

Ejercicios de Beans y lenguaje de expresiones

Índice

| | |
|------------------------------|---|
| 1 Bean cronómetro..... | 2 |
| 2 Datos de los usuarios..... | 2 |

1. Bean cronómetro

Crear un *bean* denominado **beans.CronoBean** que sirva como "cronómetro". Cuando el *bean* se inicialice, debe guardar internamente el momento de su creación. El *bean* tendrá una propiedad `segundos`, que devolverá el número de segundos transcurridos desde su creación. Para calcular tiempos, se puede usar el método `System.currentTimeMillis()`, que devuelve el número de milisegundos transcurridos entre el instante actual y el 1/1/1970.

Probar el bean en una página llamada **cronoini.jsp** que lo inicialice (mostrando un mensaje HTML que indique que se está inicializando) y muestre el número de segundos transcurridos. Acceder al *bean* desde otra página **crono2.jsp** que muestre el mismo temporizador (¿qué ámbito debe tener el *bean* para funcionar correctamente?).

2. Datos de los usuarios

En la plantilla se tiene implementado el bean **beans.UsuarioBean** explicado en teoría, con los campos que se guardan del usuario. Se tiene el formulario **index_usuarios.html** que sirve para recoger los datos y enviárselos a la página **main.jsp** que se encargará de crear el bean con los datos y mostrar un mensaje de bienvenida. La página también deberá tener un enlace *Ver datos* para mostrar los datos del usuario, que apunta a la página **datosusuarios.jsp**.

Se pide:

- Implementar la página **main.jsp** para que cree un bean (si no existe ya), actualice los valores correspondientes (número de visitas y fecha de la última visita), y saque todos los datos por pantalla. Modificar el ámbito del bean para que tenga validez suficiente como para verse desde diferentes páginas (pero que sean beans distintos para usuarios distintos).
- Implementar la página **datosusuarios.jsp** para que muestre los datos de la persona utilizando lenguaje de expresiones. Para ello deberéis volver a cargar el *bean* en esta página, utilizando `jsp:useBean` antes de mostrar sus datos.
- Indicad, mediante "SI" o "NO" en la página, y utilizando lenguaje de expresiones, si el usuario ha superado las 10 visitas.

AYUDA: se admite el operador `'?:'` en lenguaje de expresiones. Si tenemos las visitas en la variable *visitas*, podríamos poner algo como:

```
${visitas > 10?'SI':'NO'}
```

