Day-4

<class> tag is used to map POJO class DB Table
<id> tag is used to map POJO identity with primary key
<property>tag is used to map POJO properties with db columns

Hibernate CRUD methods---->save/load/update/delete

**To retrieve record
i)load method but it throws Exception if record not found
ii)use get method[returns null if record not found]
_____

**Accessing multiple databases using hibernate

By default name of hibernate configuration file is
       hibernate.cfg.xml
we can change it
multiple db
----
here we will create 2 configuration files
oracle.cfg.xml
mysql.cfg.xml

//client code
Configuration config1=new Configuration().configure
       ("oracle.cfg.xml");
SessionFactory factory1=config1.buildSessionFactory();
_____

Configuration config2=new Configuration().configure
       ("mysql.cfg.xml");
SessionFactory factory2=config2.buildSessionFactory();
_____
Session session1=factory1.getCurrentSession();
session1.save(emp);
Session session2=factory2.getCurrentSession();
session2.save(emp);
_____

**Generator classes
  ----
generator class is used to autogenerate the values for POJO identity/PK
1)assigned -programmer will supply the value
eg:-<generator class="assigned"/>

Here hibernate will auto generate the value not programmer
2)increment -auto generate start from 1 increment by 1
eg:-<generator class="increment"/>

3)sequence-auto start with x & increment by y
eg:-
 <generator class="sequence">
  <param name="sequence">mysequnc</param>
  <param name="parameters">START WITH 1000 INCREMENT BY
       2</param>
 </generator>

4)identity-unique integer values

5)uuid.hex-unique hexadecimal values

6)uuid.string-unique string values

7)native-let hibernate decide how to generate the values
eg:identity/increment/sequence

8)hilo-auto unique values but always new value wud be greater than old value
min-low=1200
table-name="mytabl"
1222   1250   1263   1289

9)seqhilo-auto values just like hilo
1222 1224 1226.....

Lab)
Book-->bookCode,title,author,price,publicationDate
**save/get/update/delete book information
**use generator class increment/sequence to auto generate bookCode
**create 4 console-based clients & use Scanner
client1--->save 4 records
client2---> load based on bookcode
client3--->update (give discount on book) accept code & disc in %
client4--->delete based on book code
client5--->swing client to save & load the data

**create a package com.hibernate.util --->HibernateUtil---> static method
public static SessionFactory obtainSessionFactory(){
        ....
}
In client
SessionFactory factory=HibernateUtil.obtainSessionFactory();