**Implementation of hibernate**:

//add annotations

@Entity // belongs to package jpapi

Public class Alien{

@Id //specifies primary key

Private int aid;

Private string aname;

Private string color;

//w v needto write getters and settrs for that you need to click right and select source and slect gettes and setters then v can

public int getAid(){

retuen aid;}

public void setAid(int aid){

this.aid = aid;

}

Public string getAname(){

Return aname;

}

Public void setAname(string aname){

This.aname = aname;

}

Public string getcolor(){

Return color;

}

Public void setColor(string color){

This.aname = aname;

}

}

**Create another app:**

Public class app

{

P s v m(s[] ar)

{

Alien ts = new alien();

Ts.setAid(1);

Ts.stAname(“hdfks”);

Ts.setcolor(“blue”);

//add depencnsies for hibernate and mysql in order to add library from mvnrepository sight and add hibrnatedepencies v4.1.6.

//configuration

Configuration conn = new configuration().configure().addAnnotatedcclass.Alien();;//by adding configure it will goto configurationfile.

//sessionfactory is a interface so v cant add object here so add configuration

Serviceregistry reg = new serviceregistrybuilder().applysettings(con.getproperties().buildserviceregistry();//by adding this please add another details of id because of primary key.

Sessionfactory sf = conn.buildsessionfactory(reg);

//sssion is a interface so v cant add object here so v add session factory

Session session = sf.opensession();

Transaction tx = session.begintransaction();//to support acid PROPERTIES

Session.Save(ts);

Tx.commit

}}

//here v get error dilect means which dbmc ur connect and u have to spefify usename and password.

Rightclick on ur project select other and hibernate->

Hibernate configurationfile->dilect=mysql/oracle, driverclass:com.mysql.jdbc.deiver, <url:jdbc:mysql://locolhost:3307/neon>, password:, username: ->finish

We get a page configuration file:

In this table we have to create table:

By specifying: **<property> name=”hbm2ddl.auto”>update</property>**//if table is not there place create instead of update, if v use create every time it will create new table and delete previous table.

Now togo mysql find alien table.

Here hibernate is converting into mysql if v want to see add property

**<property>Name=”show\_sql”>true</property>**

Now execute it shows all queries:

If u want to create atbale add in config file create indtead of update and write

**To change entityname which takes tablename same:**

@entity(name = alien-table)//so it will create a table name like entityname as alien-table;

**If u want only table name then add**

@table(name=alien-table)// so it will create table table and entity is diddernt add before class.

**If u want to change column name:**

@column(name=”aliencolumn”; // add this before column variable which changes the color to aliencolor

**If u want to delete name from table:**

@transient //before name varaibale

**HOW TO FETCH DATA FROM DATABASE USING HIBERNATE:**

TO fetch data v need to have some data in db ; use update

Delete the values from app classwhich are used for calling all name, color and id,

Write tostring method in alian class:

Public string tostring(){

Return :Alien[aid=” + aid + “, aname” + aname “, color=” + color=” + color +”];

}

And now delete session.save line

And print objct ts like sysout(“ts”); at last and add

ts =(alien) sssion.get(alian, 101); instead of sssion.save line

and write alien ts = null;

**HOW TO USE EMBEDABLE FIELDS / OBJECTS**

we can use another class alienname:

create another class

private class alianname{

private string fname;

private string laname;}

add getters and setters for this.and also tostring method

and now add this aliannam to alian class like

private Alianname name;

and now change the getters and setters method to name;

NOW goto app class and add session.save(ts); and also set the values in appclass like ts.setaid(12) and add create in configuration file and add

Alienname an = new alienname();

An,setfname(“”);

An.setlname();

And add ts.aname(an);

Add

**@embedable** in new alienname class so that class alian class will have all properties of alienname class