

DESARROLLO WEB INTEGRADO



Universidad
Tecnológica
del Perú

Tema de la sesión

JFS - DataTables

Conocimientos previos

- *Dialogamos con la participación de todos los alumnos acerca del tema JSF - DataTables*



Dudas de la clase anterior

Alguna duda o consulta acerca del tema de la sesión anterior: JSF – Tag Library



Logro de aprendizaje

Al finalizar la sesión, el estudiante comprende los conceptos básicos de arquitecturas Java SE y Java EE y los elementos de estas arquitecturas.



Utilidad

- *Dialogamos con los alumnos acerca de la Utilidad de JSF - DataTables*

Secuencia y explicación

- JSF – Lenguaje EL
- JSF - DataTables

JSF – Expression Languages (EL)

- Basado en el EL incluido en JSP 2.0
- EL proporciona un conjunto de objetos implícitos que permiten al desarrollador acceder a parámetros de la petición, cabeceras HTTP, etc.
- Se puede utilizar EL para sentencias lógicas y matemáticas, y también es posible mezclar valores literales con expresiones

JSF – Expression Languages (EL)

- **Diferencias entre JSF EL y JSP 2.0 EL:**
 - JSF utiliza el signo (#) para marcar el principio de una expresión, en vez del símbolo (\$).
 - Las expresiones JSF pueden ser también de dos vías. En otras palabras pueden o recuperar el valor de una propiedad o actualizarla.
 - JSF EL también te permite referenciar métodos de objetos.
 - Algunas características específicas de JSP no están disponibles, como el ámbito de página.
 - Las funciones JSP EL no están soportadas oficialmente

JSF – Expression Languages (EL)

- Se puede utilizar JSF EL para enlazar componentes a objetos que exponen propiedades javabean, colecciones, y tipos de datos sencillos.
- EL puede también ser utilizado para referenciar métodos y crear sentencias lógicas o numéricas. También soporta sentencias anidadas

Ejemplo	Descripción
<code># { miBean.value }</code>	Retorna la propiedad value del objeto almacenado bajo la clave miBean
<code># { miBean['value'] }</code>	Es lo mismo que <code># { miBean.value }</code>
<code># { miArrayList[5] }</code>	Retorna el quinto elemento de una lista almacenada bajo la clave miArrayList.
Estas <code># { (user.balance > 100) ? 'logueado' : 'no logueado' }</code>	Retorna la cadena “Estás logueado” si la propiedad balance del objeto almacenado bajo la clave user es mayor que 100. Retorna “Estás no logueado” en otro caso.

JavaServer Faces – Data Table

- Las tablas HTML son muy usadas a la hora de estructurar y mostrar por pantalla los componentes de una aplicación web. JSF le permite organizar dichos elementos mediante la etiqueta `h:dataTable`

JavaServer Faces – Data Tables

La etiqueta h:dataTable

La etiqueta h:dataTable itera sobre los datos contenidos en la tabla, para formar la tabla HTML. A continuación se muestra un ejemplo de su uso:

```
<h:dataTable value="#{bean.elementos}" var="elemento">

    <h:column>
        <%-- componentes de la columna izquierda --%>
        <h:outputText value="#{elemento}"/>
    </h:column>

    <h:column>
        <%-- siguiente columna de componentes --%>
        <h:outputText value="#{elemento}"/>
    </h:column>
    <%-- mas columnas--%>

</h:dataTable>
```

JavaServer Faces – Data Tables

El atributo `value` representa los datos sobre los que `h:dataTable` itera; estos datos pueden ser:

- Un array
- Una instancia de `java.util.List`
- Una instancia de `java.sql.ResultSet`
- Una instancia de `javax.servlet.jsp.jstl.sql.Result`
- Una instancia de `javax.faces.model.DataModel`

El identificador que llama a cada elemento es especificado con el atributo `var`.

JavaServer Faces – Data Tables

Cada columna puede contener un número ilimitado de componentes como elementos adicionales.

Atributos para la etiqueta `h:dataTable`:

Atributos	Descripción
<code>bgcolor</code>	Color de fondo de la tabla
<code>border</code>	Anchura del borde de la tabla
<code>cellpadding</code>	Espacio de las celdas de la tabla
<code>cellspacing</code>	Espacio entre las celdas de la tabla
<code>columnClasses</code>	Lista separada por comas de las clases CSS para las columnas

JavaServer Faces – Data Tables

Atributos	Descripción
first	Índice de la primera fila mostrada en la tabla
footerClass	Clase CSS para el pie de tabla
headerClass	Clase CSS para la cabecera de tabla
rowClasses	Lista separada por comas de las clases CSS para las filas
rules	Especificación para las líneas dibujadas entre las celdas. Los valores permitidos son: <i>groups</i> , <i>rows</i> , <i>columns</i> , <i>all</i>
var	Nombre de la variable creada por la tabla que representa el valor del elemento actual

JavaServer Faces – Data Tables

Atributos	Descripción
binding, id, rendered, style- Class, value	Atributos básicos
dir, lang, style, title, width	Atributos HTML 4.0
onclick, ondblclick, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup	Atributos de eventos DHTML

Activar W
Ve a Configu

Cabeceras y pie de tabla

En el caso de que desee decorar mas su tabla o aclarar el contenido de la misma, puede añadir los elementos de cabecera (header) y pie (footer) de tabla. Estos elementos son facetas adicionales a la tabla y cuyo uso se muestra a continuación:

```
<h:dataTable>
  ...
  <h:column>
    <f:facet name="header">
      <%-- La cabecera del componente va aquí--%>
    </f:facet>

    <%-- La columna va aquí--%>

    <f:facet name="footer">
      <%-- El pie de la columna va aquí--%>
    </f:facet>
  </h:column>
  ...
</h:dataTable>
```

Componentes JSF incluidos en celda tabla

Se puede ubicar cualquier componente JSF dentro de una columna de tabla. Su uso no difiere del visto hasta ahora, declarando los componentes dentro de las columnas de la tabla.

Editando celdas

Para editar la celda de una tabla, basta con colocar un componente de entrada en las celdas que quiere que sean editadas.

Estilos para filas y columnas

h:dataTable posee atributos para las clases CSS sobre los siguientes elementos:

- La tabla como un todo (styleClass)
- Cabeceras y pie de tablas (headerClass y footerClass)
- columnas individuales (columnClasses)
- filas individuales (rowClasses)

Técnicas de scroll

Usar una barra de scroll es una técnica muy simple. Para ello se usa la directiva div. Su sintaxis es la siguiente:

```
<div style="overflow:auto; width:100%; height:200px">  
  <h:dataTable...>  
    <h:column>  
      ...  
    </h:column>  
    ...  
  </h:dataTable>  
</div>
```

Ejemplos de la etiqueta h:dataTable

Una simple tabla con nombres

Jose,	<i>Jose</i>	<i>Jose</i>
Manuel,	<i>Manuel</i>	<i>Manuel</i>
Andres,	<i>Andres</i>	<i>Andres</i>
Carlos,	<i>Carlos</i>	<i>Carlos</i>

Una simple tabla con componentes JSF

Numeros	Campo Texto	Boton	Checkbox	Enlaces	radio boton	listbox
1	<input type="text" value="1"/>	<input type="button" value="1"/>	<input type="checkbox"/>	1	<input checked="" type="radio"/> Si <input type="radio"/> no	<div>siempre</div> <div>a menudo</div> <div>alguna vez</div>
2	<input type="text" value="2"/>	<input type="button" value="2"/>	<input type="checkbox"/>	2	<input type="radio"/> Si <input checked="" type="radio"/> no	<div>siempre</div> <div>a menudo</div> <div>alguna vez</div>
3	<input type="text" value="3"/>	<input type="button" value="3"/>	<input type="checkbox"/>	3	<input type="radio"/> Si <input checked="" type="radio"/> no	<div>siempre</div> <div>a menudo</div> <div>alguna vez</div>
4	<input type="text" value="4"/>	<input type="button" value="4"/>	<input type="checkbox"/>	4	<input checked="" type="radio"/> Si <input type="radio"/> no	<div>a menudo</div> <div>alguna vez</div> <div>nunca</div>
5	<input type="text" value="5"/>	<input type="button" value="5"/>	<input type="checkbox"/>	5	<input checked="" type="radio"/> Si <input type="radio"/> no	<div>siempre</div> <div>a menudo</div> <div>alguna vez</div>

Espacio práctico autónomo

- *Realizamos la actividad que se encuentra en la plataforma siguiendo las indicaciones respectivas.*

Resumen

- Las arquitecturas de las soluciones de Java ES tienen dos aspectos: una arquitectura lógica y una arquitectura de implementación.
- La arquitectura Java EE es framework de desarrollo de software libre para construir aplicaciones de internet en Java. Integra tecnologías como AJAX y XML, JavaServer Faces, Java Persistence e Enterprise Java Beans, entre otras.

Bibliografía

- *Rafael Luis Granados La Paz. (2015). Desarrollo de aplicaciones web en el entorno servidor (UF1844).*
- *Gil Vera, Juan Carlos. Frameworks para el desarrollo de prototipos WEB: un caso de aplicación.*
- *Rafael Luis Granados La Paz. (2015). Desarrollo de aplicaciones web en el entorno servidor (UF1844).*





**Universidad
Tecnológica
del Perú**