**MÓDULO 10**

**EJERCICIOS RESUELTOS BASE DE DATOS**

**PREGUNTA 1**

Crea una base de datos en Mysql con el nombre biblioteca\_online, a partir del script Biblioteca\_online.sql ubicado en el campus. Esta base de datos contendrá las siguientes tablas:

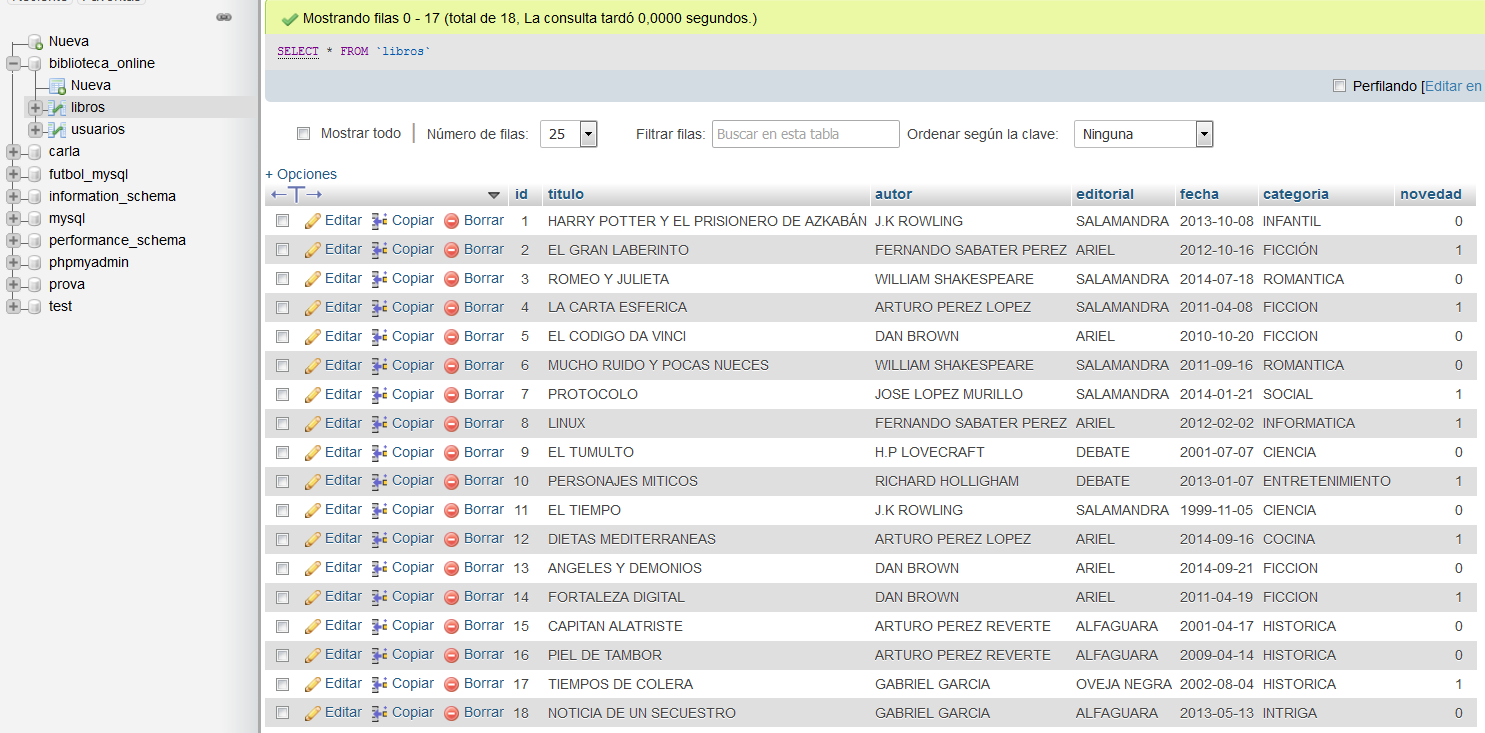
* Usuarios
* Libros

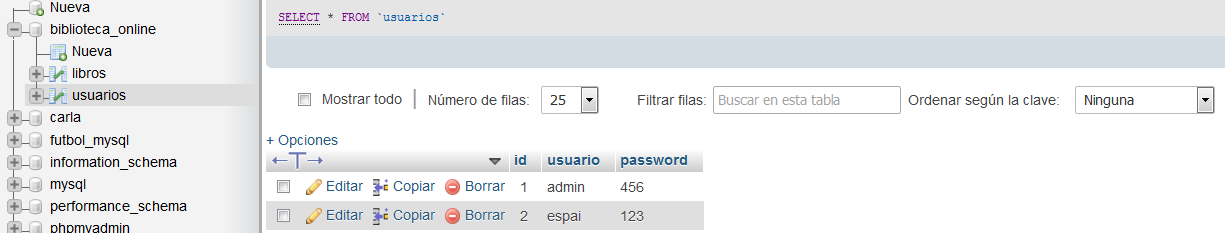
La tabla Usuarios tendrá los campos usuario y contraseña.

La tabla Libros tendrá los siguientes campos:

* Id
* Titulo
* Autor
* Editorial
* Fecha
* Categoría

Esta base de datos contiene los mismos datos que existen en el fichero “libros.txt”.

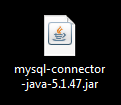
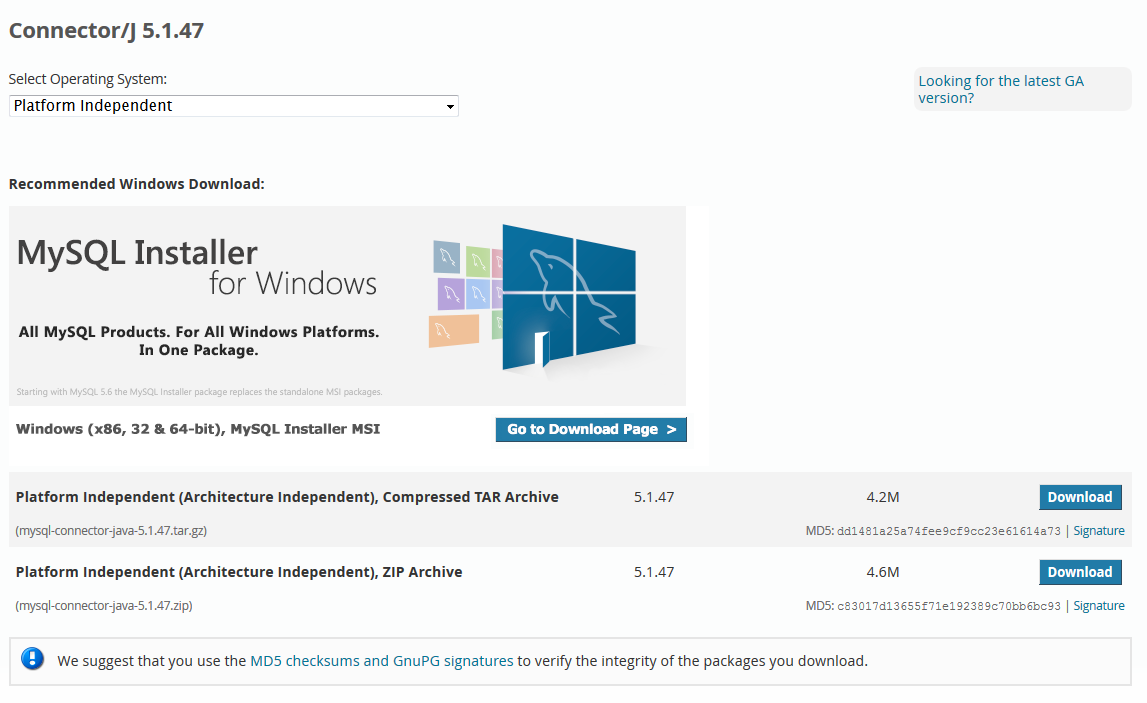


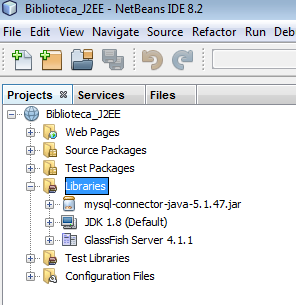
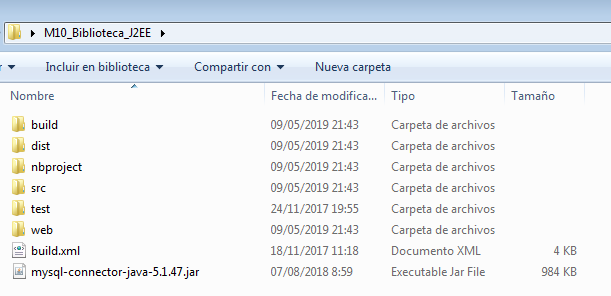


**PREGUNTA 2**

Descarga el connector java-mysql versión 5.1 e inserta el driver jar en el proyecto J2EE con el fin de crear una conexión JDBC con la base de datos biblioteca\_online.

<https://dev.mysql.com/downloads/connector/j/5.1.html>

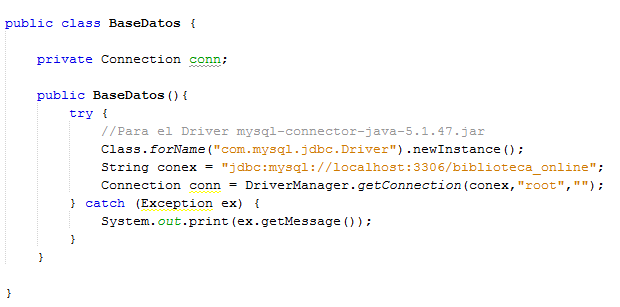




**PREGUNTA 3**

Crea una clase que llamaremos “BaseDatos.java” que encapsulará todo el trabajo que hagamos de conexión, inserción, modificación, borrado y consulta en la base de datos mysql dentro de nuestra aplicación.

Debe tener un objeto miembro de tipo connection, y en su constructor debe de ser inicializado para que apunte a la base de datos biblioteca\_online



**PREGUNTA 4**

Modifica GestorBibliotecaServlet para que el acceso a la aplicación se haga de acuerdo con los usuarios introducidos en la base datos mysql. Realiza en la clase BaseDatos.java una función similar a la siguiente:

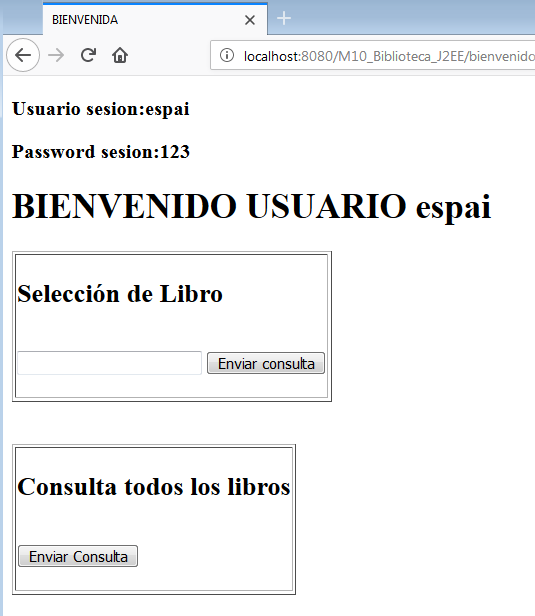
public boolean compruebaUsuario(String usuario, String password)





**PREGUNTA 5**

Modifica bienvenido.jsp para que ofrezca la posibilidad de mostrar la lista de todos los libros existentes. El nuevo formulario debe llamar a un nuevo Servlet GestionLibrosServlet, que se pedirá crear en la siguiente pregunta.



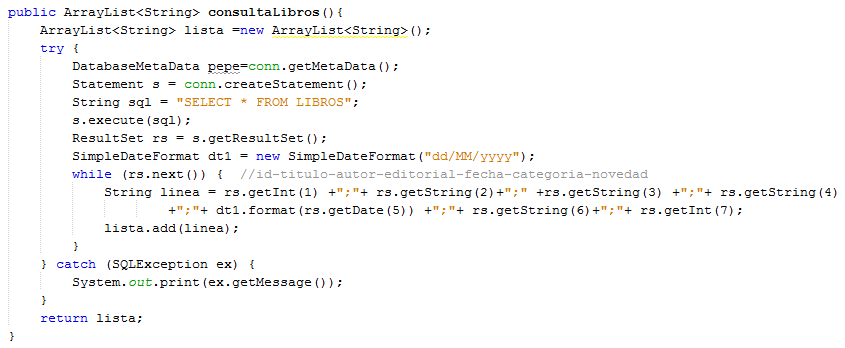
  
**PREGUNTA 6**

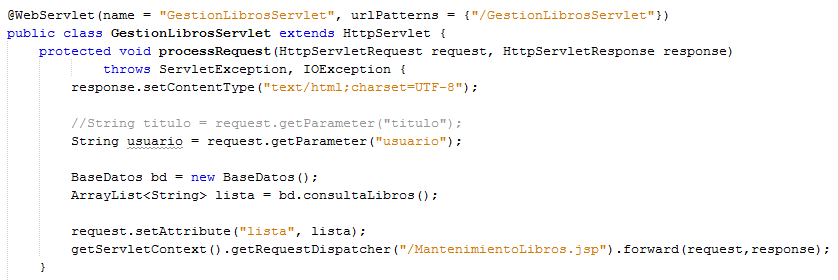
Crea el servlet GestionLibrosServlet el cual procesará la consulta que mostrará todos los libros de la biblioteca y la redirigirá hacia el jsp MantenimientoLibros.jsp. Realiza en la clase BaseDatos.java una función similar a la siguiente:

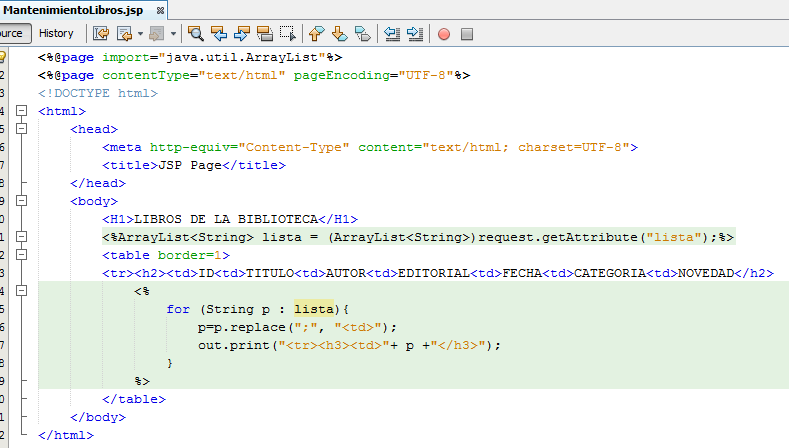
public ArrayList<String> consultaLibros()

La función consultaLibros debe retornar la consulta de los libros en un ArrayList, de manera que cada elemento sea una cadena con todos campos de un libro separados por punto y coma.

El servlet debe pasarle el ArrayList de la consulta a MantenimientoLibros.jsp como un atributo del request y finalmente MantenimientoLibros.jsp debe mostrar en una tabla html los campos de todos los libros.





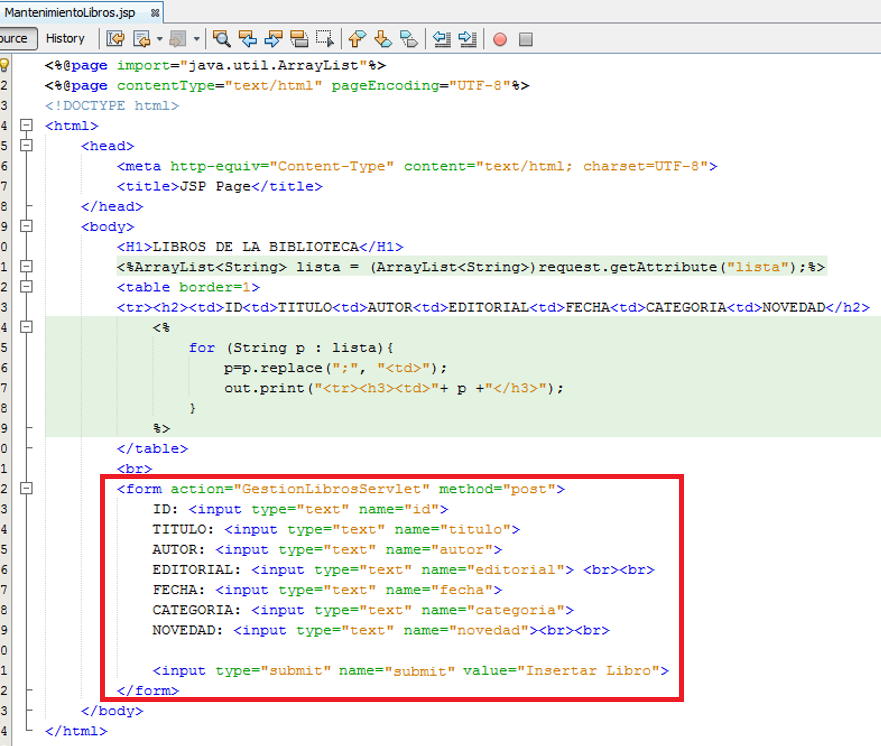




**PREGUNTA 7**

En MantenimientoLibros.jsp crea un formulario que permita introducir nuevos libros en la base de datos. Debe proporcionar los campos: Id, Titulo, Autor, Editorial, Fecha, Categoría y Novedad. El servlet GestionLibrosServlet será el action del formulario.

NOTA: Las diferentes llamadas al servlet GestionLibrosServlet se distinguirán usando el name y el value del botón submit.



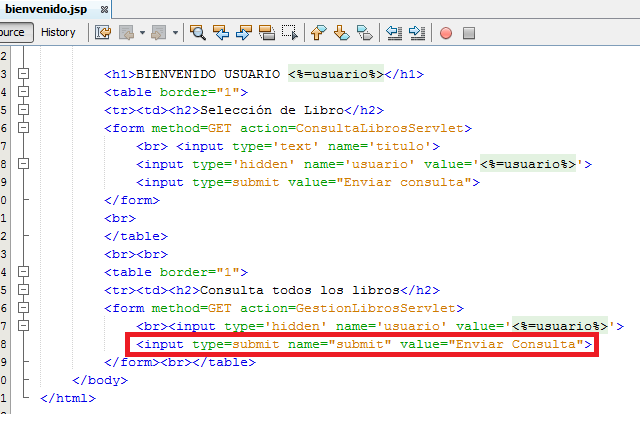


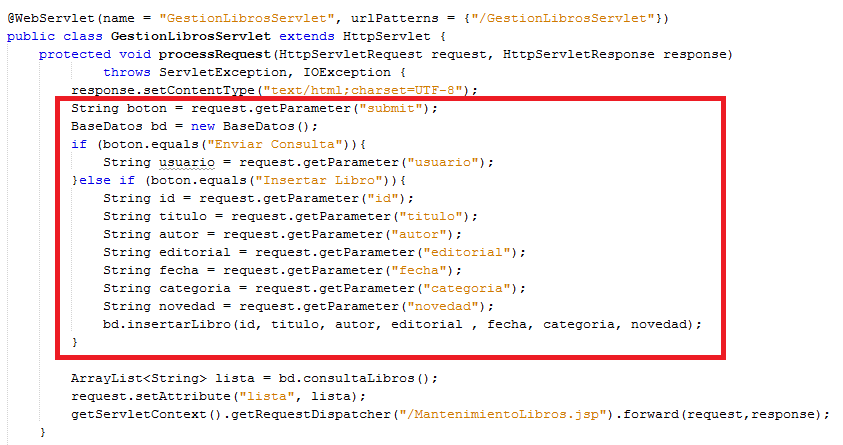
**PREGUNTA 8**

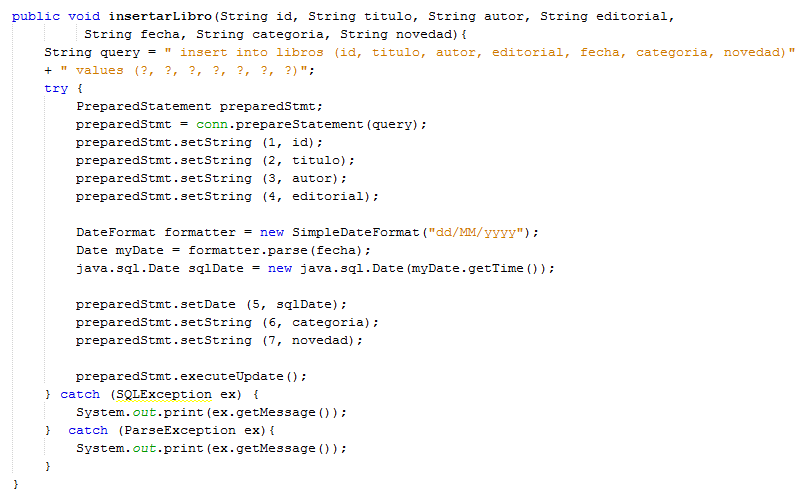
Implementa la acción de insertar un libro en la base de datos desde el formulario MantenimientoLibros.jsp, utilizando para ello el servlet GestionLibrosServlet. Realiza en la clase BaseDatos.java una función similar a la siguiente:

public void insertarLibro(String id, String titulo, String autor, String editorial, String fecha, String categoria, String novedad)

NOTA: Distingue las diferentes llamadas al servlet GestionLibrosServlet mediante el value del botón submit respectivo.







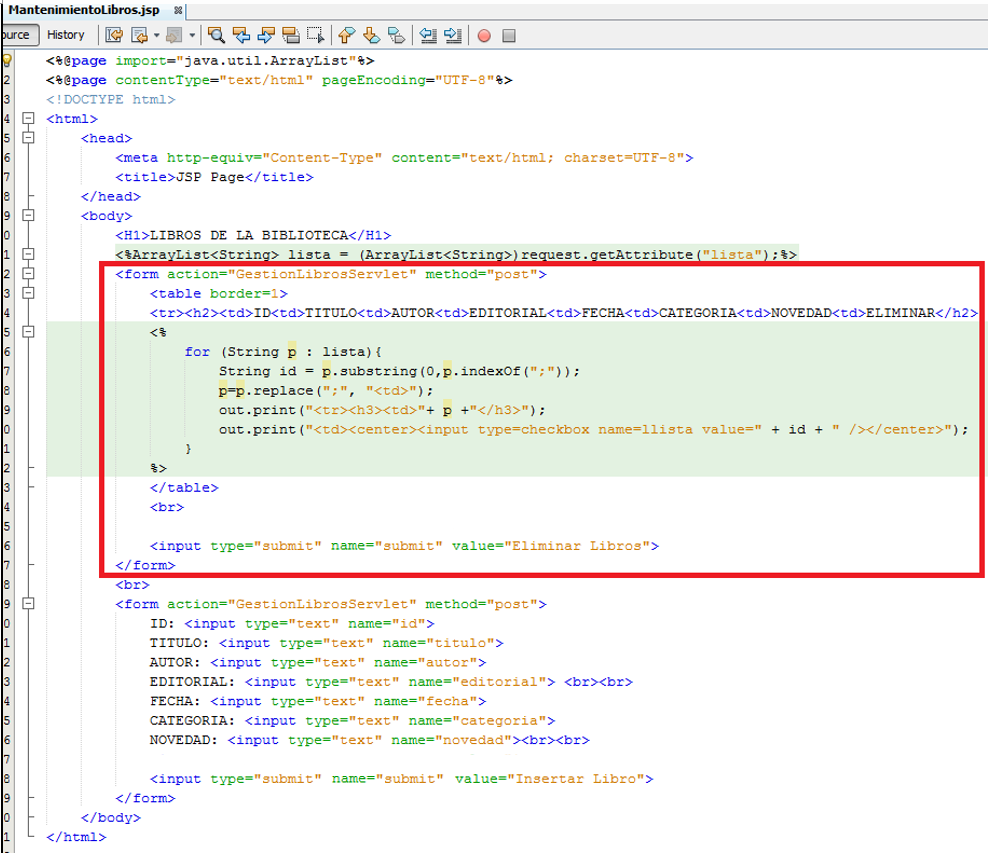




**PREGUNTA 9**

Añade a la página anterior la posibilidad de eliminar libros. En MantenimientoLibros.jsp agrega una nueva columna “Eliminar” formada por checkboxes a la derecha de cada entrada de la lista. Cada checkbox servirá para la posterior selección y eliminación de un libro de la base de datos.





**PREGUNTA 10**

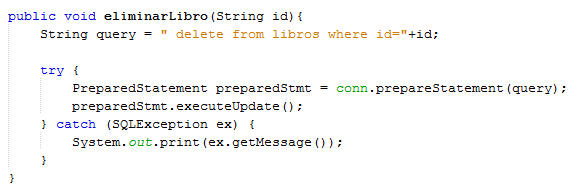
Crea el código necesario en el servlet GestionLibrosServlet que permita eliminar los libros seleccionados. Después del proceso de eliminación, se debe redirigir al usuario de nuevo al listado de MantenimientoLibros.jsp, para que se vean los cambios realizados.

Realiza en la clase BaseDatos.java una función similar a la siguiente:

public void eliminarLibro (String id)



En la clase BaseDatos.java:







**PREGUNTA 11**

Vamos a permitir también la edición de los libros ya existentes. Añade una nueva columna “Modificar” de checkboxes a la derecha de cada fila de libros, y un nuevo botón “Recuperar Libro”. Al seleccionar un libro y clicar en el nuevo botón, el formulario de MantenimientoLibros.jps se llenará con la información del libro a editar. Realiza en la clase BaseDatos.java una función similar a la siguiente:

public String recuperarLibro (String id)

Una vez estén los datos del libro en el formulario, el usuario tendrá la opción de modificarlos. Podrá apretar el botón si quiere guardar las modificaciones, o “cancelar” si desea descartar los cambios. Tanto un botón como otro, después de hacer su función deberán llevarnos de nuevo al listado de libros (GestionLibrosServlet) para que veamos los cambios que se han realizado. Realiza en la clase BaseDatos.java una función similar a la siguiente:

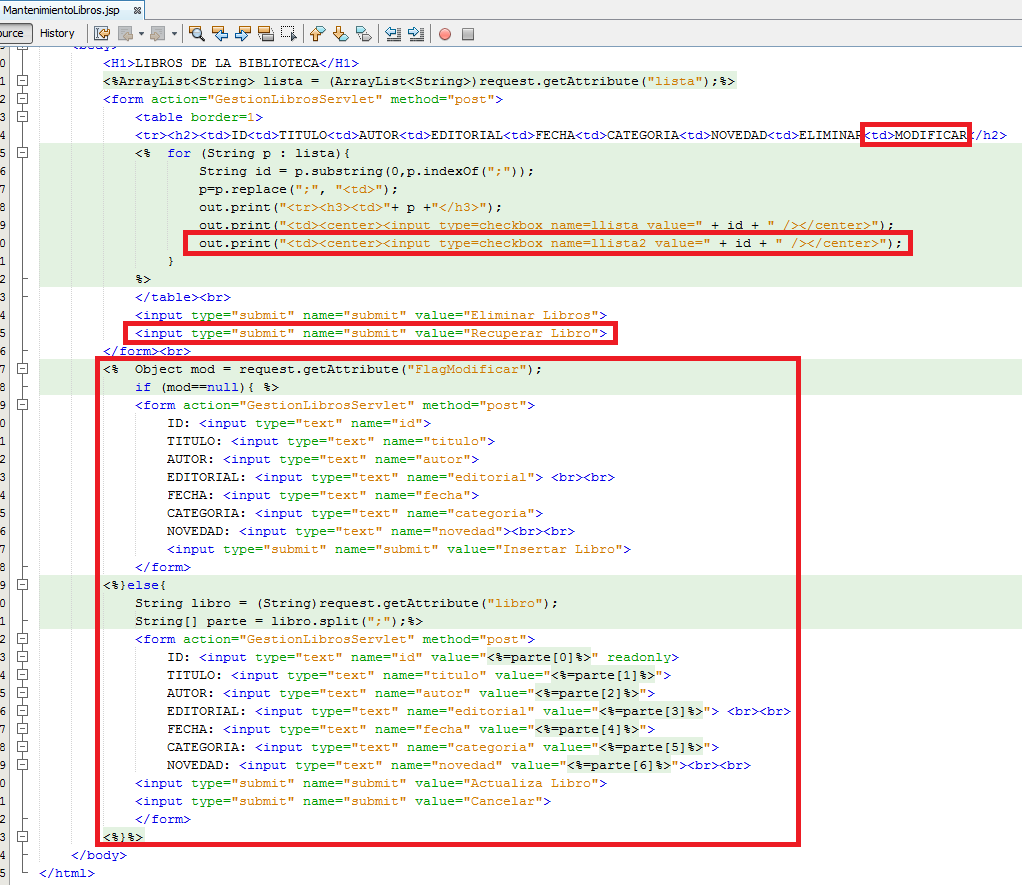
public void modificarLibro(String id, String titulo, String autor, String editorial,

String fecha, String categoria, String novedad){

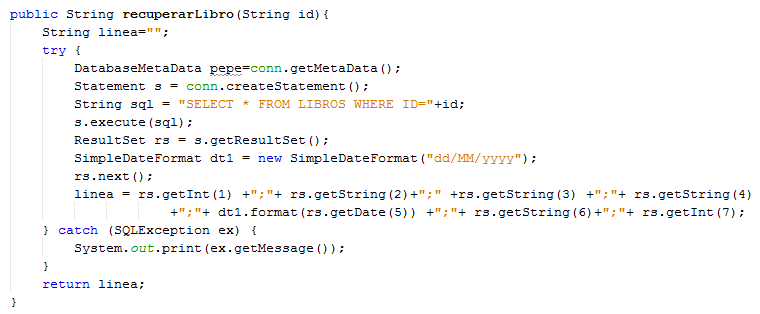


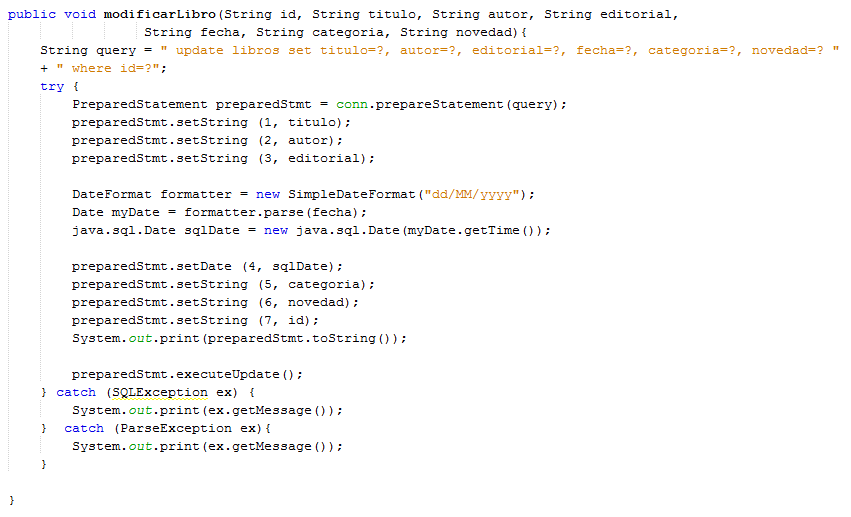












**PREGUNTA 12**

Modifica el servlet ConsultaLibrosServlet para que a partir de ahora, busque en la base de datos los libros consultados.

