

**EXPT: 1****DATE:****PROGRAM USING CLASSES & OBJECTS****AIM:**

To develop a java application to generate the electricity bill using classes and objects.

**ALGORITHM:**

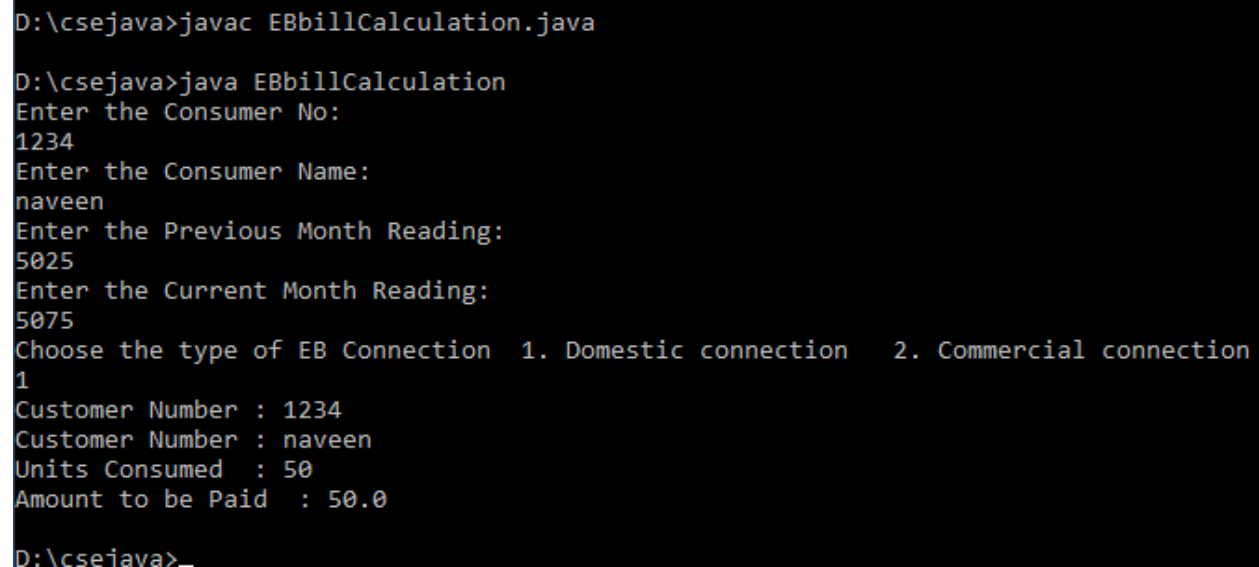
1. Start the program. Type the program in notepad and store with class name and with .java extension.
2. Create a class named EBillCalculation with the necessary variables.
3. Get input for consumer number, name, previous and current meter readings.
4. Calculate units consumed as current reading minus previous reading.
5. Ask the user to select the type of connection: Domestic or Commercial.
6. Based on the type, calculate the bill using slab-wise rates.
7. Domestic: ₹1/unit up to 100, ₹2.50/unit for next 100, ₹4/unit up to 500 and ₹6/unit beyond 500.
8. Commercial: ₹2/unit up to 100, ₹4.50/unit for next 100, ₹6/unit up to 500 and ₹7/unit beyond 500.
9. Display consumer details, units consumed, and the total bill amount.
10. Compile and run the program using the following commands in command prompt:  
  
    To compile: javac ex1.java  
  
    To run: java ex1
11. End the program.

**PROGRAM:**

```
import java.io.*;
import java.util.*;
class EBbillCalculation
{
int cno;
String cname;
int pm_reading,cm_reading,units; double billpay;
void getdetails()
{
Scanner in = new Scanner(System.in);
System.out.println("Enter the Consumer No:");
cno=in.nextInt();
System.out.println("Enter the Consumer Name:");
cname=in.next();
System.out.println("Enter the Previous Month Reading:");
pm_reading=in.nextInt();
System.out.println("Enter the Current Month Reading:");
cm_reading=in.nextInt();
units= cm_reading-pm_reading;
System.out.println("Choose the type of EB Connection 1.Domestic
connection 2. Commercial connection");
int ch=in.nextInt();
switch(ch)
{
case 1:
Domestic_Calculation(units);
break;
case 2:
Commercial_Calculation(units);
break;
}
}
void Domestic_Calculation(int units)
{
billpay = 0; if(units<=100)
{
billpay=units*1.00;
}
```

```
else if(units>100 && units<=200)
{
billpay=100*1.00+(units-100)*2.50;
}
else if(units>200 && units<=500)
{
billpay=100*1.00+200*2.50+(units-200)*4.00;
}
else if(units>500)
{
billpay =100*1.00+200*2.50+500*4.00+(units-500)*6.00;
}
show();
System.out.println("Amount to be Paid : " + billpay);
}
void Commercial_Calculation(int units)
{
billpay = 0; if(units<=100)
{
billpay=units*2.00;
}
else if(units>100 && units<=200)
{
billpay=100*2.00+(units-100)*4.50;
}
else if(units>200 && units<=500)
{
billpay=100*2.00+200*4.50+(units-200)*6.00;
}
else if(units>500)
{
billpay =100*2.00+200*4.50+500*6.00+(units-500)*7.00;
}
show();
System.out.println("Amount to be Paid : " + billpay);
}
public void show()
{
System.out.println("Customer Number : " + cno);
System.out.println("Customer Name : " + cname);
```

```
System.out.println("Units Consumed : " + units);
}
}
class ex1
{
public static void main(String[] args)
{
EBbillCalculation eb=new EBbillCalculation();
eb.getdetails();
}
}
```

**OUTPUT:**

```
D:\csejava>javac EBbillCalculation.java

D:\csejava>java EBbillCalculation
Enter the Consumer No:
1234
Enter the Consumer Name:
naveen
Enter the Previous Month Reading:
5025
Enter the Current Month Reading:
5075
Choose the type of EB Connection  1. Domestic connection  2. Commercial connection
1
Customer Number : 1234
Customer Number : naveen
Units Consumed   : 50
Amount to be Paid : 50.0

D:\csejava>
```

**RESULT:**

Thus the java application for generating the electricity bill is created and executed successfully.

**EXPT:2****DATE:****PROGRAM USING PACKAGES****AIM:**

To develop a java application to implement currency converter, distance converter and time converter using the concept of packages.

**ALGORITHM:**

1. Start the program.
2. Create three packages:
3. currencyconversion which contains ex2acurrency class with methods for currency conversion.
4. distanceconversion which contains ex2distance class with methods for distance conversion.
5. timeconversion which contains ex2timer class with methods for time conversion.
6. In each class, define appropriate methods for conversion.
7. Create a converter class with main method.
8. Import the packages and create objects for each class.
9. Display a menu to the user with conversion options.
10. Based on the user's choice, call the corresponding method.
11. Repeat until the user chooses to exit.
12. Run the following commands:  

```
javac -d . ex2acurrency.java
javac -d . ex2distance.java
javac -d . ex2timer.java
javac converter.java
java converter
```
13. End the program.

**PROGRAM:****currency.java**

```
package currencyconversion;
import java.util.*;
public class ex2acurrency
{
double inr,usd;
double euro,yen;
Scanner in=new Scanner(System.in);
public void dollartorupee()
{
System.out.println("Enter dollars to convert into Rupees:");
usd=in.nextInt();
inr=usd*67;
System.out.println("Dollar =" +usd+"equal to INR="+inr);
}
public void rupeetodollar()
{
System.out.println("Enter Rupee to convert into Dollars:");
inr=in.nextInt();
usd=inr/67;
System.out.println("Rupee =" +inr+"equal to Dollars="+usd);
}
public void eurotorupee()
{
System.out.println("Enter euro to convert into Rupees:");
euro=in.nextInt();
inr=euro*79.50;
System.out.println("Euro =" +euro +"equal to INR="+inr);
}
public void rupeetoeuro()
{
System.out.println("Enter Rupees to convert into Euro:");
inr=in.nextInt();
euro=(inr/79.50);
System.out.println("Rupee =" +inr +"equal to Euro="+euro);
}
public void yentorupee()
{
System.out.println("Enter yen to convert into Rupees:");
```

```
yen=in.nextInt();
inr=yen*0.61;
System.out.println("YEN="+yen +"equal to INR="+inr);
}
public void rupeetoyen()
{
System.out.println("Enter Rupees to convert into Yen:");
inr=in.nextInt();
yen=(inr/0.61);
System.out.println("INR="+inr +"equal to YEN"+yen);
}
}
```

**distance.java**

```
package distanceconversion;
import java.util.*;
public class ex2distance
{
double km,m,miles;
Scanner sc = new Scanner(System.in);
public void kmtom()
{
System.out.print("Enter in km ");
km=sc.nextDouble();
m=(km*1000);
System.out.println(km+"km" +"equal to"+m+"metres");
}
public void mtokm()
{
System.out.print("Enter in meter ");
m=sc.nextDouble();
km=(m/1000);
System.out.println(m+"m" +"equal to"+km+"kilometres");
}
public void milestokm()
{
System.out.print("Enter in miles");
miles=sc.nextDouble();
km=(miles*1.60934);
System.out.println(miles+"miles" +"equal to"+km+"kilometres");
}
}
```

```
public void kmtomiles()
{
    System.out.print("Enter in km");
    km=sc.nextDouble();
    miles=(km*0.621371);
    System.out.println(km+"km" +"equal to"+miles+"miles");
}
}
```

**timer.java**

```
package
timeconversion;
import java.util.*;
public class
ex2timer
{
    int hours,seconds,minutes;
    int input;
    Scanner sc = new Scanner(System.in);
    public void secondstohours()
    {
        System.out.print("Enter the number of seconds: ");
        input = sc.nextInt();
        hours = input / 3600;
        minutes = (input % 3600) / 60;
        seconds = (input % 3600) % 60;
        System.out.println("Hours: " + hours);
        System.out.println("Minutes: " + minutes);
        System.out.println("Seconds: " + seconds);
    }
    public void minutestohours()
    {
        System.out.print("Enter the number of minutes: ");
        minutes=sc.nextInt();
        hours=minutes/60;
        minutes=minutes%60;
        System.out.println("Hours: " + hours);
        System.out.println("Minutes: " + minutes);
    }
    public void hourstominutes()
    {
```



```
System.out.println("enter the no of hours");
hours=sc.nextInt();
minutes=(hours*60);
System.out.println("Minutes: " + minutes);
}
public void hourstoseconds()
{
System.out.println("enter the no of hours");
hours=sc.nextInt();
seconds=(hours*3600);
System.out.println("Minutes: " + seconds);
}
}
```

### **converter.java**

```
import java.util.*;
import java.io.*;
import currencyconversion.ex2acurrency;
import distanceconversion.ex2distance;
import timeconversion.ex2timer;
class converter
{
public static void main(String args[])
{
Scanner s=new Scanner(System.in);
int choice,ch;
ex2acurrency c=new ex2acurrency();
ex2distance d=new ex2distance();
ex2timer t=new ex2timer();
do
{
System.out.println("1.dollar to rupee ");
System.out.println("2.rupee to dollar ");
System.out.println("3.Euro to rupee ");
System.out.println("4..rupee to Euro ");
System.out.println("5.Yen to rupee ");
System.out.println("6.Rupee to Yen ");
System.out.println("7.Meter to kilometer ");
System.out.println("8.kilometer to meter ");
System.out.println("9.Miles to kilometer ");
System.out.println("10.kilometer to miles");
```

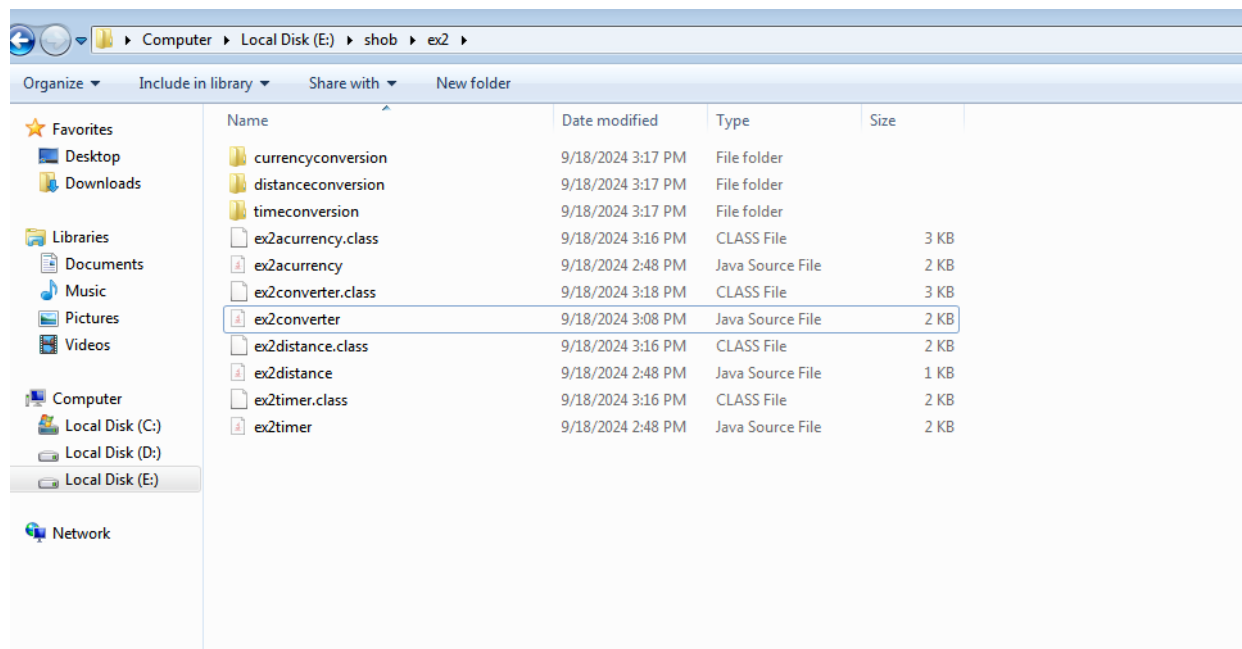
```
System.out.println("11.Hours to Minutes");
System.out.println("12.Hours to Seconds");
System.out.println("13.Seconds to Hours");
System.out.println("14.Minutes to Hours");
System.out.println("Enter ur choice");
choice=s.nextInt();
switch(choice)
{
case 1:
{
c.dollartorupee();
break;
}
case 2:
{
c.rupeetodollar();
break;
}
case 3:
{
c.eurotorupee();
break;
}
case 4:
{
c.rupeetoeuro();
break;
}
case 5:
{c.yentorupee();
break;}
case 6 :
{
c.rupeetoyen();
break;
}
case 7 :
{
d.mtokm();
break;
```

```
    }  
    case 8 :  
    {  
    d.kmtom();  
    break;  
    }  
    case 9 :  
    {  
    d.milestokm();  
    break;  
    }  
    case 10 :  
    {  
    d.kmtomiles();  
    break;  
    }  
    case 11 :  
    {  
    t.hourstominutes();  
    break;  
    }  
    case 12 :  
    {  
    t.hourstoseconds();  
    break;  
    }  
    case 13 :  
    {  
    t.secondstohours();  
    break;  
    }  
    case 14 :  
    {  
    t.minutestohours();  
    break;  
    }  
    }  
    System.out.println("Enter 0 to quit and 1 to continue ");  
    ch=s.nextInt();  
    }while(ch==1);  
    }  
}
```

**OUTPUT:**

```
E:\shob\ex2>javac -d . ex2acurrency.java
E:\shob\ex2>javac -d . ex2timer.java
E:\shob\ex2>javac -d . ex2distance.java
error: invalid flag: .
Usage: javac <options> <source files>
use --help for a list of possible options
E:\shob\ex2>javac -d . ex2distance.java
E:\shob\ex2>javac exconverter.java
error: file not found: exconverter.java
Usage: javac <options> <source files>
use --help for a list of possible options
E:\shob\ex2>javac ex2converter.java
E:\shob\ex2>java ex2converter
1.dollar to rupee
2.rupee to dollar
3.Euro to rupee
4..rupee to Euro
5.Yen to rupee
6.Rupee to Yen
7.Meter to kilometer
8.kilometer to meter
9.Miles to kilometer
10.kilometer to miles
11.Hours to Minutes
12.Hours to Seconds
13.Seconds to Hours
14.Minutes to Hours
Enter ur choice
1
Enter dollars to convert into Rupees:
2000000
Dollar =2000000.0equal to INR=1.34E7
Enter 0 to quit and 1 to continue
1
1.dollar to rupee
2.rupee to dollar
3.Euro to rupee
4..rupee to Euro
5.Yen to rupee
6.Rupee to Yen
7.Meter to kilometer
8.kilometer to meter
9.Miles to kilometer
10.kilometer to miles
11.Hours to Minutes
12.Hours to Seconds
13.Seconds to Hours
14.Minutes to Hours
Enter ur choice
7
Enter in meter 40000
40000.0equal to40.0kilometres
Enter 0 to quit and 1 to continue
1
```

```
Enter 0 to quit and 1 to continue
1
1.dollar to rupee
2.rupee to dollar
3.Euro to rupee
4..rupee to Euro
5.Yen to rupee
6.Rupee to Yen
7.Meter to kilometer
8.kilometer to meter
9.Miles to kilometer
10.kilometer to miles
11.Hours to Minutes
12.Hours to Seconds
13.Seconds to Hours
14.Minutes to Hours
Enter ur choice
14
Enter the number of minutes: 180
Hours: 3
Minutes: 0
Enter 0 to quit and 1 to continue
```



## RESULT:

Thus the java application to implement currency converter, distance converter and time converter was implemented and executed successfully.

**EXPT:3****DATE:****PROGRAM USING INHERITANCE****AIM:**

To develop a java application to generate pay slip for different category of employees using the concept of inheritance.

**ALGORITHM:**

1. Start the program.
2. Create a base class employee with attributes: name, ID, address, email, and mobile number.
3. Add methods to get and display employee details.
4. Create subclasses named programmer, asstprofessor, associateprofessor, and professor extending employee.
5. In each subclass, input basic pay and calculate DA (97%), HRA (10%), PF (12%), and club fund (0.1%) of basic pay.
6. Calculate gross salary and net salary, then display the pay slip.
7. In the main method, create the objects for the inherited classes and invoke the necessary methods to display the Pay slip. show a menu to select employee type.
8. Based on the choice, create an object, get details, and generate the pay slip.
9. Repeat until the user chooses to exit.
10. End the program.

**PROGRAM:**

```
import java.util.*;
import java.io.*;
class employee
{
    int empid;
    long mobile;
    String name, address, mailid;
    Scanner get = new Scanner(System.in);
    void getdata()
    {
        System.out.println("Enter Name of the Employee");
        name = get.nextLine();
        System.out.println("Enter Mail id");
        mailid = get.nextLine();
        System.out.println("Enter Address of the Employee:");
        address = get.nextLine();
        System.out.println("Enter employee id ");
        empid = get.nextInt();
        System.out.println("Enter Mobile Number");
        mobile = get.nextLong();
    }
    void display()
    {
        System.out.println("Employee Name: "+name);
        System.out.println("Employee id : "+empid);
        System.out.println("Mail id : "+mailid);
        System.out.println("Address: "+address);
        System.out.println("Mobile Number: "+mobile);
    }
}
class programmer extends employee
{
    double salary,bp,da,hra,pf,club,net,gross;
    void calculateprog()
    {
        System.out.println("Enter basic pay");
        bp = get.nextDouble();
        da=(0.97*bp);
        hra=(0.10*bp);
```

```
pf=(0.12*bp);
club=(0.1*bp);
gross=(bp+da+hra);
net=(gross-pf-club);
System.out.println("*****");
System.out.println("PAY SLIP FOR PROGRAMMER");
System.out.println("*****");
System.out.println("Basic Pay:Rs"+bp);
System.out.println("DA:Rs"+da);
System.out.println("PF:Rs"+pf);
System.out.println("HRA:Rs"+hra);
System.out.println("CLUB:Rs"+club);
System.out.println("GROSS PAY:Rs"+gross);
System.out.println("NET PAY:Rs"+net);
}
}
class asstprofessor extends employee
{
double salary,bp,da,hra,pf,club,net,gross;
void calculateasst()
{
System.out.println("Enter basic pay");
bp = getNextDouble();
da=(0.97*bp);
hra=(0.10*bp);
pf=(0.12*bp);
club=(0.1*bp);
gross=(bp+da+hra);
net=(gross-pf-club);
System.out.println("*****");
System.out.println("PAY SLIP FOR ASSISTANT PROFESSOR");
System.out.println("*****");
System.out.println("Basic Pay:Rs"+bp);
System.out.println("DA:Rs"+da);
System.out.println("HRA:Rs"+hra);
System.out.println("PF:Rs"+pf);
System.out.println("CLUB:Rs"+club);
System.out.println("GROSS PAY:Rs"+gross);
System.out.println("NET PAY:Rs"+net);
}
```



```
}
class associateprofessor extends employee
{
double salary,bp,da,hra,pf,club,net,gross;
void calculateassociate()
{
System.out.println("Enter basic pay");
bp = get.nextDouble();
da=(0.97*bp);
hra=(0.10*bp);
pf=(0.12*bp);
club=(0.1*bp);
gross=(bp+da+hra);
net=(gross-pf-club);
System.out.println("*****");
System.out.println("PAY SLIP FOR ASSOCIATE PROFESSOR");
System.out.println("*****");
System.out.println("Basic Pay:Rs"+bp);
System.out.println("DA:Rs"+da);
System.out.println("HRA:Rs"+hra);
System.out.println("PF:Rs"+pf);
System.out.println("CLUB:Rs"+club);
System.out.println("GROSS PAY:Rs"+gross);
System.out.println("NET PAY:Rs"+net);
}
}
class professor extends employee
{
double salary,bp,da,hra,pf,club,net,gross;
void calculateprofessor()
{
System.out.println("Enter basic pay");
bp = get.nextDouble();
da=(0.97*bp);
hra=(0.10*bp);
pf=(0.12*bp);
club=(0.1*bp);
gross=(bp+da+hra);
net=(gross-pf-club);
System.out.println("*****");
```

```
System.out.println("PAY SLIP FOR PROFESSOR");
System.out.println("*****");
System.out.println("Basic Pay:Rs"+bp);
System.out.println("DA:Rs"+da);
System.out.println("HRA:Rs"+hra);
System.out.println("PF:Rs"+pf);
System.out.println("CLUB:Rs"+club);
System.out.println("GROSS PAY:Rs"+gross);
System.out.println("NET PAY:Rs"+net);
}
}
class ex3
{
public static void main(String args[])
{
int choice,cont;
do
{
System.out.println("PAYROLL");
System.out.println(" 1.PROGRAMMER \t 2.ASSISTANT PROFESSOR \t 3.ASSOCIATE
PROFESSOR \t 4.PROFESSOR ");
Scanner c = new Scanner(System.in);
choice=c.nextInt();
switch(choice)
{
case 1:
{
programmer p=new programmer();
p.getdata();
p.display();
p.calculateprog();
break;
}
case 2:
{
asstprofessor asst=new asstprofessor();
asst.getdata();
asst.display();
asst.calculateasst();
break;
```

```
    }  
    case 3:  
    {  
        associateprofessor asso=new associateprofessor();  
        asso.getdata();  
        asso.display();  
        asso.calculateassociate();  
        break;  
    }  
    case 4:  
    {  
        professor prof=new professor();  
        prof.getdata();  
        prof.display();  
        prof.calculateprofessor();  
        break;  
    }  
    }  
    System.out.println("Do u want to continue 0 to quit and 1 to continue ");  
    cont=c.nextInt();  
    }while(cont==1);  
    }  
    }
```

**OUTPUT:**

```
C:\Users\Administrator>cd..
C:\Users>cd..
C:\>E:
E:\>cd E:\shob
E:\shob>javac ex3.java
E:\shob>java ex3
PAYROLL
1.PROGRAMMER      2.ASSISTANT PROFESSOR    3.ASSOCIATE PROFESSOR    4.PROFESSOR
1
Enter Name of the Employee
Abinaya
Enter Mail id
abi@saec.ac.in
Enter Address of the Employee:
Mogappair
Enter employee id
002
Enter Mobile Number
9080706945
Employee Name: Abinaya
Employee id : 2
Mail id : abi@saec.ac.in
Address: Mogappair
Mobile Number: 9080706945
Enter basic pay
80000
*****
PAY SLIP FOR PROGRAMMER
*****
Basic Pay:Rs80000.0
DA:Rs77600.0
PF:Rs9600.0
HRA:Rs8000.0
CLUB:Rs8000.0
GROSS PAY:Rs165600.0
NET PAY:Rs148000.0
Do u want to continue 0 to quit and 1 to continue
/
```

**RESULT:**

Thus the java application to generate pay slips for different categories of employees was implemented using inheritance and the program was executed successfully.

**EXPT: 4****DATE:****PROGRAM USING ABSTRACT CLASS AND POLYMORPHISM****AIM:**

To write a java program to create an abstract class named Shape .

**ALGORITHM:**

1. Start the program.
2. Define an abstract class Shape with two variables and an abstract method Area().
3. Create subclasses: Rectangle, Triangle, and Circle, each extending Shape.
4. In each subclass, implement the Area() method to compute and display the area.
5. In the main() method:
  - a. Create an object of Rectangle and call Area().
  - b. Create an object of Triangle and call Area().
  - c. Create an object of Circle and call Area().
6. Each object will execute its version of the Area() method demonstrating polymorphism.
7. End the program.

**PROGRAM:**

```
abstract class Shape
{
double a=3,b=4;
abstract void Area();
}
class Rectangle extends Shape
{
double rectarea;
void Area()
{
rectarea=a*b;
System.out.println("Area of rectangle is: "+rectarea);
}
}
class Triangle extends Shape
{

double triarea;
void Area()
{
triarea=(0.5)*a*b;
System.out.println("Area of triangle is: "+triarea);
}
}

class Circle extends Shape
{
double circlearea;
void Area()
{

circlearea=3.14*a*a;
System.out.println("Area of circle is: "+circlearea);
}
}
public class Example
{
public static void main(String[] args)
```

```
{  
Rectangle r=new Rectangle();  
r.Area();  
Triangle t=new Triangle();  
  
t.Area();  
Circle c=new Circle();  
c.Area();  
}  
}
```

**OUTPUT:**

```
E:\shob>javac ex6.java  
  
E:\shob>java ex6  
Area of rectangle is: 12.0  
Area of triangle is: 6.0  
Area of circle is: 28.259999999999998
```

**RESULT:**

Thus the program for abstract class has been written and executed successfully.