REGISTERED CUSTOMER(100 POINTS)

```
import java.util.*;
class Address
{
    String l1,l2,city,pin;
    Address(String a,String b,String c,String d)
    {
        l1=a;
        l2=b;
        city=c;
        pin=d;
    }

    void setl1(String x)
    {
     this.l1=x;
    }
    String getl1()
    {
     return this.l1;
    }

    void setl2(String x)
    {
     this.l2=x;
    }
    String getl2()
    {
     return this.l2;
    }
```

```
    void setcity(String x)
    {
     this.city=x;
    }
    String getcity()
    {
     return this.city;
    }

    void setpin(String x)
    {
     this.pin=x;
    }
    String getpin()
    {
     return this.pin;
    }

}
class Customer
{
    String custid, custname;
    Address address;
    Customer(String custid, String custname, Address address)
    {
        this.custid=custid;
        this.custname=custname;
        this.address=address;
    }

    String getcustid()
    {
     return this.custid;
```

```
    }
    String getcustname()
    {
     return this.custname;
    }

    String getl1()
    {
     return this.address.l1;
    }
    String getl2()
    {
     return this.address.l2;
    }
    String getcity()
    {
     return this.address.city;
    }
    String getpin()
    {
     return this.address.pin;
    }
}

class RegCustomer extends Customer
{
    double fees;
    RegCustomer(String custid, String custname, Address address,double fees)
    {
        super(custid,custname,address);
        this.fees=fees;
    }
```

```
void setcustid(String x)
{
 this.custid=x;
}
void setcustname(String x)
{
 this.custname=x;
}
void setfees(double x)
{
 this.fees=x;
}
void setl1(String x)
{
 this.address.l1=x;
}
void setl2(String x)
{
 this.address.l2=x;
}
void setcity(String x)
{
 this.address.city=x;
}
void setpin(String x)
{
 this.address.pin=x;
}

double getfees()
{
 return this.fees;
}
```

```
    void display()
    {
        System.out.println("Customer Id :"+this.custid+"\nCustomer Name
:"+this.custname+"\nCustomer fees :"+this.fees);
        System.out.println("Address 1 :"+this.address.l1+"\nAddress 2
:"+this.address.l2+"\nCity :"+this.address.city);
        System.out.println("Pin :"+this.address.pin);
    }
}
public class source
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        String l1=sc.nextLine();
        String l2=sc.nextLine();
        String city=sc.nextLine();
        String pin=sc.nextLine();
        Address a=new Address(l1,l2,city,pin);
        String custId=sc.nextLine();
        String custName=sc.nextLine();
        double fees=sc.nextDouble();
        RegCustomer ob=new RegCustomer(custId,custName,a,fees);
        ob.display();
    }
}
```

BEAUTY PARLOUR(100)

```
import java.io.*;
import java.util.*;

class Customers
{
    String name;
    public String getName(){
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public boolean isMember() {
        return member;
    }
    public void setMember(boolean member) {
        this.member = member;
    }
    public String getMembertype() {
        return membertype;
    }
    public void setMembertype(String membertype) {
        this.membertype = membertype;
    }
    boolean member;
    String membertype;
    Customers(String name)
    {
        this.name=name;
    }
    @Override
```

```
    public String toString() {
        return "Customer [name=" + name + ", member=" + member
                + ", membertype=" + membertype + "]";
    }



}


class Visit
{
    String name;
    Customers cust;
    double serviceExpense;
    double productExpense;
    double totalExpense;

    Visit(Customers cust)
    {
        this.cust=cust;

    }

    public String getName()
    {
        return cust.getName();
    }
    public double getServiceExpense()
    {
        return serviceExpense;
    }
```

```
    public void setServiceExpense(double serviceExpense)
    {
        this.serviceExpense=serviceExpense;
    }

    public double getProductExpense() {
        return productExpense;
    }

    public void setProductExpense(double productExpense) {
        this.productExpense = productExpense;



    }

    public double totalExpense()
    {
        double dis=0;
        double dis1=0;
        //System.out.println("Mtype in tot exp:"+cust.getMembertype());
        if(cust.getMembertype().equals("null"))
        {
            return serviceExpense+productExpense;
        }
        else

            dis= serviceExpense *
DiscountRate.getServiceDiscountRate(cust.getMembertype());
        dis1=productExpense*DiscountRate.proddiscount;
        double prodiscount=productExpense-dis1;
        double totalExpense1=serviceExpense-dis;
//System.out.println("After Discount on service:"+totalExpense1);
        System.out.println(totalExpense1);
```

```
        double totalExpense2=productExpense-dis1;
//System.out.println("After Discount on product:"+totalExpense2);
        System.out.println(totalExpense2);



//return totalExpense=serviceExpense-dis;
//return totalExpense=serviceExpense+prodiscount;
        return totalExpense=totalExpense1+totalExpense2;


    }


    @Override
/*public String toString() {
 return "Visit [ cust=" + cust + ", serviceExpense="
   + serviceExpense + ", productExpense=" + productExpense
   +" Discount
Rate="+DiscountRate.getServiceDiscountRate(cust.getMembertype())+"]";
}*/


    public String toString() {
        return "[Customer Name:"+cust+"Service
Expense:"+serviceExpense+"Discount:"+DiscountRate.getServiceDiscountRate(cus
t.getMembertype())+"]Product Discout:"+DiscountRate.proddiscount+"Product
Discount:"+DiscountRate.getProductDiscountRate(cust.getMembertype());




    }


}


class DiscountRate
{
    static double premiumService=0.2;
```

```
    static double goldService=0.15;
    static double silverService=0.1;

    static double prodsilverService=0.1;

    static double prodgoldService=0.1;

    static double prodpremiumService=0.1;



    static double proddiscount=0.1;

    public static double getServiceDiscountRate(String service)
    {
        //System.out.println("Mtype is:"+service);
        if(service.equals("Premium"))
        {

            return premiumService;
        }
        else if(service.equals("Gold"))
        {
            return goldService;
        }
        else if(service.equals("Silver"))
        {
            return silverService;
        }
        else if(service.equals("null"))
        {
            System.out.println("Not Qualified for any Discounts on
Service/Products");
```

```
        }
        return 0;

    }

    public static double getProductDiscountRate(String service)
    {

        if(service.equals("Premium"))
        {
return prodpremiumService;
        }
        else if(service.equals("Gold"))
        {
            return prodgoldService;
        }
        else if(service.equals("Silver"))
        {
            return prodsilverService;
        }
        else
            return 0;







    }

}

// Class name should be "Source",
```

```
// otherwise solution won't be accepted

public class Source {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in);

//System.out.println("Enter Customer Name");

        String name=s.next();

//System.out.println("Enter true for membership or false for no
membership");

        boolean b=s.nextBoolean();

//System.out.println("membership Type(Gold/Silver/Premium)");

        String mtype=s.next();

//System.out.println("Enter Service Expense");

        double serexp=s.nextDouble();

//System.out.println("Enter Product Expense");

        double prodexp=s.nextDouble();

        Customers c=new Customers(name);

        Visit v=new Visit(c);

        c.setMember(b);

        c.setMembertype(mtype);

        v.setServiceExpense(serexp);

        v.setProductExpense(prodexp);

        DiscountRate.getServiceDiscountRate(c.getMembertype());

        System.out.println(c.getName());

        System.out.println(c.getMembertype());

        System.out.println(v.getServiceExpense());

        System.out.println(v.getProductExpense());

        System.out.println(v.totalExpense());

    }

}
```

TWO PERSON(100)

```
Scanner s=new Scanner(System.in);
  String name=s.next();
  int a=s.nextInt();
  String g=s.next();
  String name1=s.next();
  int a1=s.nextInt();
  String g1=s.next();
  if(name.equals(name1) && a==a1 && g.equals(g1))
  {
      System.out.println("The persons are same...");
  }
  else
  {
       System.out.println("The persons are different...");
  }
```

PHONE BOOK(not sure but only solution)

```
public void setPhoneNumber(long phoneNumber) throws
InvalidPhoneNumberExcception {
    String phone = Long.toString(phoneNumber);
    int mobileChecker = Pattern.matches("^[6-9][0-9]{9}$", phone) ? 1 : -1;
    if (mobileChecker == 1) {
      this.phoneNumber = phoneNumber;
    } else {
      throw new InvalidPhoneNumberExcception();
    }
  }

  public Address getAddress() {
    return address;
  }

  public void setAddress(Address address) {
    this.address = address;
  }

  @Override
  public String toString() {
    return String.format(
        "Customer [userId=%s, emailId=%s, password=%s, firstName=%s,
lastName=%s, city=%s, gender=%s, phoneNumber=%s, address=%s]",
        userId, emailId, password, firstName, lastName, city, gender,
phoneNumber, address);
  }

}

class Address {
```

```
private String city;
private String state;
private int zip;
private String country;

Address() {

}

public Address(String city, String state, int zip, String country) {
  this.city = city;
  this.state = state;
  this.zip = zip;
  this.country = country;
}

public String getCity() {
  return city;
}

public void setCity(String city) {
  this.city = city;
}

public String getState() {
  return state;
}

public void setState(String state) {
  this.state = state;
}

public int getZip() {
```

```
    return zip;
  }

  public void setZip(int zip) {
    this.zip = zip;
  }

  public String getCountry() {
    return country;
  }

  public void setCountry(String country) {
    this.country = country;
  }

  @Override
  public String toString() {
    return String.format("Address [city=%s, state=%s, zip=%s, country=%s]",
city, state, zip, country);
  }

}

public class Source {
  public static void main(String[] args) throws InvalidNameException,
InvalidPhoneNumberExcception {

  }
}
```

DATE MONTH EXCEPTION(100)

```
import java.util.Scanner;



class MonthException extends Exception{
 public MonthException(String message){
  super(message);
 }
}
class DayException extends Exception{
 public DayException(String message){
  super(message);
 }
}
class YearException extends Exception{
 public YearException(String message){
  super(message);
 }
}
public class TestException{
  public static void main(String[] args) {
   int monthnum;
   int monthDays=0;
   String monthName="";
   String date="";
   Scanner input=new Scanner(System.in);
   System.out.println("Please enter a date in this format:
Month/Day/Year.");
   date=input.next();
   String[] pars=date.split("/");
   int month=Integer.parseInt(pars[0]);
   int day=Integer.parseInt(pars[1]);
   int year=Integer.parseInt(pars[2]);
```

```
switch(month){
 case 1:
  monthName="January";
  monthDays=31;
 case 2:
  monthName="February";
  monthDays=28;
 case 3:
  monthName="March";
  monthDays=31;
 case 4:
  monthName="April";
  monthDays=30;
 case 5:
  monthName="May";
  monthDays=31;
 case 6:
  monthName="June";
  monthDays=30;
 case 7:
  monthName="July";
  monthDays=31;
 case 8:
  monthName="August";
  monthDays=31;
 case 9:
  monthName="September";
  monthDays=30;
 case 10:
  monthName="October";
  monthDays=31;
 case 11:
  monthName="November";
```

```
  monthDays=30;
 case 12:
  monthName="December";
  monthDays=31;
 default:
  System.out.println("Not valid.");
}
while(true){
try{
 if(month&lt;1||month&gt;12){
  throw new MonthException("The month must be numbers 1-12.");
  }
  else{
   break;
  }
}
catch(MonthException e){
 System.out.println("Please enter a valid month: ");
 month=input.nextInt();
 continue;
}
}
while(true){
 try{
  if(day&lt;1||day&gt;monthDays){
   throw new DayException("That day does not exist in this month.");
  }
  else{
   break;
  }
 }
 catch(DayException e){
  System.out.println("Please enter a valid day: ");
```

```
      day=input.nextInt();
      continue;
     }
    }
    while(true){
     try{
      if(year&lt;=1000||year&gt;=3000){
       throw new YearException("The year must be between 1000 and 3000.");
      }
      else{
       break;
      }
     }
     catch(YearException e){
      System.out.println("Please enter a valid year: ");
      year=input.nextInt();
      continue;
     }
    }
    System.out.println("The date conversion is: " + monthName+ " " + day + ",
" + year);
   }
}
```

Rail Compartment 100

```
import java.util.Random;
import java.util.Scanner;
public class Main
{
  public static void main(String[] args) {
      Scanner s=new Scanner(System.in);
      int i=0;
      int arr []=new int[10];
      int p=s.nextInt();
      Random rand = new Random();
        int upperbound = 4;
        int int_random = rand.nextInt(upperbound);
        int_random=int_random+1;
        if(p==1)
        {
            FirstClass a= new FirstClass();
            for(i=0;i<10;i++)
            {
                a.notice();
            }
        }
        else if(p==2)
        {
            General a= new General();
            for(i=0;i<10;i++)
            {
                a.notice();
            }
        }
        else if(p==3)
        {
```

```
            Ladies a= new Ladies();
            for(i=0;i<10;i++)
            {
                a.notice();
            }
        }
        else
        {
            Luggage a= new Luggage();
            for(i=0;i<10;i++)
            {
                a.notice();
            }
        }
  }
}
 abstract class Compartment
{
    abstract void notice();
}
class FirstClass extends Compartment
{
    public void notice()
    {
       System.out.println("FirstClass Compartment");
    }
}
class Ladies extends Compartment
{
    public void notice()
    {
       System.out.println("Ladies Compartment");
    }
```

```
}
class General extends Compartment
{
    public void notice()
    {
       System.out.println("General Compartment");
    }
}
class Luggage extends Compartment
{
    public void notice()
    {
       System.out.println("Luggage Compartment");
    }
}
```