Jdbc : java database connectivity

Jdbc is an api which is used to connect and execute the queries with the db

Drivers : it is a software component that defines java application interact with databse

Jdbc requires drivers for each database for connection

Steps to connect java with database ::

1. load or register the driver

2. establish connection

3. create statement

4.execute statement

5. close connection

1) **Load the driver** : loading the driver we are making use of static method called as "forName()" which is present in "class" which takes fully qualified the vendor driver.

   class.forName("com.mysql.cj.jdbc.Driver");

2) **Establish connection** : connection is an interface present in "java.sql"package
   Since we cannot create an object for an interface we are making use of helper class called as "DriverManager". There is one static method which is overloaded present in the DriverManager class called as "getConnection()" whose return type is connection.

   A)   getConnection(String url);
   B)getConnection(String url, properties information)

C)getConnection(String url, String user, String pwd)

DB-URL :

Syntax :

Protocol : sub-protocol://hostname:portnumber/db_name

e.g :-

JDBC:mysql://localhost:3306/student_db

Connection cn=DriverManager.getConnection("JDBC:mysql://localhost:3306/student_db","root","root);

3) create statement : statement is alson an interface present in "java.sql" package.
With the help of non static method called as "CreateStatement" which is presenting connection.
We can create object for statement

e.g., Statement s=cn.createStatement();

4) Execute statement :

a) execute():- return type is boolean. DQL statements (select query)

b)executeUpdate():-return type is int. DDL statements

c)executeQuery():-return type is resultset

5) close connection :

Cn.close();

Statement	Prepared statement
It is used when sql query is to be executed once	It is used when sql query is to be executed multiple times
We cannot pass parameter at runtime.	We can pass parameter at run time
Used to execute normal sql queries	Used to dynamic sql queries
Performance is low	Performance is better than statement
Base interface (parent)	Extens interface(child)

Download jar file and install in project

https://dev.mysql.com/downloads/connector/j/

install mysql jar file and paste it in your eclipse folder. Download 4.8 M jar file.

After download and paste . you can go to eclipse.

Right click on project name --→ build path --→ add external jar libraries ---→ select mysql connector jar library ----→