**GIRE**

**Nombre del proyecto** : altaAgente

**ARP**: (Agente rapido pago)

**ARP\_TP** : (parametricas para componentes de clave valor ej : los combo)

ARP\_CF\_MENU\_PANTALLA (se cambia para algunos campos para ver el proceso que se integre)

**UTILIDADES**

**SHORCUT**

control , alt , - : busco archivos

control ,shift , F: busco cosas en los archivos

**URL**:

SERVIDOR LOCAL:

[http://localhost:7101/AltaAgentes/faces/adf.task-flow?adf.tfId=BPMFlow&adf.tfDoc=/WEB-INF/flows/test/BPMFlow.xml&bpmWorklistTaskId=9999](http://www.google.com/url?q=http%3A%2F%2Flocalhost%3A7101%2FAltaAgentes%2Ffaces%2Fadf.task-flow%3Fadf.tfId%3DBPMFlow%26adf.tfDoc%3D%2FWEB-INF%2Fflows%2Ftest%2FBPMFlow.xml%26bpmWorklistTaskId%3D9999&sa=D&sntz=1&usg=AFQjCNFNN2pn3yhlpoA6ZotbcH5Ysoprzg)

**Descripcion**:

Archivos más usados:

-TestTemplate.jspx : utiliza la view del HTPageTemplate

-HTPageTemplate.jspx

-AltaAgente.jsff el fragmento

-AltaAgenteFlow el flujo

-TestTemplate2 jspx

-bpmFlow

**RUTINA**

Después de cada acción en la clase:

-Realizar un Reformat

-Realizar un Organize Imports

Hacer la integración:

AltaAgente->ViewController-> click derecho -> deploy -> JAR(si no se deploya anteriormente agentesADFlib)

**NEGOCIO**

Entidades: las más importantes son Agente, Comisiones y Sucursales

**BASE DE DATOS**

- Cuando se cambia la tabla, se debe cambiar el modelo de negocio.

- Se debe ir a la entidad, click derecho y utilizar Synchronize with Database y se selecciona lo que se modificó ,

Se debe ir a la ViewObject para actualizar la vista, luego a attributes-> add Attribute from Entity y pasar las nuevos campos

**DEFINICIONES**

**PagDef**:

-El pagdef es la relación entre el modelo de negocio (entidades, vistas) y el fragmento, pagina.

depende de los parámetros que ejecuta en el módulo se lo voy a pasar al pagDef

**Data Bindings:** es un archivo donde indica los **pageDef** para cada **TaskFlow** y Fragmentos

**Fixed Outcome**: se agrega a donde se destina el control flow case

Agregar Parametros en TaskFlow

Overview->Parameters

Uso de clases para tipo de datos:

Oracle.jbo.domain.number

Java.lang.boolean

**- Cambiar label**

Utilizar select text Resource

Hacer un replace

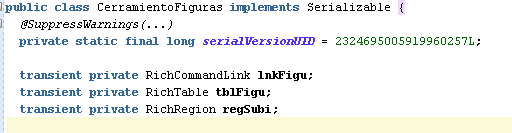
**- Replace in files**

-poner preview para visualizar antes del cambio

- match case distingue mayúsculas y minúsculas.

- match whole Word only , si encuentra la misma palabra con mas caracteres no lo reemplaza.

Transient : es un modificador que sirve para aclarar que no serialice los atributos.



**C**lientListener:

Escucha el lado del cliente, escuchando el evento de javascript

MaximumLength

Para adicionarle al largo se le puede poner + (un numero)

UtilADF.setAtributteBinding(“nombre”,valor)

Setea el valor del atributo del modelo

**LOGICA DE IMPLEMENTACION**

Replicación de una **integración** importa 3 archivos:

-la jspx

-pagDef.xml

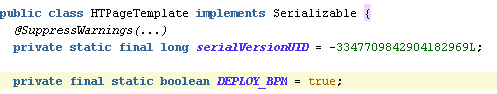
-java asociado a la pagina

Se usa el TestTemplate , TestTemplate2,Tramite, ConsultaAgente

-BpmFlow corre el mediante el HTimplementacion (TestTemplate) y se conecta con SavePointTemplateFlow por el mismo test.

- Para correr el proyecto se corre BPMFLOW

- en caso que no corra el proyecto pasar la variable DEPLOY\_BPM a false de la clase HTPageTemplate.



Cambiar el HTimplementacion por TestTemplate2 en caso que no ejecute la aplicacion

Cada guardado Parcial de Tramite se guarda en la tabla ARP\_SAVEPOINTS

Cuando se agrega un tramite :

Cambiar agregar en la tabla ARP\_TP\_TRAMITES

PANTALLA

Usar el Header Text

Select Text Resource y poner el display value como quiero la columna o texto

Los ítem value del PageDef son los atributos de la View.

VIEWITERATOR

Son como los resultset

getAllinRange es de la clase DCIiteratorBinding.class

TRAMITE.JAVA

ValidarTramiteParticular()

Toma todos los casos de cada tramite

BASE DE DATOS

Para agregar una sucursal

ARP\_SUCURSALES ----🡪 REL\_AGENTES\_SUCURSALES

SUCURSAL\_ID

REL\_AGENTES\_SUCURSALES --🡪 ARP\_AGENTES

AGENTE\_ID

**SUCURSALES**

Variables recibidas:

- modo

- tareaBPM

- usuario

- rol

-sucursal ID

- tramite ID

- solicitud ID

Modo: creada como variable de pageFlowScope de SucursalFlow , luego se la envía a SucursalStop y se filtra por una expression builder , pasa por todos los flow siguientes hasta llegar a domicilio.jsff.

Cuando quiero poner los parámetros primero voy al flow declaro lo que necesito

Luego voy al Stop que quiero pasarle.

Registro con una secuencia Correcta

Crear una secuencia en la base de datos

Crear un trigger para dispararla

Crear la entidad con el id de tipo DBSequence

**- SucursalFlow - SucursalStop.jsff -DomicilioYZonasFlow - DomicilioyZonas.jsff**

**-DomicilioFlow - Domicilio.jsff**