```
public class FamilyInsurancePolicy extends InsurancePolicies{

	public FamilyInsurancePolicy(String clientName, String policyId,
int age, long mobileNumber, String emailId) {
		super(clientName, policyId, age, mobileNumber, emailId);

	}

	public boolean validatePolicyId()
	{
		int count=0;
		if(policyId.contains("FAMILY"));
		count++;
		char ch[]=policyId.toCharArray();
		for(int i=6;i<9;i++)
  {
			if(ch[i]>='0' && ch[i]<='9')
			count++;
		}
		if(count==4)
			return true;
		else
		return false;
	}

	 public double calculateInsuranceAmount(int months, int
no_of_members)
	{
		 double amount=0;
		 if(age>=5 && age<=25)
			 amount=2500*months*no_of_members;
		 else if (age>25 && age<60)
			 amount=5000*months*no_of_members;
		 else if (age>=60)
			 amount=10000*months*no_of_members;
		 return amount;
	 }

   }




public class IndividualInsurancePolicy extends InsurancePolicies{

	public IndividualInsurancePolicy(String clientName, String policyId,
int age, long mobileNumber, String emailId) {
		super(clientName, policyId, age, mobileNumber, emailId);

	}

	public boolean validatePolicyId()
```

```
        {
                int count=0;
                if(policyId.contains("SINGLE"));
                count++;
                char ch[]=policyId.toCharArray();
                for(int i=6;i<9;i++)
                {
                        if(ch[i]>='0' && ch[i]<='9')
                                count++;
                }
                if(count==4)
                        return true;
                else
                        return false;
 }

         public double calculateInsuranceAmount(int months)
         {
                 double amount=0;
                 if(age>=5 && age<=25)
                         amount=2500*months;
                 else if (age>25 && age<60)
                         amount=5000*months;
                 else if (age>=60)
                         amount=10000*months;
                 return amount;
         }

 }




public class InsurancePolicies {

        protected String clientName;
        protected String policyId;
        protected int age;
        protected long mobileNumber;
        protected String emailId;
        public String getClientName() {
                return clientName;
        }
        public void setClientName(String clientName) {
                this.clientName = clientName;
        }
        public String getPolicyId() {
                return policyId;
        }
        public void setPolicyId(String policyId) {
                this.policyId = policyId;
```

```
 }
	public int getAge() {
		return age;
	}
	public void setAge(int age) {
		this.age = age;
	}
	public long getMobileNumber() {
		return mobileNumber;
	}
	public void setMobileNumber(long mobileNumber) {
		this.mobileNumber = mobileNumber;
	}
	public String getEmailId() {
		return emailId;
	}
	public void setEmailId(String emailId) {
		this.emailId = emailId;
	}

	public InsurancePolicies(String clientName, String policyId, int age,
long mobileNumber, String emailId) {
		super();
		this.clientName = clientName;
		this.policyId = policyId;
		this.age = age;
		this.mobileNumber = mobileNumber;
		this.emailId = emailId;
	}
}
```

```
public class SeniorCitizenPolicy extends InsurancePolicies{

	public SeniorCitizenPolicy(String clientName, String policyId, int
age, long mobileNumber, String emailId) {
		super(clientName, policyId, age, mobileNumber, emailId);

	}

	public boolean validatePolicyId()
	{
		int count=0;
		if(policyId.contains("SENIOR"));
		count++;
		char ch[]=policyId.toCharArray();
		for(int i=6;i<9;i++)
```

```
                {
                        if(ch[i]>='0' && ch[i]<='9')
                                count++;
                }
                if(count==4)
                        return true;
                else
                        return false;
        }

         public double calculateInsuranceAmount(int months, int
no_of_members)
        {
                 double amount=0;
                 if(age>=5 && age<60)
                         amount=0;
                 else if (age>=60)
                         amount=10000*months*no_of_members;
                 return amount;
        }

  }
```

```
import java.util.Scanner;

    public class UserInterface {

        public static void main(String[] args)
        {

                Scanner sc=new Scanner(System.in);
                System.out.println("Enter Client name");
                String name=sc.next();
                System.out.println("Enter Policy Id");
                String id=sc.next();
                System.out.println("Enter Client age");
                int age=sc.nextInt();
                System.out.println("Enter mobile number");
                long mnum=sc.nextLong();
                System.out.println("Enter Email Id");
                String email=sc.next();

                InsurancePolicies policy=new
InsurancePolicies(name,id,age,mnum,email);
```

```
                System.out.println("Enter the months");
                int month=sc.nextInt();

                double amount=0;
                if(id.contains("SINGLE"))
                {
                        IndividualInsurancePolicy g=new
IndividualInsurancePolicy(name,id,age,mnum,email);
                        if(g.validatePolicyId())
                        {
                                //System.out.println(g.validatePolicyId());

amount=g.calculateInsuranceAmount(month);
                                System.out.println("Name :"+name);
                                System.out.println("Email Id :"+email);
                                System.out.println("Amount to be paid
 :"+amount);
                        }
                        else
                        {

                                System.out.println("Provide valid Policy Id");
                        }
                }
                else if(id.contains("FAMILY"))
                {
                        FamilyInsurancePolicy g=new
FamilyInsurancePolicy(name,id,age,mnum,email);
                        if(g.validatePolicyId())
                        {
                                System.out.println("Enter number of
members");
                                int num=sc.nextInt();

amount=g.calculateInsuranceAmount(month,num);
                                System.out.println("Name :"+name);
                                System.out.println("Email Id :"+email);
                                System.out.println("Amount to be paid
 :"+amount);
                        }
                        else
                        {
                                System.out.println("Provide valid Policy Id");
                        }
                }
                else if(id.contains("SENIOR"))
                {
                        SeniorCitizenPolicy g=new
SeniorCitizenPolicy(name,id,age,mnum,email);
                        if(g.validatePolicyId())
                        {
                                System.out.println("Enter number of
members");
                                int num=sc.nextInt();
```

```
amount=g.calculateInsuranceAmount(month,num);
                                        System.out.println("Name :"+name);
                                        System.out.println("Email Id :"+email);
                                        System.out.println("Amount to be paid
 :"+amount);
                                }
                                else
                                {
                                        System.out.println("Provide valid Policy Id");
                                }

                        }
                        else
                                System.out.println("Provide valid Policy Id");
                }


          }
```