**JDEVELOPER(OAF) CODE:-**

**CONTROLLER CODE:-**

/\*===========================================================================+

| Copyright (c) 2001, 2005 Oracle Corporation, Redwood Shores, CA, USA |

| All rights reserved. |

+===========================================================================+

| HISTORY |

+===========================================================================\*/

package xxfe.oracle.apps.fnd.responsibility.webui;

import com.sun.rowset.internal.Row;

import oracle.apps.fnd.framework.OAException;

import oracle.apps.fnd.framework.OAApplicationModule;

import java.io.Serializable;

import oracle.apps.fnd.common.VersionInfo;

import oracle.apps.fnd.framework.OAApplicationModule;

import oracle.apps.fnd.framework.OAViewObject;

import oracle.apps.fnd.framework.webui.OAControllerImpl;

import oracle.apps.fnd.framework.webui.OAPageContext;

import oracle.apps.fnd.framework.webui.beans.OAWebBean;

import oracle.apps.fnd.wf.engine.\*;

import oracle.apps.fnd.wf.\*;

import xxfe.oracle.apps.fnd.responsibility.server.ResponsibilityAMImpl;

import java.sql.CallableStatement;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.Timestamp;

import java.text.DateFormat;

import java.text.ParseException;

import oracle.jbo.domain.Date;

import java.sql.SQLException;

import oracle.apps.fnd.framework.OARow;

import oracle.apps.fnd.framework.server.OADBTransaction;

import oracle.apps.fnd.framework.server.OAViewRowImpl;

//import oracle.jbo.server.DBTransaction;

import oracle.apps.fnd.framework.webui.beans.message.OAMessageDateFieldBean;

import oracle.apps.fnd.framework.webui.beans.message.OAMessageTextInputBean;

import oracle.apps.fnd.i18n.common.text.SimpleDateFormat;

import oracle.jbo.domain.Number ;

import oracle.jbo.RowSetIterator;

import xxfe.oracle.apps.fnd.responsibility.server.ResVOImpl;

import xxfe.oracle.apps.fnd.responsibility.server.ResVORowImpl;

/\*\*

\* Controller for ...

\*/

public class ResponsibilityCO extends OAControllerImpl

{

public static final String RCS\_ID="$Header$";

public static final boolean RCS\_ID\_RECORDED =

VersionInfo.recordClassVersion(RCS\_ID, "%packagename%");

/\*\*

\* Layout and page setup logic for a region.

\* @param pageContext the current OA page context

\* @param webBean the web bean corresponding to the region

\*/

public void processRequest(OAPageContext pageContext, OAWebBean webBean)

{

super.processRequest(pageContext, webBean);

OAApplicationModule am1 =pageContext.getApplicationModule(webBean);

OAViewObject vo = (OAViewObject)am1.findViewObject("ResVO");

String userid = String.valueOf(pageContext.getUserId());

Serializable [] parameters1 = {userid};

Class paramTypes[] = {String.class};

am1.invokeMethod("assign\_userid",parameters1,paramTypes);

//vo.setWhereClauseParam(0,userid);

vo.executeQuery();

//-------------------------------------

//Setting default Start date

//-------------------------------------

OADBTransaction trx=(OADBTransaction)am1.getOADBTransaction();

java.sql.Date val=(trx.getCurrentDBDate()).dateValue();

/\*OAMessageDateFieldBean textbean1=(OAMessageDateFieldBean )webBean.findIndexedChildRecursive("StartDate");

textbean1.setValue(pageContext,val);\*/

//-------------------------------------

//Setting default Start date- CODE 2

//-------------------------------------

ResVORowImpl row=null;

int fetchedRowCount = vo.getFetchedRowCount();

RowSetIterator SelectIter = vo.createRowSetIterator("SelectIter ");

if(fetchedRowCount != 0)

{

SelectIter .setRangeStart(0);

SelectIter .setRangeSize(fetchedRowCount);

for (int i = 0; i < fetchedRowCount; i++)

{

row = (ResVORowImpl)SelectIter .getRowAtRangeIndex(i);

row.setAttribute("StartDate",val);

}

}

//-------------------------------

//CODE TO SET MIN AVAIALBLE DATE

//-------------------------------

OAMessageDateFieldBean dtBean = (OAMessageDateFieldBean)webBean.findChildRecursive("StartDate");

OAMessageDateFieldBean dtBean1 = (OAMessageDateFieldBean)webBean.findChildRecursive("EndDate");

String dtSql= "Select (sysdate-1) min\_date from dual";

try {

Connection conn = am1.getOADBTransaction().getJdbcConnection();

PreparedStatement stmt = conn.prepareStatement(dtSql);

ResultSet rs = stmt.executeQuery(dtSql);

while (rs.next())

{

java.sql.Date minDate = rs.getDate("min\_date");

dtBean.setMinValue(minDate);

dtBean1.setMinValue(minDate);

}

}

catch(Exception e){

e.printStackTrace();

}

}

/\*\*

\* Procedure to handle form submissions for form elements in

\* a region.

\* @param pageContext the current OA page context

\* @param webBean the web bean corresponding to the region

\*/

public void processFormRequest(OAPageContext pageContext, OAWebBean webBean)

{

super.processFormRequest(pageContext, webBean);

ResponsibilityAMImpl am=(ResponsibilityAMImpl)pageContext.getApplicationModule(webBean);

OAViewObject vo = (OAViewObject)am.findViewObject("ResVO");

if(pageContext.getParameter("SearchButton")!=null)

{

String userid = String.valueOf(pageContext.getUserId());

Serializable [] parameters1 = {userid};

Class paramTypes[] = {String.class};

am.invokeMethod("assign\_userid",parameters1,paramTypes);

StringBuffer whereClause = new StringBuffer(100);

String Rsp = pageContext.getParameter("Search");

whereClause.append(" RESPONSIBILITY\_NAME like '"+ Rsp +"'");

vo.setWhereClause(whereClause.toString());

//int userid = pageContext.getUserId();

//vo.setWhereClauseParam(0,userid);

vo.executeQuery();

}

//------------------

//REQUEST METHOD

//------------------

if(pageContext.getParameter("Request")!= null )

{

int flag=0;

oracle.jbo.domain.Date startdate = null;

ResVORowImpl row=null;

String userid = new String();

String sel = new String();

String notif = new String();

int fetchedRowCount = vo.getFetchedRowCount();

RowSetIterator SelectIter = vo.createRowSetIterator("SelectIter ");

if(fetchedRowCount != 0)

{

SelectIter .setRangeStart(0);

SelectIter .setRangeSize(fetchedRowCount);

for (int i = 0; i < fetchedRowCount; i++)

{

row = (ResVORowImpl)SelectIter .getRowAtRangeIndex(i);

sel= (String)row.getAttribute("Select");

if ("Y".equals(sel))

{

userid = String.valueOf(pageContext.getUserId());

Number resid = (Number)row.getAttribute("ResponsibilityId");

Number appid = (Number)row.getAttribute("ApplicationId");

String note = (String)row.getAttribute("Notes");

startdate = (oracle.jbo.domain.Date)row.getAttribute("StartDate");

oracle.jbo.domain.Date enddate = (oracle.jbo.domain.Date)row.getAttribute("EndDate");

if(startdate == null)

{

flag =1;

break;

}

//--------------------------

//Initiating workflow

//--------------------------

Serializable [] parameter2 = {userid,resid,appid,startdate,enddate,note};

Class paramtypes2[] = {String.class,Number.class,Number.class,oracle.jbo.domain.Date.class,oracle.jbo.domain.Date.class,String.class};

am.invokeMethod("workflow\_start",parameter2,paramtypes2);

//start = (Date)row.getAttribute("StartDate");

//Passing APP ID for notification ID

//Inserting into the records

Serializable [] parameters1 = {userid,resid,appid,startdate,enddate,note};

Class paramTypes[] = {String.class, Number.class,Number.class,oracle.jbo.domain.Date.class,oracle.jbo.domain.Date.class,String.class};

am.invokeMethod("insertAction",parameters1,paramTypes);

}

}

}

if (flag == 1)

throw new OAException("ERROR!Start date cannot be empty!",OAException.ERROR);

else

throw new OAException("Submitted for approval! " ,OAException.INFORMATION);

}

//------------------------------

//REVOKING THE RESPONSIBILITIES

//------------------------------

if(pageContext.getParameter("Revoke")!= null )

{

ResVORowImpl row=null;

String respon\_check = new String();

String sel = new String();

int fetchedRowCount = vo.getFetchedRowCount();

RowSetIterator SelectIter = vo.createRowSetIterator("SelectIter ");

if(fetchedRowCount != 0)

{

SelectIter .setRangeStart(0);

SelectIter .setRangeSize(fetchedRowCount);

for (int i = 0; i < fetchedRowCount; i++)

{

row = (ResVORowImpl)SelectIter .getRowAtRangeIndex(i);

sel= (String)row.getAttribute("Select");

if ("Y".equals(sel))

{

String userid = String.valueOf(pageContext.getUserId());

Number resid = (Number)row.getAttribute("ResponsibilityId");

Number appid = (Number)row.getAttribute("ApplicationId");

//Passing APP ID for notification ID

//Inserting into the records

Serializable [] parameters1 = {userid,resid,appid};

Class paramTypes[] = {String.class, Number.class,Number.class};

am.invokeMethod("revokeAction",parameters1,paramTypes);

}

}

}

throw new OAException("ACCESS REVOKED!",OAException.INFORMATION);

}

if(pageContext.getParameter("Cancel")!= null)

{

int flag=0;

oracle.jbo.domain.Date startdate = null;

ResVORowImpl row=null;

String sel = new String();

int fetchedRowCount = vo.getFetchedRowCount();

RowSetIterator SelectIter = vo.createRowSetIterator("SelectIter ");

if(fetchedRowCount != 0)

{

SelectIter .setRangeStart(0);

SelectIter .setRangeSize(fetchedRowCount);

for (int i = 0; i < fetchedRowCount; i++)

{

row = (ResVORowImpl)SelectIter .getRowAtRangeIndex(i);

sel= (String)row.getAttribute("Select");

if ("Y".equals(sel))

{

String userid = String.valueOf(pageContext.getUserId());

Number resid = (Number)row.getAttribute("ResponsibilityId");

Number appid = (Number)row.getAttribute("ApplicationId");

Serializable [] parameters1 = {userid,resid,appid};

Class paramTypes[] = {String.class, Number.class,Number.class};

am.invokeMethod("cancelAction",parameters1,paramTypes);

}

}

}

throw new OAException("Request has been CANCELLED! ",OAException.INFORMATION);

}

}

}

**APPLICATION MODULE(AM) CODE:-**

package xxfe.oracle.apps.fnd.responsibility.server;

import java.sql.CallableStatement;

import oracle.jbo.domain.Date;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import oracle.apps.fnd.wf.engine.\*;

import oracle.apps.fnd.wf.\*;

import oracle.apps.fnd.framework.OAApplicationModule;

import oracle.apps.fnd.framework.OAAttrValException;

import oracle.apps.fnd.framework.OAViewObject;

import oracle.apps.fnd.framework.server.OAApplicationModuleImpl;

import oracle.apps.fnd.framework.server.OADBTransaction;

import oracle.apps.fnd.framework.server.OADBTransactionImpl;

import oracle.jdbc.OracleCallableStatement;

import java.sql.Types;

import oracle.jbo.domain.Number ;

import oracle.sql.NUMBER;

import oracle.apps.fnd.framework.OAException;

import oracle.jbo.JboException;

// ---------------------------------------------------------------------

// --- File generated by Oracle ADF Business Components Design Time.

// --- Custom code may be added to this class.

// --- Warning: Do not modify method signatures of generated methods.

// ---------------------------------------------------------------------

public class ResponsibilityAMImpl extends OAApplicationModuleImpl

{

/\*\*This is the default constructor (do not remove)

\*/

public ResponsibilityAMImpl()

{

}

/\*\*Sample main for debugging Business Components code using the tester.

\*/

public static void main(String[] args)

{

launchTester("xxfe.oracle.apps.fnd.responsibility.server", /\* package name \*/

"ResponsibilityAMLocal" /\* Configuration Name \*/);

}

/\*public void insertFake()

{

OAViewObject vo = (OAViewObject)am1.findViewObject("ResVO");

vo.setQueryFake();

}\*/

//Procedure to assign the USER ID to global var in pkg

public void assign\_userid(String userid)

{

OADBTransaction txn = (OADBTransaction)getTransaction();

CallableStatement cs = txn.createCallableStatement("begin XXFE\_APPROVERS\_LIST.XX\_ASSIGN\_USERID(:1); end;",1);

try

{

cs.setInt(1,Integer.parseInt(userid));

cs.execute();

cs.close();

}

catch(SQLException e)

{

throw OAException.wrapperException(e);

}

}

//Procedure for Request Button.

public void insertAction(String userid, Number resid,Number appid,oracle.jbo.domain.Date startdate,oracle.jbo.domain.Date enddate,String note)

{

// int userid\_i=userid.intValue();

int resid\_i= resid.intValue();

int appid\_i=appid.intValue();

OADBTransaction txn = (OADBTransaction)getTransaction();

CallableStatement cs = txn.createCallableStatement("begin XXFE\_APPROVERS\_LIST.XX\_REQUEST(:1,:2,:3,:4,:5,:6); end;",1);

String d1 = txn.getOANLSServices().dateToString(startdate);

String d2 = txn.getOANLSServices().dateToString(enddate);

java.sql.Date startd= txn.getOANLSServices().stringToDate(d1);

java.sql.Date endd= txn.getOANLSServices().stringToDate(d2);

try

{

cs.setInt(1,Integer.parseInt(userid));

cs.setInt(2,resid\_i);

cs.setInt(3,appid\_i);

cs.setDate(4,startd);

cs.setDate(5,endd);

cs.setString(6,note);

cs.execute();

cs.close();

}

catch(SQLException e)

{

throw OAException.wrapperException(e);

}

}

//--------------------

//METHOD TO REVOKE ACCESS

//------------------------

public void revokeAction(String userid, Number resid,Number appid)

{

// int userid\_i=userid.intValue();

int resid\_i= resid.intValue();

int appid\_i=appid.intValue();

OADBTransaction txn = (OADBTransaction)getTransaction();

CallableStatement cs = txn.createCallableStatement("begin XXFE\_APPROVERS\_LIST.XX\_REVOKE(:1,:2,:3); end;",1);

try

{

cs.setInt(1,Integer.parseInt(userid));

cs.setInt(2,resid\_i);

cs.setInt(3,appid\_i);

cs.execute();

cs.close();

}

catch(SQLException e)

{

throw OAException.wrapperException(e);

}

}

/\*\*Container's getter for ResVO

\*/

//----------------------------

//CANCEL METHOD

//---------------------------

public void cancelAction(String userid,Number resid,Number appid)

{

{

// int userid\_i=userid.intValue();

int resid\_i= resid.intValue();

int appid\_i=appid.intValue();

OADBTransaction txn = (OADBTransaction)getTransaction();

CallableStatement cs = txn.createCallableStatement("begin XXFE\_APPROVERS\_LIST.XX\_CANCEL(:1,:2,:3); end;",1);

try

{

cs.setInt(1,Integer.parseInt(userid));

cs.setInt(2,resid\_i);

cs.setInt(3,appid\_i);

cs.execute();

cs.close();

}

catch(SQLException e)

{

throw OAException.wrapperException(e);

}

}

}

//--------------------

//WORKFLOW!!!!

//--------------------

public void workflow\_start(String userid,Number resid,Number appid, oracle.jbo.domain.Date startdate,oracle.jbo.domain.Date enddate,String note)

{

OADBTransaction txn = (OADBTransaction)getTransaction();

int dummy\_var = 0;

int resid\_i= resid.intValue();

int appid\_i=appid.intValue();

String d1 = txn.getOANLSServices().dateToString(startdate);

String d2 = txn.getOANLSServices().dateToString(enddate);

java.sql.Date startd= txn.getOANLSServices().stringToDate(d1);

java.sql.Date endd= txn.getOANLSServices().stringToDate(d2);

CallableStatement cs = txn.createCallableStatement("BEGIN XXFE\_APPROVERS\_LIST.XX\_WORKFLOW\_START(:1,:2,:3,:4,:5,:6); END;",1);

try

{

cs.setInt(1,Integer.parseInt(userid));

cs.setInt(2,resid\_i);

cs.setInt(3,appid\_i);

cs.setDate(4,startd);

cs.setDate(5,endd);

cs.setString(6,note);

cs.execute();

cs.close();

}

catch(SQLException e)

{

throw OAException.wrapperException(e);

}

}

public ResVOImpl getResVO()

{

return (ResVOImpl)findViewObject("ResVO");

}

}