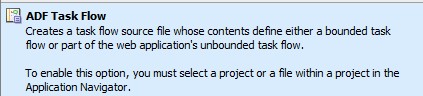
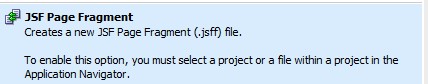
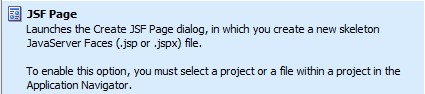
1. Creo un Task Flow



1. Luego Los Fragmentos de las paginas de las cuales quiero crear (.jsff)



1. Luego Creo una pagina (.jpsx)

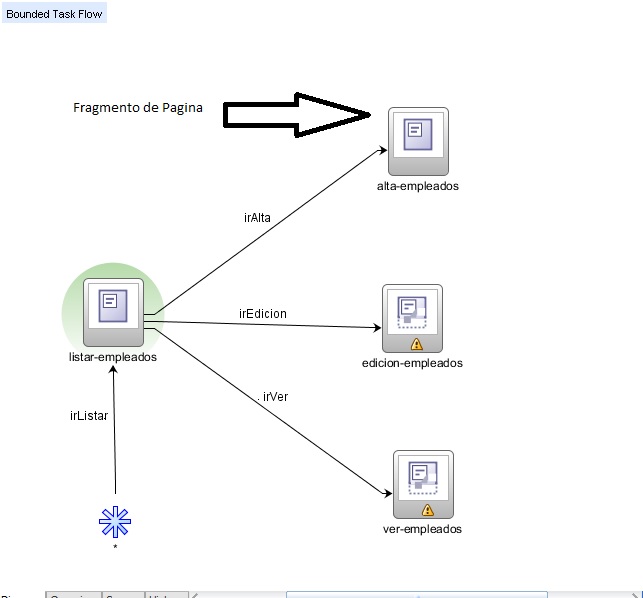


1. Arrastro el Task Flow a la página y lo denomino Región

En El Task Flow

Utilizo la herramienta **Control Flow Case** que son las conexiones entre Fragmentos de Pagina y nombro esa conexión.

El **wildcard Flow rule** es una conexión en la cual todos los fragmentos pueden utilizar.



**FORMULARIO**

PANEL GROUP LAYOUT: es un componente que agrupa otros componentes.

PANEL FORM LAYOUT: es un componente que a agrupa componentes como formulario común.

PANEL HEADER: es un componente que crea un panel con etiquetas para inserción de componentes.

PANEL ACCORDION: es un componente que panel que se minimiza.

PANEL BOX: es un componente que crea un panel con título y es interactivo.

PANEL SPLITTER: es un componente que crea un panel que divide la pantalla en 2 partes.

PANEL TABBED: es un componente que crea un panel con solapas.

PANEL WINDOW: es un componente que crea un panel genérico.

POPUP: es un componente que genera un panel invisible en el cual se puede interactuar.

ROUTER:es como si fuera una funcion IF

Expandir Tabla: **STYLE** -> styleClass: AFStretchWidth, Luego Behavior-> FetchSize: #{bindings.TablaEjemploIterator.estimatedRowCount}

AutoHeightsRows: 0

ningun componente que esta dentro de un for each se puede bindear

1. Utilizar la solapa Structure de Jdeveloper para analizar el árbol de los componentes
2. Crear un “Panel Header” y utilizar las herramientas Panel Form Layout o Panel Group Layout
3. Arrastrar las entidades encapsuladas del Data Control hacia el form.
4. En el Form Principal creo un botón en la solapa Data Controls->Operations->CreateInsert y lo arrastro al form para crear un objeto.
5. En Properties utilizo la propiedad Action para redirigirlo hacia un Fragmento de página.
6. Los botones del formulario deben volver al formulario principal, en caso de error y no poder volver, utilizar la propiedad Inmediate.
7. Para guardar los cambios en la base ya sea Alta, Baja o Modificación debe ser guardados con la Operations Commit.

ACLARACION: **Para realizar un ABM y utilizar operaciones como CreateInserts, Delete, etc. que llamen al PopUp y ejecuten la acción, se deberá realizar en una serie de pasos.**

**CREACION DE GET Y SET DE LOS COMPONENTES**

Selecciono la herramienta -> propiedades-> binding-> edit -> lo asocio al backing bean, y pongo nombre al método

Aclaración:

**Para que los componentes perduren en el ciclo de vida debemos crear una clase JAVA para que cuando se cierre un formulario perdure en el tiempo y morirá cuando se termina la clase**.

PLANTILLA DE ORACLE

Utilizar plantillas para el refresco de la tablas.

**CREACION DE UN ABM CON POPUP**

Link: <https://www.youtube.com/watch?v=-HK8nvnnxyk&feature=em-share_video_user>

Herramientas utilizadas

1) Task Flow

2) Pagina con Plantilla de Oracle

3) View (Task Flow)

4) Table

5) Command Button

6) PopUp

7) Dialog

8) el Managed Bean del Task Flow

9) Clase Java

10) Uso de Page Definition

11) Propiedad PopUpFetchListener

12) BindingContainer y OperationsBindings