**Inheritance Concept in Java**

Inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object. It is an important part of OOPs (Object Oriented programming system).

The **idea behind inheritance** in Java is that you **can create new classes that are built upon existing classes.** When you inherit from an existing class, you can **reuse methods and fields of the parent class.** Moreover, you **can add new methods and fields in your current class also.**

**How to create a project and java file**

**Steps to create project, package and java file.**

File → New → Java project and give project name which you want to create

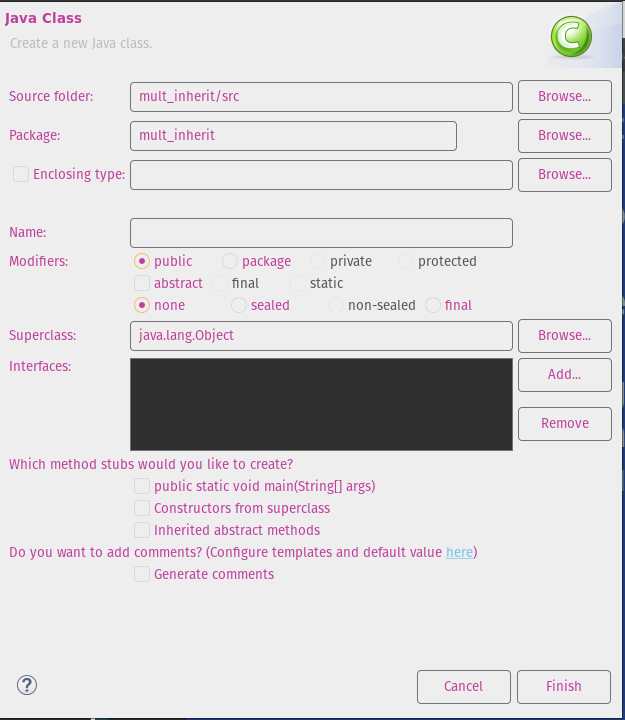
and press **don’t create module**

Click **Package explorer and select the project you have created** ( in our case it is mult\_inherit)

Right click on the mult\_inherit project folder → New → package and give package name ( in our case it is mult\_inherit)



right click on the mult\_inherit package folder → New → Class



Then type the Package name that is mult\_inherit and class name you want to create in Name column that is Emptest & then pick the check box public static void main and press finish button , you will have a class file with Emptest class with main() in side.

Copy and past the following coding

package mult\_inherit;

//Example program for multilevel inheritance in java with super() and method overriding method

//base class or parent class

class Emp {

protected String name;

protected int salary;

// constructor used to initialize the objects

Emp()

{

name=" ";

salary=0;

}

public void getdata(String name, int salary)

{

this.name=name;

this.salary=salary;

}

public void dispdata()

{

System.out.println("Employee name : "+name);

System.out.println("Employee salary : "+salary);

}

}

//derived or child class

class SkilledAsst extends Emp{

protected int allowance;

SkilledAsst()

{

// calling the base class constructor

super();

allowance=0;

}

public void getdata(String name, int salary, int allowance)

{

super.getdata(name,salary);

this.allowance=allowance;

}

public void dispdata()

{

super.dispdata();

System.out.println("Employee's allowance : "+allowance);

}

}

//inheriting multilevel classes

class Manager extends SkilledAsst {

private int bonus;

Manager ()

{

super();

bonus=0;

}

public void getdata(String name, int salary, int allowance, int bonus)

{

super.getdata(name,salary,allowance);

this.bonus=bonus;

}

public void dispdata()

{

super.dispdata();

System.out.println("Employee's bonus : "+bonus);

}

}

//Emptest.java

public class Emptest{

public static void main(String args[]){

Emp e = new Emp();

SkilledAsst s = new SkilledAsst();

Manager m= new Manager();

System.out.println("Details immediately after declaring objects : ");

e.dispdata();

s.dispdata();

m.dispdata();

System.out.println("Details after setting the data : ");

e.getdata("Ram",1200);

s.getdata("Ravi",1400,1000);

m.getdata("Raja",1800,1500,2000);

System.out.println("~~~~~~~~~~~~Employee's details~~~~~~~~~~~~ ");

e.dispdata();

System.out.println("~~~~~~~~~~~~SkilledAsst's details ~~~~~~~~~~~~ ");

s.dispdata();

System.out.println("~~~~~~~~~~~~Manager's details~~~~~~~~~~~~ ");

m.dispdata();

}

}

o/p

Details immediately after declaring objects :

Employee name :

Employee salary : 0

Employee name :

Employee salary : 0

Employee's allowance : 0

Employee name :

Employee salary : 0

Employee's allowance : 0

Employee's bonus : 0

Details after setting the data :

~~~~~~~~~~~~Employee's details~~~~~~~~~~~~

Employee name : Ram

Employee salary : 1200

~~~~~~~~~~~~SkilledAsst's details ~~~~~~~~~~~~

Employee name : Ravi

Employee salary : 1400

Employee's allowance : 1000

~~~~~~~~~~~~Manager's details~~~~~~~~~~~~

Employee name : Raja

Employee salary : 1800

Employee's allowance : 1500

Employee's bonus : 2000