**Question 1**

**Source Code**

public class Employee  
{  
String name,Position;  
int id;  
  
public Employee()  
{  
 name="";  
 id=0;  
 Position="";  
}  
  
public Employee(String nm,int i,String pos)  
{  
 name=nm;  
 id=i;  
 Position=pos;  
}  
  
public void setName(String nm){name=nm;}  
public void setId(int i){id=i;}  
public void setPosition(String pos){Position=pos;}  
  
public String getName(){return name;}  
public int getId(){return id;}  
public String getPosition(){return Position;}  
  
public String toString()  
{  
 return("\nName :"+getName()+"\nID :"+getId()+"\nPosition :"+getPosition());  
}   
  
}

public class Permanent extends Employee  
{  
double basicSalary;  
double houseAllowance;  
  
public Permanent()  
{  
 basicSalary=0.0;  
 houseAllowance=0.0;  
}  
  
public Permanent(String nm,int i,String pos,double bs,double ha)  
{  
 super(nm,i,pos);  
 basicSalary=bs;  
 houseAllowance=ha;  
}  
  
public void setBasicSalary(double bs){basicSalary=bs;}  
public void setHouseAllowance(double ha){houseAllowance=ha;}  
  
public double getBasicSalary(){return basicSalary;}  
public double getHouseAllowance(){return houseAllowance;}  
  
public double calcSalary()  
{  
 double salary=getBasicSalary()+getHouseAllowance()-(getBasicSalary()\*0.11)-(getBasicSalary()\*0.07);  
 return salary;  
}  
  
public String toString()  
{  
 return(super.toString()+"\nBasic Salary :"+getBasicSalary()+"\nHouse Allowance :"+getHouseAllowance());  
}  
}

public class PartTime extends Employee  
{  
double ratePerHour;  
double totWorkingHour;  
  
public PartTime()  
{  
 ratePerHour=0.0;  
 totWorkingHour=0.0;  
}  
  
public PartTime(String nm,int i,String pos,double rph,double twh)  
{  
 super(nm,i,pos);  
 ratePerHour=rph;  
 totWorkingHour=twh;  
}  
  
public void setRatePerHour(double rph){ratePerHour=rph;}  
public void setTotWorkingHour(double twh){totWorkingHour=twh;}  
  
public double getRatePerHour(){return ratePerHour;}  
public double getTotWorkingHour(){return totWorkingHour;}  
  
public double calcSalary()  
{  
 double salary=getRatePerHour()\*getTotWorkingHour();  
 return salary;  
}  
  
public String toString()  
{  
 return(super.toString()+"\nRate per Hour :"+getRatePerHour()+"\nTotal Working Hour :"+getTotWorkingHour());  
}  
}

import java.util.Scanner;  
public class EmployeeApp  
{  
 public static void main(String args[])  
 {  
 Scanner scan1=new Scanner(System.in);  
 Scanner scan2=new Scanner(System.in);  
   
 System.out.println("Enter number of employees in the company :");  
 int numEmp=scan2.nextInt();  
   
 Employee emp[]=new Employee[numEmp];  
   
 for(int x=0;x<numEmp;x++)  
 {  
 System.out.println("\nEnter Name :");  
 String name=scan1.nextLine();  
 System.out.println("Enter ID :");  
 int id=scan2.nextInt();  
 System.out.println("Enter position :");  
 String Position=scan1.nextLine();  
 emp[x]=new Employee(name,id,Position);  
   
 if(Position.equalsIgnoreCase("Permanent"))  
 {  
 System.out.println("Enter Basic Salary :");  
 double basicSalary=scan2.nextDouble();  
 System.out.println("Enter House Allowance :");  
 double houseAllowance=scan2.nextDouble();  
 emp[x]=new Permanent(name,id,Position,basicSalary,houseAllowance);  
 }  
 else if(Position.equalsIgnoreCase("PartTime"))  
 {  
 System.out.println("Enter Rate per Hour:");  
 double ratePerHour=scan2.nextDouble();  
 System.out.println("Enter Total Working Hour :");  
 double totWorkingHour=scan2.nextDouble();  
 emp[x]=new PartTime(name,id,Position,ratePerHour,totWorkingHour);  
 }  
 }  
   
 int countPm=0,countPt=0;  
 double totPm=0,totPt=0;  
 for(int y=0;y<numEmp;y++)  
 {  
 if(emp[y] instanceof Permanent)  
 {  
 Permanent per=(Permanent)emp[y];  
 totPm+=per.calcSalary();  
 System.out.println(per.toString()+"\n");  
 countPm++;  
 }  
   
 if(emp[y] instanceof PartTime)  
 {  
 PartTime part=(PartTime)emp[y];  
 totPt+=part.calcSalary();  
 System.out.println(part.toString()+"\n");  
 countPt++;  
 }  
   
 }  
 System.out.println("No. of permanent employees :"+countPm);  
 System.out.println("No. of part-time employees :"+countPt);  
 System.out.println("Total payment for permanent employees (RM) :"+totPm);  
 System.out.println("Total payment for part-time employees (RM) :"+totPt);  
 System.out.println("Total payment for all employees (RM) :"+(totPm+totPt));  
 }  
}

**Output i/o**

 ----jGRASP exec: java EmployeeApp  
Enter number of employees in the company :  
4  
Enter Name :  
A  
Enter ID :  
1  
Enter position :  
Permanent  
Enter Basic Salary :  
1000  
Enter House Allowance :  
100  
  
Enter Name :  
B  
Enter ID :  
2  
Enter position :  
PartTime  
Enter Rate per Hour:  
5  
Enter Total Working Hour :  
50  
  
Enter Name :  
C  
Enter ID :  
3  
Enter position :  
Parttime  
Enter Rate per Hour:  
8  
Enter Total Working Hour :  
30  
  
Enter Name :  
D  
Enter ID :  
4  
Enter position :  
Permanent  
Enter Basic Salary :  
2000  
Enter House Allowance :  
100

  
Name :A  
ID :1  
Position :Permanent  
Basic Salary :1000.0  
House Allowance :100.0  
  
  
Name :B  
ID :2  
Position :PartTime  
Rate per Hour :5.0  
Total Working Hour :50.0  
  
  
Name :C  
ID :3  
Position :Parttime  
Rate per Hour :8.0  
Total Working Hour :30.0  
  
  
Name :D  
ID :4  
Position :Permanent  
Basic Salary :2000.0  
House Allowance :100.0  
  
No. of permanent employees :2  
No. of part-time employees :2  
Total payment for permanent employees (RM) :2660.0  
Total payment for part-time employees (RM) :490.0  
Total payment for all employees (RM) :3150.0  
  
 ----jGRASP: operation complete.

**Question 2**

**Source code**

public class Food  
{  
 String name;  
 int quantityOfOrder;  
 boolean member;  
   
 public Food(String nm,int qoo,boolean mem)  
 {  
 name=nm;  
 quantityOfOrder=qoo;  
 member=mem;  
 }  
   
 public void setName(String nm){name=nm;}  
 public void setQuantityOfOrder(int qoo){quantityOfOrder=qoo;}  
 public void setMember(boolean mem){member=mem;}  
   
 public String getName(){return name;}  
 public int getQuantityOfOrder(){return quantityOfOrder;}  
 public boolean getMember(){return member;}  
   
 public String toString()  
 {  
 return("\nName :"+getName()+"\nQuantity of order :"+getQuantityOfOrder()+"\nMember :"+getMember());  
 }  
   
}

public class WesternFood extends Food  
{  
 int foodSet;  
 boolean desert;  
   
 public WesternFood(String nm,int qoo,boolean mem,int fset,boolean des)  
 {  
 super(nm,qoo,mem);  
 foodSet=fset;  
 desert=des;  
 }  
   
 public void setFoodSet(int fset){foodSet=fset;}  
 public void setDesert(boolean des){desert=des;}  
   
 public int getFoodSet(){return foodSet;}  
 public boolean getDesert(){return desert;}  
   
 public double Payment()  
 {  
 double amount=0;  
 if(getFoodSet()==1)  
 {amount=30;}  
 else if(getFoodSet()==2)  
 {amount=20;}  
 else if(getFoodSet()==3)  
 {amount=15;}  
   
   
 if(getDesert()==true)  
 {amount=amount+10.90;}  
   
 if(super.getMember()==true)  
 {amount=amount-(amount\*0.10);}  
   
 return amount;  
 }  
   
 public String toString()  
 {  
 return(super.toString()+"\nfoodSet :"+getFoodSet()+"\nDesert :"+getDesert()+"\nAmount :"+Payment());  
 }  
}

import java.util.Scanner;  
public class FoodApp  
{  
 public static void main(String args[])  
 {  
 Scanner scan1=new Scanner(System.in);  
 Scanner scan2=new Scanner(System.in);  
   
 System.out.println("Enter number of data to be stored :");  
 int num=scan1.nextInt();  
   
 Food f[]=new Food[num];  
   
 for(int x=0;x<num;x++)  
 {  
 System.out.println("\nEnter Name :");  
 String name=scan2.nextLine();  
 System.out.println("Enter Quantity of order :");  
 int quantityOfOrder=scan1.nextInt();  
 System.out.println("Enter 'true' if member, 'false' if non-member");  
 boolean member=scan1.nextBoolean();  
 f[x]=new Food(name,quantityOfOrder,member);  
   
 System.out.println("Enter food set :");  
 int foodSet=scan1.nextInt();  
 System.out.println("Enter 'true' for desert, otherwise 'false' ");  
 boolean desert=scan1.nextBoolean();  
 f[x]=new WesternFood(name,quantityOfOrder,member,foodSet,desert);  
 }  
   
 int count=0;  
 double totamt=0;  
 double totmem=0;  
 for(int y=0;y<num;y++)  
 {  
 System.out.println(f[y].toString());  
  
 if(f[y] instanceof WesternFood)  
 {  
 WesternFood wf=(WesternFood)f[y];  
 if(wf.getDesert()==true)  
 {  
 count++;  
 }  
   
 if(wf.getFoodSet()== 1)  
 {  
 System.out.println("\nCustomer who order lamb chop :"+wf.toString());  
 }  
 totamt+=wf.Payment();   
 }   
   
 if(f[y] instanceof Food)  
 {  
 Food fd=(Food)f[y];  
 if(fd.getMember()==true)  
 {  
 totmem++;  
 }  
   
 }  
 }  
 System.out.println("\nNumber of customer who ordered desert :"+count);  
 System.out.println("Total amount of member's customer :"+totmem);  
 System.out.println("Total amount of charges for all customer :"+totamt); }  
}

**Output i/o**

 ----jGRASP exec: java FoodApp  
Enter number of data to be stored :  
3  
  
Enter Name :  
Amin  
Enter Quantity of order :  
1  
Enter 'true' if member, 'false' if non-member  
True  
Enter food set :  
3  
Enter 'true' for desert, otherwise 'false'   
false  
  
Enter Name :  
Aiman  
Enter Quantity of order :  
1  
Enter 'true' if member, 'false' if non-member  
false  
Enter food set :  
2  
Enter 'true' for desert, otherwise 'false'   
false  
  
Enter Name :  
Ahmad  
Enter Quantity of order :  
1  
Enter 'true' if member, 'false' if non-member  
true  
Enter food set :  
1  
Enter 'true' for desert, otherwise 'false'   
true  
  
Name :Amin  
Quantity of order :1  
Member :true  
foodSet :3  
Desert :false  
Amount :13.5  
  
Name :Aiman  
Quantity of order :1  
Member :false  
foodSet :2  
Desert :false  
Amount :20.0  
  
Name :Ahmad  
Quantity of order :1  
Member :true  
foodSet :1  
Desert :true  
Amount :36.81  
  
Customer who order lamb chop :  
Name :Ahmad  
Quantity of order :1  
Member :true  
foodSet :1  
Desert :true  
Amount :36.81  
  
Number of customer who ordered desert :1  
Total amount of member's customer :2.0  
Total amount of charges for all customer :70.31  
  
 ----jGRASP: operation complete.

**Question 3**

public class Order  
{  
 private String orderId,orderDate;  
 private int quantitySetA,quantitySetB,quantitySetC;  
 private final double PRICE\_SETA=187.0,PRICE\_SETB=258.0,PRICE\_SETC=377.0;  
   
 public Order(String oi,String od,int qa,int qb,int qc)  
 {  
 orderId=oi;  
 orderDate=od;  
 quantitySetA=qa;  
 quantitySetB=qb;  
 quantitySetC=qc;  
 }   
   
 public String getOrderId(){return orderId;}  
 public String getOrderDate(){return orderDate;}  
 public int getQuantitySetA(){return quantitySetA;}  
 public int getQuantitySetB(){return quantitySetB;}  
 public int getQuantitySetC(){return quantitySetC;}  
   
 public double calculateGrossPrice()  
 {

double GPrice=(getQuantitySetA()\*PRICE\_SETA)+(getQuantitySetB()\*PRICE\_SETB)+(getQuantitySetC()\*PRICE\_SETC);  
 return GPrice;  
 }  
 double NPrice;  
 public double calculateNetPrice(){ return NPrice = 0;}  
  
   
 public String toString()  
 {  
 return("\nOrder ID :"+getOrderId()+"\nOrder Date:"+getOrderDate()+"\nQuantity Set A :"+getQuantitySetA()+"\nQuantity Set B :"+getQuantitySetB()+"\nQuantity Set C :"+getQuantitySetC());  
 }  
}

public class WalkInOrder extends Order  
{  
 private String phoneNo;  
   
 public WalkInOrder(String oi,String od,int qa,int qb,int qc,String Pnum)  
 {  
 super(oi,od,qa,qb,qc);  
 phoneNo=Pnum;  
 }  
   
 public String getPhoneNo(){return phoneNo;}  
   
 public double calculateNetPrice()  
 { double NPrice=0;  
 if(super.calculateGrossPrice()>=400)  
 {   
 NPrice=super.calculateGrossPrice()-(super.calculateGrossPrice()\*0.10);  
 }  
 else  
 {  
 NPrice=super.calculateGrossPrice();  
 }return NPrice;  
 }  
   
 public String toString()  
 {  
 return(super.toString()+"\nPhone No :"+getPhoneNo());  
 }  
}

public class AgentOrder extends Order  
{  
 private double agentId;  
   
 public AgentOrder(String oi,String od,int qa,int qb,int qc,double aid)  
 {  
 super(oi,od,qa,qb,qc);  
 agentId=aid;  
 }  
   
 public double getAgentId(){return agentId;}  
   
 public double calculateNetPrice()  
 {  
 int point1=super.getQuantitySetA()\*72;  
 int point2=super.getQuantitySetB()\*85;  
 int point3=super.getQuantitySetC()\*110;  
   
 int totPoint=point1+point2+point3;  
   
 double NPrice=0;  
 if(totPoint>=700)  
 {  
 NPrice=super.calculateGrossPrice()-(super.calculateGrossPrice()\*0.40);  
 }  
 else if(totPoint>=450)  
 {  
 NPrice=super.calculateGrossPrice()-(super.calculateGrossPrice()\*0.35);  
 }  
 else if(totPoint>=200)  
 {  
 NPrice=super.calculateGrossPrice()-(super.calculateGrossPrice()\*0.30);  
 }  
 else  
 {  
 NPrice=super.calculateGrossPrice();  
  
 }  
 return NPrice;  
 }  
   
 public String toString()  
 {  
 return(super.toString()+"\nAgent ID :"+getAgentId());  
 }  
  
}

import java.util.Scanner;  
public class OrderApp  
{  
 public static void main(String args[])  
 {  
 Scanner scan1=new Scanner(System.in);  
 Scanner scan2=new Scanner(System.in);  
   
 System.out.println("\nEnter number of customer :");  
 int size=scan1.nextInt();  
 Order o[]=new Order[size];  
   
 for(int x=0;x<size;x++)  
 {  
 System.out.println("\nEnter Order ID :");  
 String orderId=scan2.nextLine();  
 System.out.println("Enter Order date :");  
 String orderDate=scan2.nextLine();  
 System.out.println("Quantity of Set A :");  
 int quantitySetA=scan1.nextInt();  
 System.out.println("Quantity of Set B :");  
 int quantitySetB=scan1.nextInt();  
 System.out.println("Quantity of Set C :");  
 int quantitySetC=scan1.nextInt();  
   
 o[x]=new Order(orderId,orderDate,quantitySetA,quantitySetB,quantitySetC);  
 System.out.println("Walk in enter 'true' else Agent enter 'false' :");  
 boolean type=scan1.nextBoolean();  
   
 if(type==true)  
 {  
 System.out.println("Enter phone no :");  
 String phoneNo=scan2.nextLine();  
 o[x]=new WalkInOrder(orderId,orderDate,quantitySetA,quantitySetB,quantitySetC,phoneNo);  
 }  
 else  
 {  
 System.out.println("Enter agent id :");  
 double agentId=scan1.nextDouble();  
 o[x]=new AgentOrder(orderId,orderDate,quantitySetA,quantitySetB,quantitySetC,agentId);  
  
 }  
 }  
   
 int totp=0;  
 for(int y=0;y<size;y++)  
 {  
 if(o[y] instanceof AgentOrder)  
 {  
 AgentOrder ao=(AgentOrder)o[y];  
  
 if(ao.calculateGrossPrice()==ao.calculateNetPrice())  
 {  
 System.out.println("Agent's ID that not receive any discount :"+ao.getAgentId());  
 }  
 }  
   
 if(o[y] instanceof WalkInOrder)  
 {  
 WalkInOrder wi=(WalkInOrder)o[y];  
 if(wi.getOrderDate().equalsIgnoreCase("25 March 2013"))  
 {  
 totp+=wi.calculateNetPrice();  
 }  
  
 }  
 }  
 System.out.println("\n Walk in customer on 25 March 2013 total purchace :"+totp);  
 }  
}

**Output i/0**

 ----jGRASP exec: java OrderApp  
  
Enter number of customer :  
5  
  
Enter Order ID :  
1111  
Enter Order date :  
25 march 2013  
Quantity of Set A :  
1  
Quantity of Set B :  
1  
Quantity of Set C :  
1  
Walk in enter 'true' else Agent enter 'false' :  
true  
Enter phone no :  
0172787738  
  
Enter Order ID :  
2222  
Enter Order date :  
23 march 2013  
Quantity of Set A :  
1  
Quantity of Set B :  
0  
Quantity of Set C :  
0  
Walk in enter 'true' else Agent enter 'false' :  
false  
Enter agent id :  
1001  
  
Enter Order ID :  
3333  
Enter Order date :  
22 march 2013  
Quantity of Set A :  
3  
Quantity of Set B :  
2  
Quantity of Set C :  
1  
Walk in enter 'true' else Agent enter 'false' :  
true  
Enter phone no :  
0182342902  
  
Enter Order ID :  
4444  
Enter Order date :  
25 march 2013  
Quantity of Set A :  
3  
Quantity of Set B :  
2  
Quantity of Set C :  
2  
Walk in enter 'true' else Agent enter 'false' :  
true  
Enter phone no :  
0178356821  
  
Enter Order ID :  
5555  
Enter Order date :  
22 march 2013  
Quantity of Set A :  
0  
Quantity of Set B :  
2  
Quantity of Set C :  
0  
Walk in enter 'true' else Agent enter 'false' :  
false  
Enter agent id :  
1002  
Agent's ID that not receive any discount :1001.0  
Agent's ID that not receive any discount :1002.0  
  
 Walk in customer on 25 March 2013 total purchace :2386  
  
 ----jGRASP: operation complete.  




**FACULTY OF COMPUTER SCIENCE AND MATHEMATICS**

**CS110**

Lab Assignment 4 – Inheritance

NAME: MUHAMMAD AL-AMIN BIN MOHD ZAINI

MATRIC NO: 2018280578

GROUP: RCS1103G

ASSIGNMENT : LAB ASSIGNMENT 4

PROGRAMME CODE: CS110

COURSE CODE: CSC238

