CREATE OR REPLACE PROCEDURE PROC\_CREATE\_SYSTFMAP

(

P\_TABLE\_NAME VARCHAR2,

P\_SYS\_COLUMNNAMES VARCHAR2 DEFAULT NULL,

P\_GBL\_TABLE\_NAME VARCHAR2 DEFAULT NULL,

P\_SEQUENCE NUMBER DEFAULT NULL

)

AS

LV\_SQLSTR VARCHAR2(3200);

LV\_ERROR\_REMARK VARCHAR2(3200);

LV\_MAX\_SEQUENCIER NUMBER(10);

LV\_GBL\_TABLE VARCHAR2(30);

LV\_SEQUENCE VARCHAR2(30);

LV\_CNT NUMBER(10);

BEGIN

-- EXEC PROC\_CREATE\_SYSTFMAP('GPSFORM5TRANSACTION','COMPANYCODE, DIVISIONCODE, YEARCODE, PFREGNO, CATEGORYCODE, EMPLOYEECODE')

IF P\_SEQUENCE IS NOT NULL THEN

DELETE FROM SYS\_TFMAP WHERE SYS\_TABLE\_SEQUENCIER = P\_SEQUENCE;

LV\_MAX\_SEQUENCIER := P\_SEQUENCE;

ELSE

SELECT (MAX(SYS\_TABLE\_SEQUENCIER)+1) INTO LV\_MAX\_SEQUENCIER FROM SYS\_TFMAP;

END IF;

LV\_GBL\_TABLE := 'GBL\_'||P\_TABLE\_NAME;

IF P\_GBL\_TABLE\_NAME IS NOT NULL THEN

LV\_GBL\_TABLE := P\_GBL\_TABLE\_NAME;

END IF;

SELECT COUNT(\*) INTO LV\_CNT FROM SYS\_TFMAP

WHERE SYS\_TABLENAME\_TEMP = LV\_GBL\_TABLE;

IF LV\_CNT > 0 THEN

SELECT SYS\_TABLE\_SEQUENCIER INTO LV\_SEQUENCE FROM SYS\_TFMAP

WHERE SYS\_TABLENAME\_TEMP = LV\_GBL\_TABLE and rownum=1;

DBMS\_OUTPUT.PUT\_LINE('SELECT \* FROM SYS\_TFMAP WHERE SYS\_TABLE\_SEQUENCIER = '||LV\_SEQUENCE||CHR(10));

LV\_ERROR\_REMARK := 'Validation Failure : '||chr(10)||'TEMPORARY TABLE ALREADY EXIST.'||chr(10)||'SYS\_TABLE\_SEQUENCIER : '||LV\_SEQUENCE||chr(10)||' SYS\_TABLENAME\_TEMP : '||LV\_GBL\_TABLE|| ' '||chr(10)||chr(10);

RAISE\_APPLICATION\_ERROR(TO\_NUMBER(FN\_DISPLAY\_ERROR( 'COMMON')),FN\_DISPLAY\_ERROR( 'COMMON',6,LV\_ERROR\_REMARK));

END IF;

DBMS\_OUTPUT.PUT\_LINE('SELECT \* FROM SYS\_TFMAP WHERE SYS\_TABLE\_SEQUENCIER = '||LV\_MAX\_SEQUENCIER||CHR(10));

INSERT INTO SYS\_TFMAP

(

SYS\_TABLE\_SEQUENCIER, SYS\_TABLENAME\_ACTUAL, SYS\_TABLENAME\_TEMP, SYS\_COLUMNNAME,

SYS\_COLUMN\_SEQUENCE, SYS\_DATATYPE, SYS\_COLUMN\_LENGTH,

SYS\_COLUMN\_PRECISION, SYS\_KEYCOLUMN,

SYS\_MAPFIELD, SYS\_TIMESTAMP, SYS\_USEMAP, SYS\_DEFAULT, TDSCODE

)

SELECT LV\_MAX\_SEQUENCIER SYS\_TABLE\_SEQUENCIER,TABLE\_NAME SYS\_TABLENAME\_ACTUAL,SYS\_TABLENAME\_TEMP,COLUMN\_NAME SYS\_COLUMNNAME,

NVL(B.SEQ\_INDX,0) SYS\_COLUMN\_SEQUENCE,DATA\_TYPE SYS\_DATATYPE,DATA\_LENGTH SYS\_COLUMN\_LENGTH,

NVL(DATA\_PRECISION,0) SYS\_COLUMN\_PRECISION,NVL(B.KEYCOLUMN, 'N') SYS\_KEYCOLUMN,

COLUMN\_NAME SYS\_MAPFIELD,SYSDATE SYS\_TIMESTAMP,'Y' SYS\_USEMAP,NULL SYS\_DEFAULT,NULL TDSCODE FROM

(

SELECT TABLE\_NAME,LV\_GBL\_TABLE SYS\_TABLENAME\_TEMP,COLUMN\_NAME,DATA\_TYPE,

DECODE(DATA\_TYPE,'NUMBER',DATA\_PRECISION,DATA\_LENGTH) DATA\_LENGTH,DECODE(DATA\_TYPE,'NUMBER',DATA\_SCALE,DATA\_PRECISION) DATA\_PRECISION

FROM COLS

WHERE TABLE\_NAME=P\_TABLE\_NAME

AND COLUMN\_NAME <> 'LASTMODIFIED'

UNION ALL

SELECT 'NONE',LV\_GBL\_TABLE, 'OPERATIONMODE' COLUMN\_NAME,'VARCHAR2' DATA\_TYPE,1 DATA\_LENGTH,NULL DATA\_PRECISION FROM DUAL

) A,

(

SELECT LEVEL SEQ\_INDX,'Y' KEYCOLUMN, trim(regexp\_substr(P\_SYS\_COLUMNNAMES, '[^,]+', 1, LEVEL)) COLNAME

FROM dual

CONNECT BY LEVEL <= regexp\_count(P\_SYS\_COLUMNNAMES, ',')+1

)B

WHERE A.COLUMN\_NAME=B.COLNAME(+);

DBMS\_OUTPUT.PUT\_LINE('EXEC PROC\_CREATE\_GBL\_TMP\_TABLES('||LV\_MAX\_SEQUENCIER||',0)');

-- EXECUTE IMMEDIATE LV\_SQLSTR;

END;

/

ALTER TABLE JSWRAPPER

DROP PRIMARY KEY CASCADE;

DROP TABLE JSWRAPPER CASCADE CONSTRAINTS;

CREATE TABLE BAGGAGE\_WEB.JSWRAPPER

(

JSCRIPT\_ID VARCHAR2(20 BYTE),

TAG VARCHAR2(500 BYTE),

JSCRIPT\_TYPE VARCHAR2(100 BYTE),

JSCRIPT\_TEXT VARCHAR2(3999 BYTE),

REMARKS VARCHAR2(1000 BYTE),

MODULE VARCHAR2(100 BYTE),

CREATEDBY VARCHAR2(100 BYTE),

LASTMODIFIED DATE DEFAULT SYSDATE

)

CREATE UNIQUE INDEX JSWRAPPER\_PK ON BAGGAGE\_WEB.JSWRAPPER

(JSCRIPT\_ID) ;

ALTER TABLE JSWRAPPER ADD (

CONSTRAINT JSWRAPPER\_PK

PRIMARY KEY

(JSCRIPT\_ID)

USING INDEX JSWRAPPER\_PK

ENABLE VALIDATE);

SET DEFINE OFF;

Insert into JSWRAPPER

(JSCRIPT\_ID, TAG, JSCRIPT\_TYPE, JSCRIPT\_TEXT, REMARKS,

MODULE, CREATEDBY, LASTMODIFIED)

Values

('JS0001', 'REIMBURSEMENT\_LTA', 'JSON', '{'||CHR(10)||' data: [],'||CHR(10)||' colWidths: [80, 210, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100, 100],'||CHR(10)||' startRows: 2,'||CHR(10)||' colHeaders: [''Token <br>No<br>'','||CHR(10)||' ''Employee<br> Name<br>'','||CHR(10)||' ''Date of<br>Joining<br>'','||CHR(10)||' ''Eligible<br>Amount'','||CHR(10)||' ''Amount<br>Taken'','||CHR(10)||' ''Previous<br> Year Amount'','||CHR(10)||' ''Previous<br>Year Taken'','||CHR(10)||' ''Previous<br>Year Balance'','||CHR(10)||' ''Total<br>Due'','||CHR(10)||' ''Amount<br>Claim Current'','||CHR(10)||' ''Amount<br>Claim Previous'','||CHR(10)||' ''Balance<br>Amount'','||CHR(10)||' ''For The<br>Year''],'||CHR(10)||' fillHandle: false,'||CHR(10)||' manualColumnResize: true,'||CHR(10)||' fixedColumnsLeft: 2,'||CHR(10)||' manualColumnFreeze: true,'||CHR(10)||' columns: ['||CHR(10)||' { data: "TOKENNO", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'' },'||CHR(10)||' { data: "EMPLOYEENAME", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'' },'||CHR(10)||' { data: "DATEOFJOIN", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'', className: "htCenter" },'||CHR(10)||' { data: "COMPONENTAMOUNT", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTTAKEN", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "PREVCOMPONENTAMOUNT", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "PREVIOUSYRTAKEN", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "PREVIOUSYRBALANCE", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "TOTALDUE", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTCLAIMCURRENT", type: ''numeric'', format: ''0.00'', renderer: editablecolor, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTCLAIMPREVIOUS", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "BALANCEAMOUNT", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "YEARCODE", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' ],'||CHR(10)||' afterChange: AfterChange,'||CHR(10)||'}', NULL,

'PIS', 'CHIRANJIT GHOSH', TO\_DATE('2/7/2020 1:21:29 PM', 'MM/DD/YYYY HH:MI:SS AM'));

Insert into JSWRAPPER

(JSCRIPT\_ID, TAG, JSCRIPT\_TYPE, JSCRIPT\_TEXT, REMARKS,

MODULE, CREATEDBY, LASTMODIFIED)

Values

('JS0002', 'REIMBURSEMENT\_MEDICAL', 'JSON', '{'||CHR(10)||' data: [],'||CHR(10)||' colWidths: [80, 210, 100, 100, 100, 100, 100, 100, 100, 100],'||CHR(10)||' startRows: 2,'||CHR(10)||' colHeaders: ['||CHR(10)||' ''Token <br>No<br>'','||CHR(10)||' ''Employee<br> Name<br>'','||CHR(10)||' ''Date of<br>Joining<br>'','||CHR(10)||' ''Eligible<br>Amount'','||CHR(10)||' ''Amount<br>Taken'','||CHR(10)||' ''Total<br>Due'','||CHR(10)||' ''Amount<br>Claim Current'','||CHR(10)||' ''Amount<br>Claim Previous'','||CHR(10)||' ''Balance<br>Amount'','||CHR(10)||' ''For The<br>Year'''||CHR(10)||' ],'||CHR(10)||' fillHandle: false,'||CHR(10)||' manualColumnResize: true,'||CHR(10)||' fixedColumnsLeft: 2,'||CHR(10)||' manualColumnFreeze: true,'||CHR(10)||' columns: ['||CHR(10)||' { data: "TOKENNO", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'' },'||CHR(10)||' { data: "EMPLOYEENAME", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'' },'||CHR(10)||' { data: "DATEOFJOIN", type: ''text'', readOnly: true, renderer: noneditablecolor, language: ''en'', className: "htCenter" },'||CHR(10)||' { data: "COMPONENTAMOUNT", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTTAKEN", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "TOTALDUE", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTCLAIMCURRENT", type: ''numeric'', format: ''0.00'', renderer: editablecolor, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "AMOUNTCLAIMPREVIOUS", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "BALANCEAMOUNT", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' { data: "YEARCODE", type: ''numeric'', format: ''0.00'', readOnly: true, renderer: noneditablecolorNumeric, language: ''en'', className: "htRight" },'||CHR(10)||' ],'||CHR(10)||' afterChange: AfterChange,'||CHR(10)||'}', NULL,

'PIS', 'CHIRANJIT GHOSH', TO\_DATE('2/7/2020 1:18:06 PM', 'MM/DD/YYYY HH:MI:SS AM'));

COMMIT;

SET DEFINE OFF;

Insert into JSWRAPPER

(JSCRIPT\_ID, TAG, JSCRIPT\_TYPE, JSCRIPT\_TEXT, REMARKS,

MODULE, CREATEDBY, LASTMODIFIED)

Values

('JS0003', 'REIMBURSEMENT\_JSFUNCTIONS', 'FUNCTIONS', 'function CreateDynamicGrid() {'||CHR(13)||CHR(10)||' var KeyData = document.getElementById(''mainContent\_ddlCOMPONENTCODE'').value;'||CHR(13)||CHR(10)||' var operationmode = document.getElementById(''mainContent\_txtOPTMODE'').value;'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' var data = [];'||CHR(13)||CHR(10)||' $.ajax({'||CHR(13)||CHR(10)||' type: "POST",'||CHR(13)||CHR(10)||' url: "pgPISReimbursementDetails.aspx/GetHandsOnGridData",'||CHR(13)||CHR(10)||' contentType: "application/json; charset=utf-8",'||CHR(13)||CHR(10)||' data: JSON.stringify({ strKeyData: KeyData, strParams: ''GETHANSONDATA'', strOperationMode: operationmode }),'||CHR(13)||CHR(10)||' dataType: "json",'||CHR(13)||CHR(10)||' async: false,'||CHR(13)||CHR(10)||' beforeSend: function () {'||CHR(13)||CHR(10)||' InvokeLoading();'||CHR(13)||CHR(10)||' },'||CHR(13)||CHR(10)||' success: function (res) {'||CHR(13)||CHR(10)||' DisposeLoading();'||CHR(13)||CHR(10)||' if (res.d != null && res.d.length > 0) {'||CHR(13)||CHR(10)||' var strVal = String(res.d);'||CHR(13)||CHR(10)||' var data = eval(String(strVal).replace(/\n|\r/g, ""))'||CHR(13)||CHR(10)||' var $container = $("div").find("#dataTable");'||CHR(13)||CHR(10)||' $container.handsontable(data[0]);'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' },'||CHR(13)||CHR(10)||' error: function (xhr, status) {'||CHR(13)||CHR(10)||' DisposeLoading();'||CHR(13)||CHR(10)||' alert("An error occurred: " + status);'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' });'||CHR(13)||CHR(10)||'};'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||'function AfterChange(changes, data) {'||CHR(13)||CHR(10)||' if (!changes) {'||CHR(13)||CHR(10)||' return;'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' $.each(changes, function (index, element) {'||CHR(13)||CHR(10)||' var change = element;'||CHR(13)||CHR(10)||' var rowIndex = change[0];'||CHR(13)||CHR(10)||' var columnIndex = returnColIndex(change[1]);'||CHR(13)||CHR(10)||' //alert(change + " " + columnIndex);'||CHR(13)||CHR(10)||' if (change[1] == "AMOUNTCLAIMCURRENT") {'||CHR(13)||CHR(10)||' var bal = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("TOTALDUE"));'||CHR(13)||CHR(10)||' var chkbal = change[3];'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' if (isNaN(chkbal)) {'||CHR(13)||CHR(10)||' alert(chkbal + " is not a number.");'||CHR(13)||CHR(10)||' $("#dataTable").handsontable(''setDataAtCell'', rowIndex, returnColIndex("AMOUNTCLAIMCURRENT"), change[2]);'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' else if (chkbal > bal) {'||CHR(13)||CHR(10)||' //alert("Claim Amount (" + chkbal + ") can not be greater than Total Due Amount (" + bal + ")");'||CHR(13)||CHR(10)||' //$("#dataTable").handsontable(''setDataAtCell'', rowIndex, returnColIndex("AMOUNTCLAIMCURRENT"), change[2]);'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' else if (chkbal < 0) {'||CHR(13)||CHR(10)||' alert("Claim Amount (" + chkbal + ") can not be less than zero (0)");'||CHR(13)||CHR(10)||' $("#dataTable").handsontable(''setDataAtCell'', rowIndex, returnColIndex("AMOUNTCLAIMCURRENT"), change[2]);'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' else {'||CHR(13)||CHR(10)||' //BALANCE AMOUNT=(ELIGIBLEAMOUNT+PREVIOUSYRBALENCE)-(PREVIOUSYRTAKEN+AMOUNTCLAIMCURRENT+AMOUNTCLAIMPREVIOUS)'||CHR(13)||CHR(10)||' var COMPONENTAMOUNT = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("COMPONENTAMOUNT"));'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' var compCode = document.getElementById(''mainContent\_ddlCOMPONENTCODE'').value;'||CHR(13)||CHR(10)||' var PREVIOUSYRBALANCE = 0;'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' if (compCode == ''LTA'') {'||CHR(13)||CHR(10)||' PREVIOUSYRBALANCE = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("PREVIOUSYRBALANCE"));'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' var AMOUNTTAKEN = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("AMOUNTTAKEN"));'||CHR(13)||CHR(10)||' var AMOUNTCLAIMCURRENT = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("AMOUNTCLAIMCURRENT"));'||CHR(13)||CHR(10)||' var AMOUNTCLAIMPREVIOUS = $("#dataTable").handsontable(''getDataAtCell'', rowIndex, returnColIndex("AMOUNTCLAIMPREVIOUS"));'||CHR(13)||CHR(10)||' //$("#dataTable").handsontable(''setDataAtCell'', rowIndex, returnColIndex("BALANCEAMOUNT"), (COMPONENTAMOUNT + PREVIOUSYRBALANCE - (AMOUNTTAKEN + AMOUNTCLAIMCURRENT + AMOUNTCLAIMPREVIOUS)));'||CHR(13)||CHR(10)||' $("#dataTable").handsontable(''setDataAtCell'', rowIndex, returnColIndex("BALANCEAMOUNT"), (COMPONENTAMOUNT + PREVIOUSYRBALANCE - (AMOUNTTAKEN + AMOUNTCLAIMCURRENT)));'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' });'||CHR(13)||CHR(10)||' document.getElementById(''mainContent\_txtGRIDDATA'').value = '''';'||CHR(13)||CHR(10)||' fn\_GetGridData();'||CHR(13)||CHR(10)||'};'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||'function returnColIndex(searchStr) {'||CHR(13)||CHR(10)||' var DataFields = $("#dataTable").handsontable(''getSettings'').columns;'||CHR(13)||CHR(10)||' for (var j = 0; j < DataFields.length; j++) {'||CHR(13)||CHR(10)||' if (DataFields[j].data == searchStr) {'||CHR(13)||CHR(10)||' return j;'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' }'||CHR(13)||CHR(10)||' return -1;'||CHR(13)||CHR(10)||'};'||CHR(13)||CHR(10)||'function rePopulateGrid() {'||CHR(13)||CHR(10)||' var data1 = document.getElementById(''mainContent\_txtGRIDDATA'').value;'||CHR(13)||CHR(10)||' var data = jQuery.parseJSON(data1);'||CHR(13)||CHR(10)||' CreateDynamicGrid();'||CHR(13)||CHR(10)||' $("div").find("#dataTable").handsontable("loadData", data);'||CHR(13)||CHR(10)||'};'||CHR(13)||CHR(10)||CHR(13)||CHR(10)||' '||CHR(13)||CHR(10)||'function fn\_GetGridData() {'||CHR(13)||CHR(10)||' var $container = $("div").find("#dataTable");'||CHR(13)||CHR(10)||' var handsontable = $container.data(''handsontable'');'||CHR(13)||CHR(10)||' var myData = handsontable.getData();'||CHR(13)||CHR(10)||' //alert(''calling charge data, No of rows'' +rowCount);'||CHR(13)||CHR(10)||' document.getElementById(''mainContent\_txtGRIDDATA'').value = JSON.stringify(myData);'||CHR(13)||CHR(10)||'};', NULL,

'PIS', 'CHIRANJIT GHOSH', TO\_DATE('2/7/2020 1:18:56 PM', 'MM/DD/YYYY HH:MI:SS AM'));

COMMIT;

SET DEFINE OFF;

Insert into JSWRAPPER

(JSCRIPT\_ID, TAG, JSCRIPT\_TYPE, JSCRIPT\_TEXT, REMARKS,

MODULE, CREATEDBY, LASTMODIFIED)

Values

('JS0004', 'returnFunction', 'FUNCTIONS', ''||CHR(10)||' function returnFunction(fnName) {'||CHR(10)||CHR(10)||' var fnStr = "";'||CHR(10)||' //var fn = eval("Function(\"str\", \"alert(str + '' B'');\")");'||CHR(10)||' // var fn = Function("str", "alert(str + '' a'');");'||CHR(10)||' $.ajax({'||CHR(10)||' type: "POST",'||CHR(10)||' url: "../../../DataAccessService.asmx/GetJSFunctionBody",'||CHR(10)||' contentType: "application/json; charset=utf-8",'||CHR(10)||' data: JSON.stringify({ functionName: fnName }),'||CHR(10)||' async: false,'||CHR(10)||' dataType: "json",'||CHR(10)||' beforeSend: function () {'||CHR(10)||' InvokeLoading();'||CHR(10)||' },'||CHR(10)||' success: function (res) {'||CHR(10)||' //fn = res.d;'||CHR(10)||' DisposeLoading();'||CHR(10)||' fnStr = res.d;'||CHR(10)||' //fnStr = "Function(" + paramName + ", " + "\"" + res.d + "\")";'||CHR(10)||' //fn = Function("str", res.d);'||CHR(10)||' },'||CHR(10)||' error: function (xhr, status) {'||CHR(10)||' DisposeLoading();'||CHR(10)||' alert("An error occurred: " + status);'||CHR(10)||' }'||CHR(10)||' });'||CHR(10)||CHR(10)||' //var fn = eval(fnStr);'||CHR(10)||' //return fn;'||CHR(10)||' return fnStr;'||CHR(10)||' };'||CHR(10)||'', NULL,

'STORES', 'CHIRANJIT GHOSH', TO\_DATE('3/20/2020 3:10:52 PM', 'MM/DD/YYYY HH:MI:SS AM'));

COMMIT;

DROP TABLE PISREIMBURSEMENTDETAILS CASCADE CONSTRAINTS;

CREATE TABLE PISREIMBURSEMENTDETAILS

(

COMPANYCODE VARCHAR2(10 BYTE),

DIVISIONCODE VARCHAR2(10 BYTE),

YEARCODE VARCHAR2(10 BYTE),

FORYEARCODE VARCHAR2(10 BYTE),

YEARMONTH VARCHAR2(10 BYTE),

WORKERSERIAL VARCHAR2(10 BYTE),

TOKENNO VARCHAR2(10 BYTE),

CATEGORYCODE VARCHAR2(10 BYTE),

GRADECODE VARCHAR2(10 BYTE),

COMPONENTCODE VARCHAR2(10 BYTE),

COMPONENTAMOUNT NUMBER(11,2),

TRANSACTIONTYPE VARCHAR2(10 BYTE),

ADDLESS VARCHAR2(10 BYTE),

REMARKS VARCHAR2(250 BYTE),

BILLAMOUNT NUMBER(11,2),

PAIDAMOUNT NUMBER(11,2),

LEAVEDATEFROM DATE,

LEAVEDATETO DATE,

SYSROWID VARCHAR2(50 BYTE),

USERNAME VARCHAR2(100 BYTE),

LASTMODIFIED DATE DEFAULT SYSDATE,

TRANTYPE VARCHAR2(50 BYTE)

)

EXEC PROC\_CREATE\_SYSTFMAP('PISREIMBURSEMENTDETAILS','COMPANYCODE,COMPONENTCODE,DIVISIONCODE,TOKENNO,WORKERSERIAL,YEARCODE,YEARMONTH')

SELECT \* FROM SYS\_TFMAP WHERE SYS\_TABLE\_SEQUENCIER = 20263

EXEC PROC\_CREATE\_GBL\_TMP\_TABLES(20263,0)

SELECT \* FROM MENUMASTER\_RND

WHERE MENUCODE LIKE '01070208'

SELECT \* FROM ROLEDETAILS

WHERE MENUCODE='01070208'

SELECT \* FROM REPORTPARAMETERMASTER

WHERE REPORTTAG=''

SET DEFINE OFF;

Insert into MENUMASTER\_RND

(MENUCODE, MENUNAME, MENUDESC, MENUTYPE, MENUTAG,

MENUTAG1, MENUTAG2, ENTRYPOSSIBLE, EDITPOSSIBLE, DELETEPOSSIBLE,

REPORTSPOSSIBLE, UTILITIESPOSSIBLE, ACTIVE, PROJECTNAME, MENUDETAILS,

EFFECTIVEDIVISION, EFFECTIVEINDIVISION, PROCEDURE\_SAVE\_EDIT, URLSRC, FRM\_HEIGHT,

FRM\_WIDTH, FUNCTIONCALL\_BEFORE, FUNCTIONCALL\_AFTER, VIEWPOSSIBLE, PRINTPOSSIBLE,

DOCDATEVALIDATEFIELD, KEY\_TABLE)

Values

('01070208', NULL, 'Reimbursement Details', 'TRANSACTIONS', 'PIS REIMBURSEMENT DETAILS',

'PIS', NULL, 'Y', 'Y', 'Y',

'N', 'N', 'Y', 'PIS', NULL,

'''0001'',''0002'',''0003'',''0004''', '''0001'',''0002'',''0003'',''0004'',''0005''', NULL, 'PIS/PAGES/Transaction/pgPISReimbursementDetails.aspx', NULL,

NULL, 'prcWPS\_LoanEntry\_b4Save', NULL, NULL, NULL,

NULL, NULL);

COMMIT;

SET DEFINE OFF;

Insert into ROLEDETAILS

(ROLEID, MENUCODE, ACCESSENTRY, ACCESSEDIT, ACCESSDELETE,

ACCESSREPORT, ACCESSUTILITIES, ACCESSVIEW, ACCESSPRINT, PROJECTNAME,

SYSROWID)

Values

('1', '01070208', 'Y', 'Y', 'Y',

'X', 'X', 'X', 'X', 'PIS',

'202005271045580008869959');

Insert into ROLEDETAILS

(ROLEID, MENUCODE, ACCESSENTRY, ACCESSEDIT, ACCESSDELETE,

ACCESSREPORT, ACCESSUTILITIES, ACCESSVIEW, ACCESSPRINT, PROJECTNAME,

SYSROWID)

Values

('25', '01070208', 'Y', 'Y', 'N',

'X', 'X', 'X', 'X', 'PIS',

'202006171017100008930916');

COMMIT;

DROP TABLE BAGGAGE\_WEB.GTT\_PISREIMBURSEMENT CASCADE CONSTRAINTS;

CREATE GLOBAL TEMPORARY TABLE BAGGAGE\_WEB.GTT\_PISREIMBURSEMENT

(

COMPANYCODE VARCHAR2(10 BYTE),

DIVISIONCODE VARCHAR2(10 BYTE),

YEARCODE VARCHAR2(10 BYTE),

FORYEARCODE VARCHAR2(10 BYTE),

YEARMONTH VARCHAR2(10 BYTE),

WORKERSERIAL VARCHAR2(10 BYTE),

TOKENNO VARCHAR2(20 BYTE),

EMPLOYEENAME VARCHAR2(500 BYTE),

DATEOFJOIN VARCHAR2(20 BYTE),

UNITCODE VARCHAR2(20 BYTE),

CATEGORYCODE VARCHAR2(20 BYTE),

GRADECODE VARCHAR2(10 BYTE),

COMPONENTCODE VARCHAR2(20 BYTE),

COMPONENTAMOUNT NUMBER(15,2),

TRANSACTIONTYPE VARCHAR2(20 BYTE),

ADDLESS VARCHAR2(10 BYTE),

BILLAMOUNT NUMBER(15,2),

PAIDAMOUNT NUMBER(15,2),

TRANTYPE VARCHAR2(20 BYTE),

AMOUNTTAKEN NUMBER(15,2),

PREVCOMPONENTAMOUNT NUMBER(15,2),

PREVIOUSYRBALANCE NUMBER(15,2),

PREVIOUSYRTAKEN NUMBER(15,2),

TOTALDUE NUMBER(15,2),

AMOUNTCLAIMCURRENT NUMBER(15,2),

AMOUNTCLAIMPREVIOUS NUMBER(15,2),

BALANCEAMOUNT NUMBER(15,2),

LEAVEBALANCE NUMBER(15,2),

LEAVEENCASHMENT NUMBER(15,2)

)

ON COMMIT PRESERVE ROWS

NOCACHE;