

PRÁCTICA N° 1

Temas

- Implementación de Servlets.
- Recolección de información a través de páginas HTML, su procesamiento a través de *servlets* y construcción de la respuesta.

Cree un proyecto Web dinámico de nombre **Practica1Web25**, indicando en el combo "Dynamic Web module version" la **versión 2.5**. Asigne como nombre del *context root* **misServlets** (será el nombre de la aplicación Web).

De esta manera ha definido una aplicación Web propia y todas las componentes Web java (*servlets*, *jsp*, *filtros*, *listeners*, etc.) que construya van a formar parte de dicha aplicación junto con los otros recursos web como páginas HTML, imágenes, etc.

1. Antes de escribir su primer *servlet*, analice la estructura jerárquica que muestra las componentes del proyecto. Despliegue la carpeta **WEB-INF** que está dentro del proyecto Web. Analice su estructura y en particular el archivo **web.xml**.

Cree un proyecto Web dinámico de nombre **Practica1Web40**, indicando en el combo "Dynamic Web module version" la **versión 4.0**.

2. Escriba un *servlet* llamado **Bienvenida** que tome el nombre del usuario del requerimiento http que recibe como parámetro y devuelve una página HTML que dice "Bienvenido " + nombre (donde nombre es el dato recibe como entrada).

La clase del *servlet* debe residir en el paquete **misservlets**. Sobreescriba solo el método **doGet()**.

Debe escribir una página HTML llamada **saludo.html**, que contenga un formulario con un campo de entrada de texto, en donde el usuario escribirá su nombre y un botón de tipo *submit* para enviar los datos del formulario al *servlet* **Bienvenida** para su procesamiento.

- a) Ejecute el *servlet* y pruebe desde el navegador.
 - b) Modifique el archivo **saludo.html** y utilice el método "POST" para enviar los datos del formulario al *servlet* **Bienvenida**. ¿Necesita hacer algún cambio en el código del *servlet*?
 - c) Analice nuevamente las carpetas WEB-INF\classes
 - d) Pruebe acceder a su página **saludo.html** desde el navegador de su compañero. ¿Funciona? ¿Por qué?
 - e) ¿Cuáles son las diferencias entre: la estructura del proyecto que está desarrollando con el IDE y la estructura de la aplicación Web que está ejecutándose en el servidor? Confeccione un diagrama jerárquico de cada una de las estructuras y muestre la relación de contenido que existe entre ellos.
3. Escriba un *servlet* llamado **ContadorVisitas** que mantiene actualizada la cantidad de requerimientos o visitas que recibe. Cada vez que recibe un requerimiento incrementa la cantidad y devuelve una página HTML que dice: "Este *servlet* lo visitaron: X usuario/s" (X es la cantidad de visitas). La clase del *servlet* debe residir en el paquete **misservlets**. No utilice variables de clase para mantener la cantidad de visitas.

Nota: ejecute el *servlet* desde el navegador Web. Mantenga abiertas al menos tres instancias del navegador. Ejecute el *servlet* varias veces desde cada una de las ventanas. Ejecútelo desde el navegador de su compañero.

Si no llega a obtener los resultados esperados, analice la configuración del navegador (uso de archivos temporales).

4. Escriba un *servlet* llamado **LoginUsr** que tome del requerimiento http los datos de un usuario: identificación de usuario y contraseña y los valide. Si los datos son válidos devuelve como respuesta una página html de éxito en donde se informa que se trata de un usuario válido; en otro caso devuelve una página html de error en donde se informa que los datos ingresados no son válidos y un *link* a la página de **login.html**, para que el usuario vuelva a ingresar su login. La clase del *servlet* debe residir en el paquete **misservlets**.

Para validar los datos defina e inicialice dos variables de instancia del *servlet* (de tipo Vector o arreglo de Strings) que contengan nombres de usuarios y sus respectivas contraseñas.

En este ejercicio, debe escribir una página HTML llamada **login.html**, que contenga al menos: un título y un formulario con dos campos de texto con las etiquetas **Usuario** y **Contraseña** en donde el usuario ingresará su login y un **botón** de tipo *submit* para enviar los datos ingresados. Los datos del formulario se envían al *servlet* **LoginUsr** para su procesamiento. Ejecute el *servlet* desde el navegador. Abra más de un navegador.

5. Escriba un *servlet* llamado **Encuesta** que toma del requerimiento http datos acerca de los gustos de los usuarios por las mascotas, incrementa un total de acuerdo al tipo de mascota elegida y muestra una página de resultados de la encuesta. La clase del *servlet* debe residir en el paquete **misservlets**.

En este ejercicio, debe escribir una página HTML llamada **mascotas.html**, que contenga al menos: un título y un formulario con al menos 6 objetos de tipo *check-box* cuya etiqueta tenga tipos de mascotas (perro, gato, hamster, tortuga, etc.) seleccionables y un **botón** de tipo *submit* para enviar los datos ingresados. El nombre de los *check-boxes* debe ser el mismo (por ejemplo, mascotas). Los datos del formulario se envían al *servlet* **Encuesta** para su procesamiento.

La página que devuelve el *servlet* debe contener:

- un título.
- una tabla HTML con dos columnas (mascota, cantidad de votos) y tantas filas como tipos de animales.
- un texto indicando el animal más votado y el porcentaje de votos sobre el total.
- un *link* a la página de **mascotas.html**.

a) Ejecute el *servlet* desde el navegador. Abra más de un navegador.

Modifique la página **mascotas.html** reemplazando los *check-boxes* por una lista de selección múltiple HTML con el mismo nombre que le asignó a los *check-boxes*.

b) ¿Necesita modificar el código del *servlet*?