Database : It is a repository of information. The information is stored following some structured and it is defined with the help of rows and columns.

RDBMS stands for relational database management System. It stores data in the form of tables , with most commercial relational database management system using SQL [ Structured Query Language ] to access database.

Prof E. F. Codd who introduced 12 rules to have a database named as RDBMS.

RDBMS Software

1. Microsoft SQL Server
2. Oracle Database Server
3. IBM DB2
4. MySQL Database
   1. Enterprise Edition
   2. Community Edition [ Open Source ] – Maria DB, Amazon Aurora

Column = Key

Record / Row = Tuple

Table = Relation

* Tables must have at least 1 primary key. That is used for row identification.
* Table may have relationship

Relationship between tables

1. One To One
2. One To Many
3. Many To Many

SQL = Structured Query Language. / ISO 9075

1. DDL : Data Definition Language
   1. CREATE
   2. ALTER
   3. DROP
2. DQL : Data Query Language
   1. SELECT
3. DML : Data Manipulation Language
   1. INSERT
   2. UPDATE
   3. DELETE
4. TQL : Transaction Query Language
   1. COMMIT
   2. ROLLBACK

**JDBC API**

In JDK / JRE– the classes & interfaces to communicate with database are found in java.sql package. The exception being thrown in case of database error in java is SQLException.

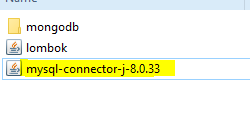
JDBC API contains abstract methods in their interfaces and implementation for them are provided by database vendors.

**Database Drivers** are the implementation of interfaces from JDBC API. Depending in the database thar you are using – you need to download database driver from vendors

**Download MySQL Database Driver**

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

for you guys – I have already downloaded and added to my github repository under drivers folder



This is a jar file - Java Archive that contains all the byte code for classes and interfaces.

To add database driver to your class path

Eclipse : Right click project -> Build Path – Configure Build Path

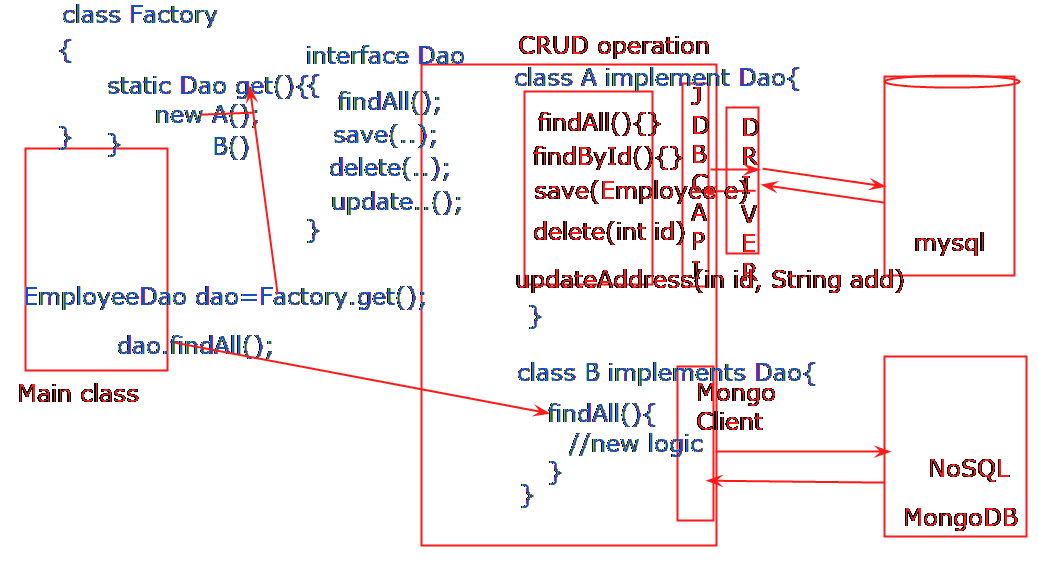
Under libraries tab – select classpath – add external jar file – apply - close

If you don’t have a constructor – a default [ no-arg ] will be supplied by compiler to byte code

If you have one – then no

[ no-arg needs to added explicitly if you need it ]

**Data Access Object [ DAO ]** Design Pattern



Try….with…Resource Statement

Interface AutoCloseable {

public void close();

}

Interface Closeable extends AutoCloseable{

}

All classes that implements AutoCloseable or Closeable – are known as Resource Statement Class.

**try**(Connection con=DriverManager.*getConnection*("jdbc:mysql://localhost:3306/deandemo", "root","admin#123");

Statement st=con.createStatement();

ResultSet rs=st.executeQuery("select \* from employee")

){

**while**(rs.next()) {

**int** id=rs.getInt("EMP\_ID");

String name=rs.getString("NAME");

**double** sal=rs.getDouble("SALARY");

**int** deptId=rs.getInt("DEPT\_ID");

Employee e=**new** Employee(id, name, sal, deptId);

empList.add(e);

}

} **catch** (SQLException e) {

e.printStackTrace();

}