Acceso a Datos

JDBC

Es una API utilizada para acceder a una base de datos

Significa Java DataBase Conectivity

Forma parte del paquete java.sql

Página con los drivers de los distintos DBMS: http://developers.sun.com/product/jdbc/drivers

Conexión con la base de datos

Para poder manipular datos de una base de datos se necesita crear una conexión

JDBC contiene una interfaz denominada Connection con este fin

Necesita la cooperación de un driver específico de la base de datos al cual estará accediendo

El driver necesita ser registrado para su posterior uso

String dbDriver = "com.mysql.jdbc.Driver";

Ejemplo:

```
String dbConnString = "jdbc:mysql://localhost/cursojava";

String dbUser = "root";

String dbPassword = "";

Class.forName(dbDriver).newInstance();
```

Connection laConexion = DriverManager.getConnection(

dbConnString, dbUser, dbPassword);

Consultas

Para poder realizar una consulta se necesita utilizar una instancia de la clase Statement

El resultado de la consulta de datos debemos almacenarla en una instancia de la clase ResultSet

Para mostrar los datos al usuario, se deberá iterar por el resultset

```
Ejemplo:
String laConsulta = "SELECT * FROM ALUMNOS";
Statement stmtConsulta = laConexion.createStatement();
ResultSet rs = stmtConsulta.executeQuery(laConsulta);
while( rs.next() )
System.out.println( "ID: " + rs.getInt("alumnoID") );
Cierre de objetos
Los objetos deberán cerrar para liberar los recursos utilizados por los mismos
La forma de cerrarlos es a través del método close()
Ejemplo:
stmtConsulta.close();
laConexion.close();
Inserción de datos
Se utiliza una instancia de la clase Statement
El método para ejecutar la sentencia es el execute()
Ejemplo:
String laInsercion = "INSERT INTO tabla ......
Statement stmtInsercion = laConexion.createStatement();
```

Actualización de datos

stmtInsercion.execute(laInsercion);

Se utiliza una instancia de la clase Statement

El método para ejecutar la sentencia es el execute() Ejemplo: String laActualizacion = "UPDATE tabla SET campo ="; Statement stmtActualizacion = laConexion.createStatement(); stmtActualizacion.execute(laActualizacion); Eliminación de datos Se utiliza una instancia de la clase Statement El método para ejecutar la sentencia es el execute() Ejemplo: String laEliminacion = "DELETE FROM tabla WHERE campo =... "; Statement stmtEliminacion = laConexion.createStatement(); stmtEliminacion.execute(laEliminacion); Conexión desde un JSP Para conectarse desde un página JSP es necesario conseguir una conexión Para conseguir una conexión dentro de un JSP se puede ejecutar siguiente código: (recordar hacer las importaciones necesarias) <% // Construcción de la conexión String dbDriver = "com.mysql.jdbc.Driver"; String dbConnString = "jdbc:mysql://localhost/educacionit"; String dbUser = "root";

```
String dbPassword = "";

// Registración del driver a utilizar

Class.forName(dbDriver).newInstance();

Connection laConexion = DriverManager.getConnection(
dbConnString, dbUser, dbPassword);

// Aquí ya se puede utilizar la conexión

<código>

%>
```

El código correspondiente a las consultas con las tablas puede realizarse debajo del código anterior presentado

Si bien este código funciona, es conveniente utilizar una clase que retorne una conexión y otras clases que realicen el acceso a datos

Qué es un DAO

Es una patrón de diseño que tiene como objetivo encapsular la lógica de acceso a datos

Significa Data Access Object

Por cada tabla, es conveniente construir una clase que tenga como responsabilidades la inserción, eliminación, modificación y consulta de los registros con esta tabla

Las páginas JSP deberían llamar a los DAOs

Connection Pooling

Es un mecanismo que permite tener varias conexiones instanciadas para ser utilizadas

Las conexiones son instanciadas una única vez cuando se inicializa la aplicación

Se logra un ahorro de tiempo significativo ya que al obtener la conexión, ésta ya está construida

Otro beneficio importante es que un Web server puede atender más pedidos simultáneos que un database server, con lo cual en ambientes de mucha carga hay que controlar que los pedidos no excedan la cantidad que soporta el database server