**Steps to connect with DB**

1. Register Driver(Load Driver)
   1. Register the class which use to perform DB activities
2. Create Connection
   1. We will connect with database by providing information(URL, User Name, Password)
3. Create Statement
   1. Use to hold sql statement(Queries)
4. Execute Statement
   1. Execute query and return generated result into application.
5. Close Connection
   1. Close connection to release resources.

**Register Driver :**

**Class.forName("oracle.jdbc.driver.OracleDriver");**

**Create Connection**

URL : (Protocol + Host(Ip) + Port + Domain + resource)

Oracle : jdbc:oracle.thin:@localhost:1521(1522):xe

Mysql : jdbc:mysql://localhost:3306/DataBaseName

UserName :

Password :

**Create statement**

We have 3 options

1. Statement(I)
   1. It is use to execute any type of queries(DDL,DML,DCL)
   2. Is slower than other options
   3. Sql Injection issue can be raised
   4. Syntax :

**Statement stmt= con.createStatement();**

1. PreparedStatement(I)
   1. It is use to execute any type of queries(DDL,DML,DCL)
   2. Is faster than statement
   3. Sql Injection issue will never raised
   4. Can write parameterized query.
   5. Syntax :

**PreparedStatement stmt=con.prepareStatement("Query");**

1. CallableStatement(I)
   1. Is use to execute procedure or function
   2. Fastest among all other option
   3. Syntax :

CallableStatement**stmt=con.prepareCall("call{Procedure/Function}");**

**Execute Statement**

We have 3 options

1. executeQuery() : ResultSet
   1. Use to execute select type for queries

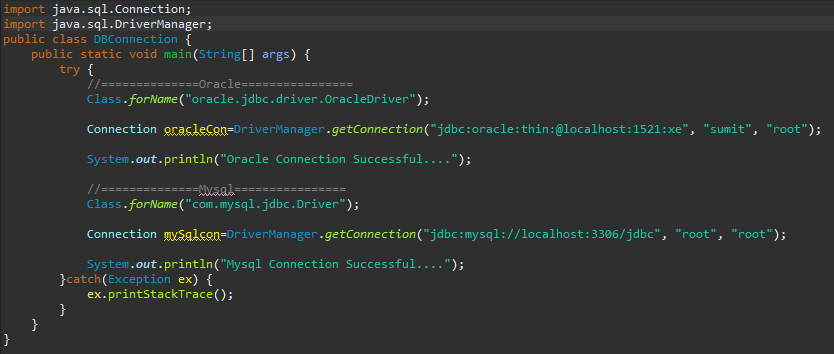
1. executeUpdate() : int
   1. Use to execute all other type of queries except select

1. execute() : boolean
   1. Use to execute all type of queries
   2. If return **true** then We can get **ResultSet**
   3. If return **false** then we can get **int**

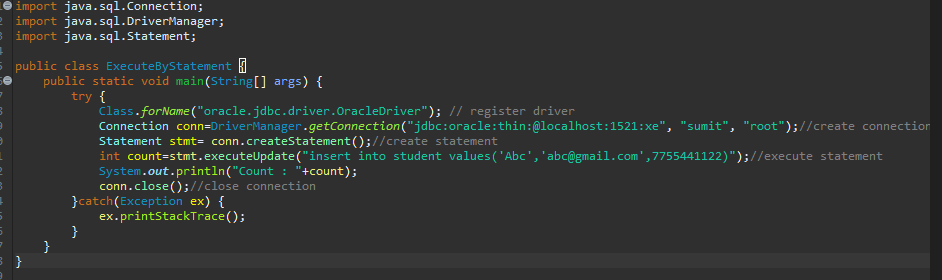
**Add External Jar file into Core java Project**

1. Right click on project
2. Build path -> Configure build path
3. Go to Lib. Tab
4. Click on add External jar file
5. Select jar file
6. Click on Apply close

**Data Base connection**



Completed all 5 Data Base steps



**Task**

WAP to register and login user

--------------------------------

Select Option

1. Register

2. Login

5. exit

if user select 1

Accept : Name, email, password from user

insert these values into database

Print "User register successfully" message

if user select 2

Accept : email and password from user and check whether user is valida or not

if user is valid the print Welcome "<UserName>" message

else print "Invalid username or password" message

if user select 5

exit from application

other than above options :

print "Invalid choice" message

Hint :

1. use scanner class to accept values from user

2. Use switch statement to check user choice and perform operation

3. Create two method 1st for register user and another for login functionality

4. System.exit(0); to Exit from application

Additional Feature :

Change password

if user select 3

Accept : email, old password, new password from user

update old password with new password only if email address and old password matches

else print "Invalid email or password" message

Hint : use update query and check email and old password by where clause.