Vivekanand Education Society's
Institute of Technology
Approved by AICTE & Affiliated to University of Mumbai

Artificial Intelligence and Data Science Department.
OOPM / Odd Sem 2021-22 / Experiment 9.

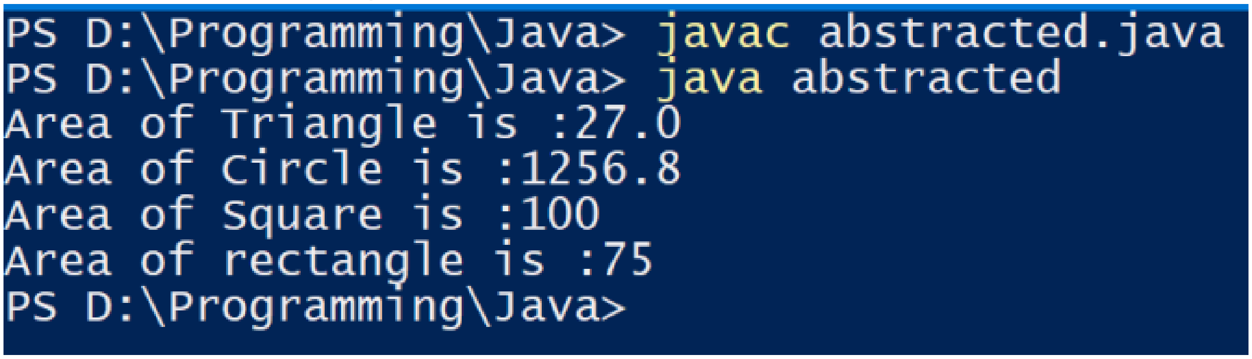
YASH SARANG.
47 / D6AD.
EXPERIMENT - 9.

AIM: To write a program to implement abstract class and abstract methods.

Program 1:

```
import java.util.*;
abstract class Shape
{
        abstract void area();
}

class triangle extends Shape
{
        int h,b;
        triangle(int h,int b)
        {
                this.h = h;
                this.b = b;
        }
        void area()
        {
                System.out.println("Area of Triangle is :" + 0.5*b*h);
        }
}

class circle extends Shape
{
        int r; circle(int r)
        {
                this.r = r;
        }
        void area()
        {
                System.out.println("Area of Circle is :" + 3.142*r*r);
        }
}

class square extends Shape
{
        int l;
        square( int l)
        {
                this.l = l;
        }
        void area()
        {
                System.out.println("Area of Square is :" + l*l);
        }
}

class rectangle extends Shape
{
        int l,b;
        rectangle( int l, int b)
        {
                this.l = l;
                this.b = b;
        }
        void area()
        {
                System.out.println("Area of rectangle is :" + l*b);
        }
}

class abstracted
{
        public static void main(String args[])
        {
                triangle T = new triangle(6,9);
                circle C = new circle(20);
                square S = new square(10);
                rectangle R = new rectangle(5,15);
                T.area();
                C.area();
                S.area();
                R.area();
        }
}
```

The output of program 1:



```
PS D:\Programming\Java> javac abstracted.java
PS D:\Programming\Java> java abstracted
Area of Triangle is :27.0
Area of Circle is :1256.8
Area of Square is :100
Area of rectangle is :75
PS D:\Programming\Java>
```

THEORY + CONCLUSION:

Abstract method & classes

Theory: i) Sometimes, it is required that the subclass should compulsarily override a method given in the superclass i.e the subclass cannot use the method of the superclass

ii) In such cases, the method of superclass, is declared to be abstract using the following general form.

```
abstract return_type method_name();
```

iii) These is no method body present since an ~~object~~ abstract method is never used by the object

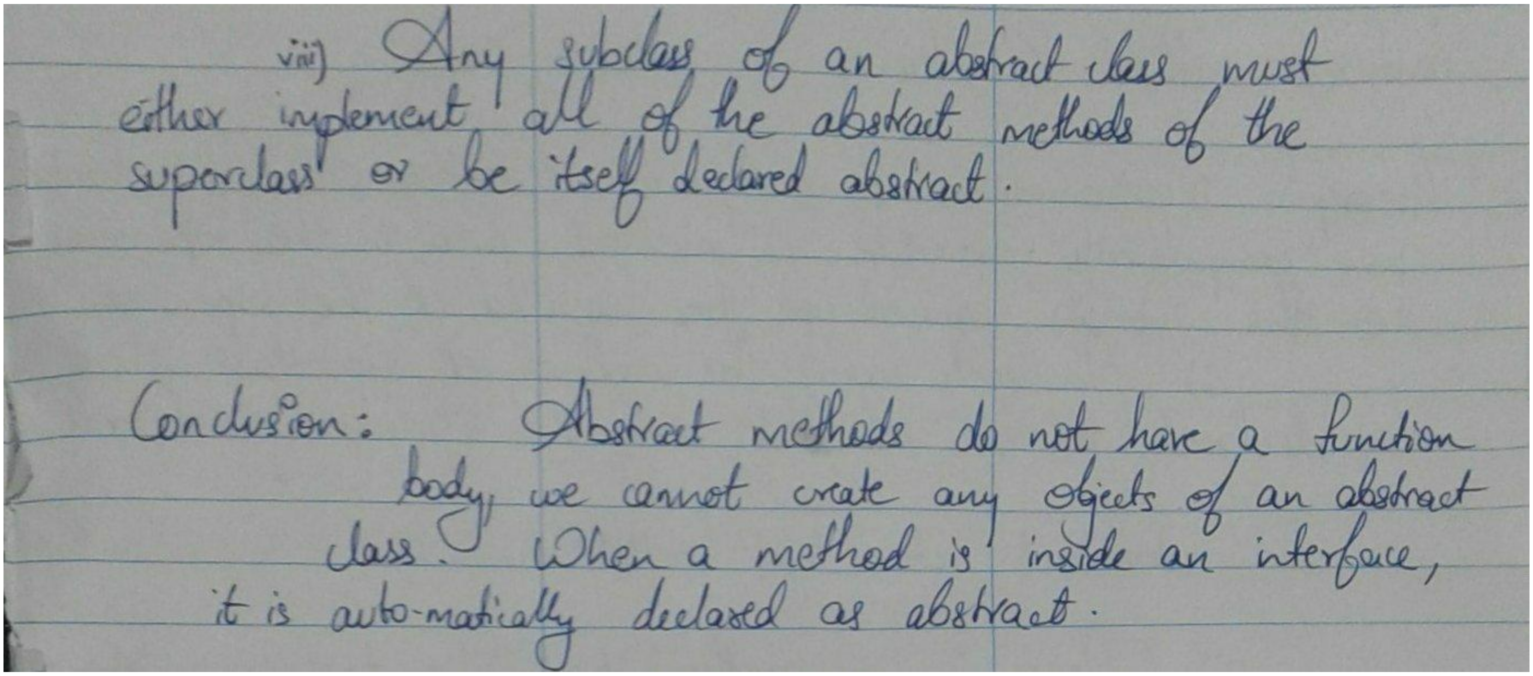
iv) A class which contains one or more abstract methods must be declared abstract. This is done using the keyword abstract in the front of keyword class at the beginning of class declaration

```
abstract class Shape {
        abstract void area ();
}
```

v) Here area () is a abstract method in a abstract class Shape.

vi) An abstract class cannot be initiated ie we cannot create objects using the new operator.

vii) A constructor or a static method cannot be declared abstract.



viii) Any subclass of an abstract class must either implement all of the abstract methods of the superclass or be itself declared abstract.

Conclusion: Abstract methods do not have a function body, we cannot create any objects of an abstract class. When a method is inside an interface, it is auto-matically declared as abstract.