIGIBLE\_YEARS'''||CHR(13)||CHR(10)||'AND COMPANYCODE=''<<COMPANYCODE>>'''||CHR(13)||CHR(10)||'AND DIVISIONCODE=''<<DIVISIONCODE>>'''||CHR(13)||CHR(10)||'AND LOANCODE=''<<LOANCODE>>''',

NULL, NULL, NULL, NULL, NULL);

COMMIT;

CREATE OR REPLACE PROCEDURE NJMCL\_WEB.PROC\_PFLOANSANCTION

(

P\_COMPANYCODE VARCHAR2,

P\_DIVISIONCODE VARCHAR2,

P\_PARAMS VARCHAR2,

P\_PARAMTAG VARCHAR2 DEFAULT NULL

)

AS

LV\_FIRST VARCHAR2(50);

LV\_SQLSTR VARCHAR2(20000);

LV\_COMPANY VARCHAR2(150);

LV\_DIVISION VARCHAR2(150);

LV\_RETMSG VARCHAR2(1000);

LV\_LOANCODE VARCHAR2(10);

LV\_LOANDATE VARCHAR2(10);

LV\_PFNO VARCHAR2(30);

LV\_LOANAMOUNT NUMBER(10,2);

LV\_QRYVAL VARCHAR2(100);

LV\_SQLERRM VARCHAR2(1000);

BEGIN

LV\_RETMSG := '#SUCCESS';

LV\_SQLSTR:=NULL;

LV\_LOANCODE := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANCODE');

LV\_PFNO := FN\_GET\_PARAMVALUE(P\_PARAMS, 'PFNO');

LV\_LOANDATE := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANDATE');

LV\_LOANAMOUNT := FN\_GET\_PARAMVALUE(P\_PARAMS, 'LOANAMOUNT');

-- DBMS\_OUTPUT.PUT\_LINE(LV\_LOANCODE);

-- DBMS\_OUTPUT.PUT\_LINE(LV\_PFNO);

-- DBMS\_OUTPUT.PUT\_LINE(LV\_LOANDATE);

DELETE FROM PF\_ERROR\_LOG

WHERE PROC\_NAME='PROC\_PFLOANSANCTION';

FOR C1 IN (

SELECT \* FROM PFLOANSANCTIONPARAM

WHERE COMPANYCODE=P\_COMPANYCODE

AND DIVISIONCODE=P\_DIVISIONCODE

AND LOANCODE=LV\_LOANCODE

ORDER BY QUERYINDEX

)

LOOP

LV\_SQLERRM := NULL;

LV\_SQLSTR := NULL;

BEGIN

IF C1.PROC\_NAME IS NOT NULL THEN

LV\_SQLSTR := 'BEGIN ' || FN\_GET\_QRYSTRING(P\_PARAMS, C1.PROC\_NAME) ||' END;';

DBMS\_OUTPUT.PUT\_LINE(LV\_SQLSTR);

INSERT INTO PF\_ERROR\_LOG(COMPANYCODE, DIVISIONCODE, PROC\_NAME,ORA\_ERROR\_MESSG,ERROR\_QUERY,PAR\_VALUES,FORTNIGHTSTARTDATE, FORTNIGHTENDDATE, REMARKS )

VALUES(P\_COMPANYCODE, P\_DIVISIONCODE, 'PROC\_PFLOANSANCTION',LV\_SQLERRM,LV\_SQLSTR,P\_PARAMS,SYSDATE,SYSDATE, 'REMARK');

EXECUTE IMMEDIATE LV\_SQLSTR;

END IF;

IF C1.QUERYSTRING IS NOT NULL THEN

LV\_SQLSTR := FN\_GET\_QRYSTRING(P\_PARAMS, C1.QUERYSTRING);

DBMS\_OUTPUT.PUT\_LINE(LV\_SQLSTR);

EXECUTE IMMEDIATE LV\_SQLSTR INTO LV\_QRYVAL;

DBMS\_OUTPUT.PUT\_LINE(LV\_QRYVAL);

INSERT INTO PF\_ERROR\_LOG(COMPANYCODE, DIVISIONCODE, PROC\_NAME,ORA\_ERROR\_MESSG,ERROR\_QUERY,PAR\_VALUES,FORTNIGHTSTARTDATE, FORTNIGHTENDDATE, REMARKS )

VALUES(P\_COMPANYCODE, P\_DIVISIONCODE, 'PROC\_PFLOANSANCTION',LV\_SQLERRM,LV\_SQLSTR,P\_PARAMS,SYSDATE,SYSDATE, 'REMARK');

EXIT WHEN LV\_QRYVAL <> '#SUCCESS';

END IF;

EXCEPTION WHEN OTHERS THEN

LV\_SQLERRM := SQLERRM ;

INSERT INTO PF\_ERROR\_LOG(COMPANYCODE, DIVISIONCODE, PROC\_NAME,ORA\_ERROR\_MESSG,ERROR\_QUERY,PAR\_VALUES,FORTNIGHTSTARTDATE, FORTNIGHTENDDATE, REMARKS )

VALUES(P\_COMPANYCODE, P\_DIVISIONCODE, 'PROC\_PFLOANSANCTION',LV\_SQLERRM,LV\_SQLSTR,P\_PARAMS,SYSDATE,SYSDATE, 'REMARK');

END;

END LOOP;

-- INSERT INTO SYS\_GBL\_PROCOUTPUT\_INFO(SYS\_SAVE\_INFO) VALUES(LV\_QRYVAL);

DELETE FROM SYS\_GBL\_PROCOUTPUT\_INFO WHERE 1=1;

INSERT INTO SYS\_GBL\_PROCOUTPUT\_INFO SELECT LV\_QRYVAL FROM DUAL;

-- dbms\_output.put\_line(lv\_sqlstr);

-- EXECUTE IMMEDIATE LV\_SQLSTR;

EXCEPTION WHEN OTHERS THEN NULL;

LV\_SQLERRM := sqlerrm ;

-- INSERT INTO PF\_ERROR\_LOG(COMPANYCODE, DIVISIONCODE, PROC\_NAME,ORA\_ERROR\_MESSG,ERROR\_QUERY,PAR\_VALUES,FORTNIGHTSTARTDATE, FORTNIGHTENDDATE, REMARKS )

-- VALUES(P\_COMPANYCODE, P\_DIVISIONCODE, 'PROC\_PFLOANSANCTION',LV\_SQLERRM,LV\_SQLSTR,P\_PARAMS,SYSDATE,SYSDATE, 'REMARK');

END;

/