

# PROYECTO JSF

1.- Crear un proyecto Java llamado LibreriaJSF.

2.- Crear los siguientes paquetes:

- app.web
- app.modelo
- app.negocio
- app.persistencia

3.- En el paquete app.modelo crear la clase encapsulada Libro con las siguientes propiedades:

- int ID
- String titulo
- String autor
- String editorial
- String isbn
- int publicacion
- double precio
- String descripcion

4.- En el paquete app.persistencia crear una interface llamada ItfzLibrosDao con los siguientes métodos:

- public boolean altaLibro(Libro libro)
- public boolean eliminarLibro(int id)
- public List<Libro> consultarTodos()
- public Libro consultarISBN(String isbn)
- public List<Libro> consultarTitulo(String titulo)
- public boolean modificarPrecio(String isbn, double precio)

5.- En el mismo paquete app.persistencia crear la clase LibrosDAO que implemente la interface anterior utilizando Hibernate.

- public boolean altaLibro(Libro libro); Debe crear un nuevo registro en la tabla con los datos del libro recibido como argumento.

- `public boolean eliminarLibro(int id);` Debe eliminar el libro de la tabla que coincida con el identificador como argumento.
- `public List<Libro> consultarTodos();` Devuelve una lista con todos los libros de la tabla.
- `public Libro consultarISBN(String isbn);` Devuelve una instancia del Libro localizado por su isbn.
- `public List<Libro> consultarTitulo(String titulo);` El argumento titulo recibido en el método puede que no sea el nombre completo. Se trata de devolver una lista con todos los libros que contienen ese dato como parte de su titulo. Utilizar el operador LIKE.
- `public boolean modificarPrecio(String isbn, double precio);` El método recibe el isbn del libro a actualizar, así como el nuevo precio.

6.- En la capa `app.negocio` crear una interface `ItfzGestionLibreria` con los mismos métodos que la interface anterior.

7.- En el mismo paquete crear la clase `GestionLibreria` que implemente la interface anterior. Todos los métodos deben acceder a la capa de persistencia a través de la interface `ItfzLibrosDAO` con una managed property.

8.- En la capa web crear:

- `CarritoBean` será un `ManagedBean` con ámbito de sesión
- `LibrosBean` será otro `ManagedBean` con ámbito de petición

9.- Se pide crear una aplicación que cumpla los siguientes requisitos:

- CRUD (Create, Read, Update, Delete)
- Comprar libros seleccionando el isbn y cantidad
- Páginas con el mismo aspecto utilizando plantillas
- I18n
- Validación de formularios y errores personalizados