LAB-4

1a)

Create an abstract class named Book. Include a String field for the book's title and a double field for the book's price. Within the class, include a constructor that requires the book title, and add two get methods—one that returns the title and one that returns the price. Include an abstract method named setPrice(). Create two child classes of Book: Fiction and NonFiction. Each must include a setPrice() method that sets the price for all Fiction Books to $24.99 and for all NonFiction Books to $37.99. Write a constructor for each subclass, and include a call to setPrice() within each. Write an application demonstrating that you can create both a Fiction and a NonFiction Book, and display their fields. Save the files as Book.java, Fiction.java, NonFiction.java, and UseBook.java.

1b)

Write an application named BookArray in which you create an array that holds 10 Books, some Fiction and some NonFiction. Using a for loop, display details about all

10 books. Save the file as BookArray.java.

CODE:

1A)

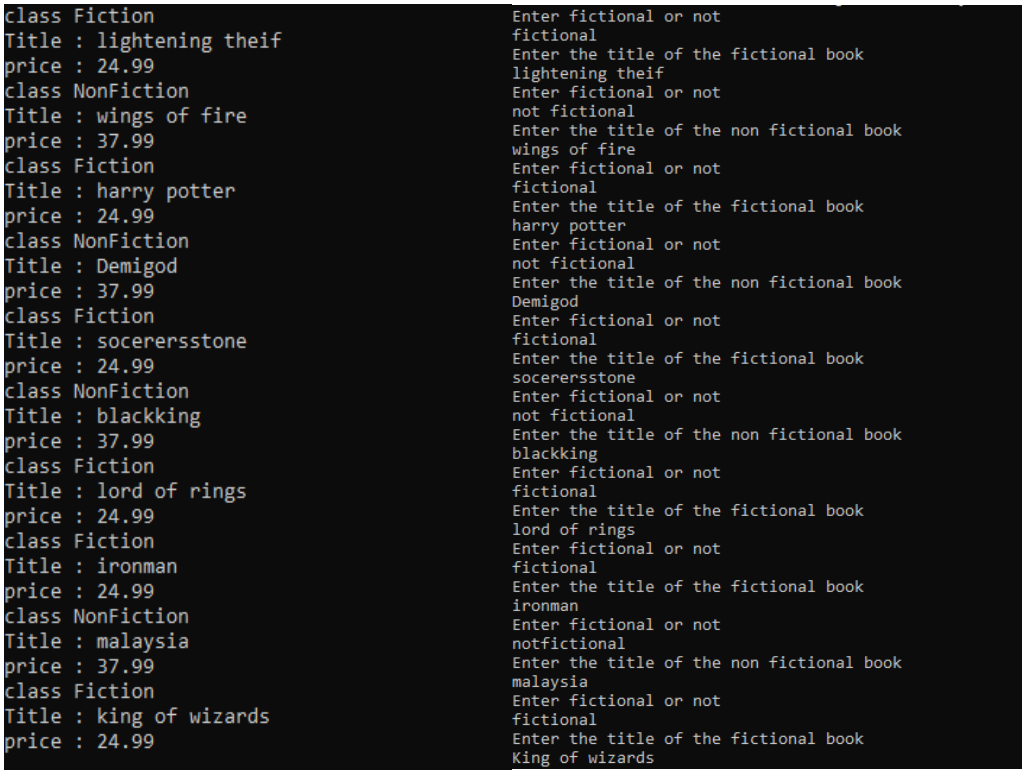
```
import java.util.*; public class UseBook {
public static void main(String args[])
{
Scanner sc=new Scanner(System.in); System.out.println("Enter the title of the book"); String s=sc.nextLine();
System.out.println("Enter Whether the book is fictional or not"); String s2=sc.nextLine();
if(s2.charAt(0)=='F'||s2.charAt(0)=='f')
{
Fiction f=new Fiction(s); f.setPrice();
System.out.println("Title of the fictional book is "+f.gettitle());
System.out.println("price of the fictional book is "+f.getprice());
}
else if(s2.charAt(0)=='N'||s2.charAt(0)=='n')
{
NonFiction nf=new NonFiction(s); nf.setPrice();
System.out.println("Title of the Non fictional book is "+nf.gettitle());

}
```

```
else
{

}

System.out.println("price of the non fictional book is "+nf.getprice());



System.out.println("Wrong input given");

sc.close();
}
}
abstract class Book
{
String title; double price;
abstract void setPrice(); Book(String title)
{
this.title=title;
}
String gettitle()
{
return title;
}
double getprice()
{
return price;
}
}
class Fiction extends Book
{
Fiction(String title)
{
super(title);
}
void setPrice()
{
price=24.99;
}
}
class NonFiction extends Book
{
```

```
 NonFiction(String title)
 {
 super(title);
 }
 void setPrice()
 {
 price=37.99;
 }
}


1B)
 import java.util.*;
 public class
 BookArray
 {
  public static void main(String args[])
  {
   Scanner sc=new
   Scanner(System.in);String
   ba[]=new String[10];
   for(int i=0;i<10;i++)
   {
    System.out.println("Enter fictional or
    not");String b1=sc.nextLine();
    if(b1.charAt(0)=='f'||b1.charAt(0)=
    ='F')
    {
     System.out.println("Enter the title of the
     fictional book");String title=sc.nextLine();
     Fiction f=new
     Fiction(title);
     f.setPrice();
     ba[i]=title+" - Fictional ";
    }
    else
    {
     System.out.println("Enter the title of the non
     fictional book");String title=sc.nextLine();
     NonFiction nf=new
     NonFiction(title);nf.setPrice();
     ba[i]=title+" - notfictional";
    }
   }
```

```
    for(int l=0;l<10;l++)
    {
     System.out.println(ba[l]);
}
 }
```

OUTPUT:



2a)

The Talk-A-Lot Cell Phone Company provides phone services for its customers. Create an abstract class named PhoneCall that includes a String field for a phone number and a double field for the price of the call. Also include a constructor that requires a phone number parameter and that sets the price to 0.0. Include a set method for the price. Also include three abstract get methods—one that returns the phone number, another that returns the price of the call, and a third that displays information about the call. Create two child classes of PhoneCall: IncomingPhoneCall and OutgoingPhoneCall. The IncomingPhoneCall constructor passes its phone number parameter to its parent's constructor and sets the price of the call to 0.02. The method that displays the phone call information displays the phone number, the rate, and the price of the call (which is the same as the rate). The OutgoingPhoneCall class includes an additional field that holds the time of the call in minutes. The constructor requires both a phone number and the time. The price is 0.04 per minute, and the display method shows the details of the call, including the phone number, the rate per minute, the number of

minutes, and the total price. Write an application that demonstrates you can instantiate and display both IncomingPhoneCall and OutgoingPhoneCall objects. Save the files as PhoneCall.java, IncomingPhoneCall.java, OutgoingPhoneCall.java, and DemoPhoneCalls.java.

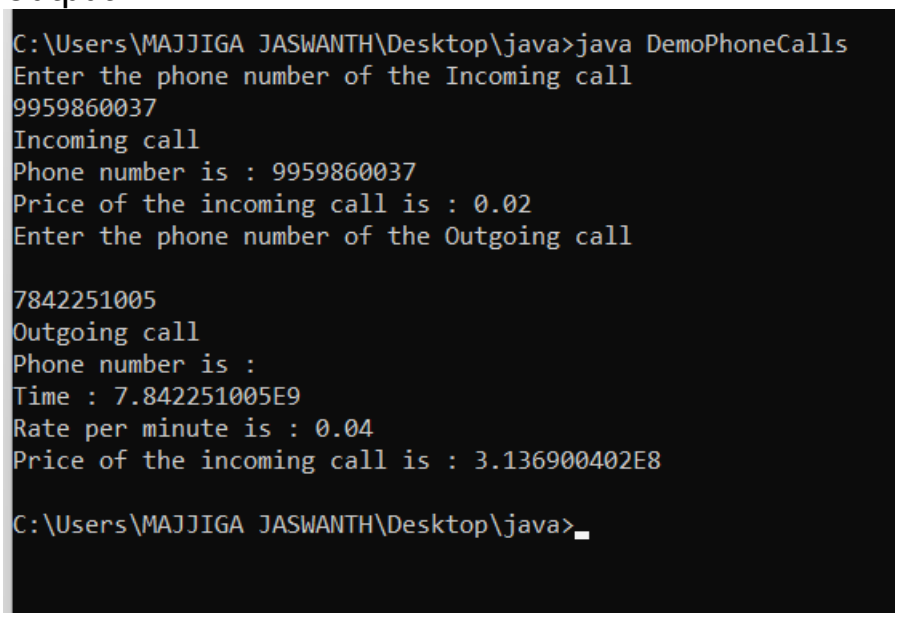
CODE:

```
import java.util.*;
public class DemoPhoneCalls {
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the phone number of the
		Incoming call");String phnno=sc.nextLine();
		IncomingPhoneCall ipc=new
		IncomingPhoneCall(phnno);ipc.display();
		System.out.println("Enter the phone number of the
		Outgoing call");String phnno1=sc.nextLine();
		double time=sc.nextDouble();
		OutgoingPhoneCall opc=new
		OutgoingPhoneCall(phnno1,time);opc.display();
		sc.close();
	}
}
abstract class
	PhoneCall{
	String
	phnnum;
	double price;
	PhoneCall(String phnno)
	{
		phnnum=phn
		no;price=0.0;
	}
	abstract String
	getphnnum();abstract
	double getprice();
	abstract void display();
}
class IncomingPhoneCall extends PhoneCall
{
	IncomingPhoneCall(String phnno)
	{
		super(phnn
		o);
		price=0.02
```

```
                ;
        }
        public String getphnnum()
        {
                return phnnum;
        }
        public double getprice()
        {
                return price;
        }
        public void display()
        {
                System.out.println("Incoming call");
                System.out.println("Phone number is :
                "+getphnnum()); System.out.println("Price of the
                incoming call is : "+getprice());
        }
}
class OutgoingPhoneCall extends PhoneCall
{
        double time;
        OutgoingPhoneCall(String phnno,double time)
        {
                super(phnno);
                this.time=tim
                e;
                price=time*0.
                04;
        }
        public String getphnnum()
        {
                return phnnum;
        }
        public double getprice()
        {
                return price;
        }
        public void display()
        {
                System.out.println("Outgoing call");
                System.out.println("Phone number is :
                "+getphnnum());System.out.println("Time :
                "+time); System.out.println("Rate per minute is :
                0.04");
```

```
            System.out.println("Price of the incoming call is : "+getprice());
        }
}
```

Output:



2b)
Write an application in which you assign data to a mix of eight IncomingPhoneCall and OutgoingPhoneCall objects into an array. Use a for loop to display the data. Save the file as PhoneCallArray.java.
Code:

```
import java.util.*;
public class PhoneCallArray {
public static void main(String args[])
{
Scanner sc=new Scanner(System.in); PhoneCall a[]=new PhoneCall[8]; for(int
i=0;i<8;i++)
{
System.out.println("Enter incoming or outgoing"); String b1=sc.nextLine();
if(b1.charAt(0)=='i'||b1.charAt(0)=='I')
{
System.out.println("Enter the phonenumber to the incoming call"); String
title=sc.nextLine();
IncomingPhoneCall ipc=new IncomingPhoneCall(title); a[i]=ipc;
}
else
{


call");
```

```
System.out.println("Enter the phonenumber and the time taken for the
outgoing

String title=sc.nextLine(); double time=sc.nextDouble();

String num=sc.nextLine();

OutgoingPhoneCall opc=new OutgoingPhoneCall(title,time); a[i]=opc;
}
}
for(int i=0;i<8;i++)
{
a[i].display();
}
sc.close();
}
}
```

Output:

```
C:\Users\MAJJIGA JASWANTH\Desktop\java>javac PhoneCallArray.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac PhoneCallArray.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>java PhoneCallArray
Enter incoming or outgoing
9959860037
Enter the phonenumber and the time taken for the outgoing call
7842251005
2
Enter incoming or outgoing
incoming
Enter the phonenumber to the incoming call
8985050021
Enter incoming or outgoing
outgoing
Enter the phonenumber and the time taken for the outgoing call
8985050021
56
Enter incoming or outgoing
outgoing
Enter the phonenumber and the time taken for the outgoing call
5489568785
8
Enter incoming or outgoing
4587458745
Enter the phonenumber and the time taken for the outgoing call
4587458745
9
Enter incoming or outgoing
incoming
Enter the phonenumber to the incoming call
8787878787
Enter incoming or outgoing
incoming
Enter the phonenumber to the incoming call
9595959595
Enter incoming or outgoing
outgoing
Enter the phonenumber and the time taken for the outgoing call
2525262622
89
Outgoing call
Phone number is : 7842251005
Time : 2.0
Rate per minute is : 0.04
Price of the incoming call is : 0.08
Incoming call
Phone number is : 8985050021
```

```
Price of the incoming call is : 0.02
Outgoing call
Phone number is : 8985050021
Time : 56.0
Rate per minute is : 0.04
Price of the incoming call is : 2.24
Outgoing call
Phone number is : 5489568785
Time : 8.0
Rate per minute is : 0.04
Price of the incoming call is : 0.32
Outgoing call
Phone number is : 4587458745
Time : 9.0
Rate per minute is : 0.04
Price of the incoming call is : 0.36
Incoming call
Phone number is : 8787878787
Price of the incoming call is : 0.02
Incoming call
Phone number is : 9595959595
Price of the incoming call is : 0.02
Outgoing call
Phone number is : 2525262622
Time : 89.0
Rate per minute is : 0.04
Price of the incoming call is : 3.56

C:\Users\MAJJIGA JASWANTH\Desktop\java>
```

3a)

Create an abstract class called GeometricFigure. Each figure includes a height, a width, a figure type, and an area. Include an abstract method to determine the area of the figure. Create two subclasses called Square and Triangle. Create an application that demonstrates creating objects of both subclasses, and store them in an array. Save the files as GeometricFigure.java, Square.java, Triangle.java, and UseGeometric.java.

CODE:

```
import java.util.*;
abstract class GeometricFigure
{
        String
        figuretype;int
        height;
        int width;
        GeometricFigure(String type,int hght,int width)
        {
                figuretype=type;
```

```
                height=hght;
                this.width=width
                ;
        }
        abstract void area();
}
class Square extends GeometricFigure
{
        Square(String type,int side)
        {
                super(type,side,side);
        }
        void area()
        {
                System.out.println("The area of "+figuretype+" is
"+((double)(height)*(double)(height)));
        }
}
class Triangle extends GeometricFigure
{
        Triangle(String type,int hght,int width)
        {
                super(type,hght,width);
        }
        void area()
        {
                System.out.println("The area of "+figuretype+" is
"+(0.5*(double)(height)*(double)(width)));
        }
}
Import java.util.*;
public class UseGeometric
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter the side of the square");
                int s=sc.nextInt();
                Square s2=new
                Square("Square",s);s2.area();
                System.out.println("Enter the height and width of the triangle");
                int hght=sc.nextInt();
                int width=sc.nextInt();
                Triangle t=new
                Triangle("Triangle",hght,width);t.area();
        }
}
```

Output:

```
C:\Users\MAJJIGA JASWANTH\Desktop\java>javac GeometricFigure.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac Square.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac Traingle.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac UseGeometric.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>java UseGeometric
Enter the side of the square
4
The area of Square is 16.0
Enter the height and width of the triangle
4
4
The area of Triangle is 8.0

C:\Users\MAJJIGA JASWANTH\Desktop\java>
```

3b)

Create an application that demonstrates creating objects of both subclasses and store them in an array. Save the files as GeometricFigure.java, Square.java, Triangle.java, and UseGeometric.java

Code:

```
import java.util.*;
interface
SidedObject
{
 public void displaySides();
}
abstract class GeometricFigure2 implements SidedObject
{
        String
        figuretype;int
        height;
        int width;
        GeometricFigure2(String type,int hght,int width)
        {
                figuretype=type;
                height=hght;
                this.width=width
                ;
        }
        abstract void area();
}
```

```
class Square2 extends GeometricFigure2
{
	Square2(String type,int side)
	{
		super(type,side,side);
	}
	void area()
	{
		System.out.println("The area of "+figuretype+" is
"+((double)(height)*(double)(height)));
	}
 public void displaySides()
 {
  System.out.println("No of sides : 4");
 }
}
class Triangle2 extends GeometricFigure2
{
	Triangle2(String type,int hght,int width)
	{
		super(type,hght,width);
	}
	void area()
	{
		System.out.println("The area of "+figuretype+" is
"+(0.5*(double)(height)*(double)(width)));
	}
 public void displaySides()
 {
  System.out.println("No of sides : 3");
 }
}
public class UseGeometric2
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the side of the square");
		int s=sc.nextInt();
		Square2 s2=new
		Square2("Square",s);s2.area();
  s2.displaySides();
		System.out.println("Enter the height and width of the triangle");
		int hght=sc.nextInt();
		int width=sc.nextInt();
		Triangle2 t=new
		Triangle2("Triangle",hght,width);t.area();
  t.displaySides();
	}
}
```

Output:

```
C:\Users\MAJJIGA JASWANTH\Desktop\java>javac GeometricFigure1.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac Square1.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac Traingle1.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>javac UseGeometric1.java

C:\Users\MAJJIGA JASWANTH\Desktop\java>java UseGeometric1
Enter the side of the square
6
The area of Square is 36.0
No of sides : 4
Enter the height and width of the triangle
58
42
The area of Triangle is 1218.0
No of sides : 3

C:\Users\MAJJIGA JASWANTH\Desktop\java>
```

Majjiga jaswanth
20bcd7171