LABO IV

Acceso a Datos.

* **Librería JDBC(Java Database Connectivity)**

JDBC es una API de la biblioteca estándar.

API = Interfaz de programación de Aplicaciones.

Realiza tareas de comunicación con distintas bases de datos.

Las clases e interfaces se encuentran en el paquete java.sql

Cada implementación se denomina driver, y es especifica para un motor o versión

Consiste en un archivo jar.

Permite establecer conexiones con las db .

Permite ejecutar consultas sin usar herramientas administrativas propias.

* **Clase DriverManager**

Permite establecer conexiones con las db .

Se requiere cargar previamente en memoria el driver, por única vez.

Class.forName(“com.microsoft.sqlserver.jdbc.SQLServerDriver”);

Para establecer una conexión se ejecuta el método

getConnection de la clase DriverManager.

DriverManager. getConnection(“url”,”usuario”,”contraseña”);

Retorna un objeto Connection ya instanciado y conectado.

En caso de fallar, se lanza una SQLException.

El Objeto Connection conn; permite crear objetos Statement y PreparedStatement.

conn.createStatement();

conn.prepareStatement(sql);

Luego de ser usada debe ser cerrada conn.close();

El método createStatement retorna una instancia Statement lista para ejecutar consultas.

Statement st = conn.createStatement();

st.executeUpdate(sql); //CONSULTAS QUE NO DEVUELVEN FILAS(INSERT, UPDATE, DELETE)

ResultSet rs = sc.executeQuery(sql); // CONSULTAS DEVUELVE FILAS(CONSULTAS SELECT)

Se avanza a una fila siguiente con el método next()

Cuando se supera la ultima fila, next() devuelve false;

rs.close();

rs.getInt() //Cuando una columna contiene números enteros

rs.getString() //Cuando una columna contiene cadenas de texto

rs.getBoolean() //Cuando una columna contiene valores bit booleanos

Reciben un numero del orden de la columna o su nombre.

prepareStatement = puede ser ejecutado multiples veces.

Puede recibir parámetros (?,?,?);

Ps.setInt(1,155)

Ps.setString(2,”Leo Messi”);

**PATRON MVC**

Consiste en una solución bien probada a un problema común.

Se basa en el concepto de separación de código en capas.

Vista: Presentación de los datos.

Controlador: atender las peticiones y componentes para la toma de decisiones.

Modelo: Contempla la lógica de negocio y los datos asociados con la aplicación.

**que pasa si me conecto a un servidor pero está apagado?**

El servidor está apagado, la conexión no se puede establecer.

El metodo getConection va a arrojar una excepción, por eso debemos ejecutarlo dentro de un try.

Para capturar la excepción lo hacemos desde el bloque catch.

**que es un dto y para que sirve?**

Los DTO(Data transfer object) son un tipo de objetos que sirven únicamente para transportar datos.

EL DTO contiene las propiedades del objeto.

Datos que pueden tener su origen en una o más entidades de información.

Son utilizados para mostrar determinada información.

**que es un driver jdbc?**

JDBC es una API de la biblioteca estándar.

Realiza tareas de comunicación con distintas bases de datos.

Las clases e interfaces se encuentran en el paquete java.sql

Cada implementación se denomina driver, y es especifica para un motor o versión

Permite establecer conexiones con las db .

Permite ejecutar consultas sin usar herramientas administrativas propias.

**que linea tengo que escibir si quiero sacar la segunda columna de un resultset sabiendo que es de tipo float?**

rs.getFloat(2)