

❖ به نام خدا

❖ برنامه سازی پیشرفته

❖ Abstract classes

مثال:

یک کلاس abstract برای اشکال هندسی تعریف کنید. این کلاس شامل رنگ اشکال، تعداد ابعاد اشکال و تاریخ ایجاد اشکال باشد، سپس دو کلاس مثلث و مستطیل از کلاس فوق مشتق شوند. کلاس مثلث دارای مشخصات سه ضلع مختلف و متدهای محاسبه محیط و مساحت باشد. کلاس مستطیل دارای طول و عرض و متدهای محاسبه محیط و مساحت باشد. این متدها به طور مناسب تعریف گردد. برنامه ای بنویسید که ۱۰ شیء از کلاس مثلث و ۱۰ شیء از کلاس مستطیل در یک آرایه مشترک خوانده شود. سپس یک متد تعریف کنید که نشان دهد مساحت مستطیل برابر با مساحت مثلث است؟ یا نه. این متد برای تمام اعضا از آرایه دو به دو مطابق شکل زیر که یکی از نوع مستطیل است و یکی از نوع مثلث است، فراخوانی گردد و پیغام مناسب چاپ شود.



```
public abstract class Geoobject2 {  
  
    private String color = "red";  
    private int dim;  
    private java.util.Date dateCreated;  
  
    protected Geoobject2(){  
        dateCreated = new java.util.Date();  
    }  
  
    protected Geoobject2(String color, int dim){  
        this.color = color;  
        this.dim = dim;  
        dateCreated = new java.util.Date();  
    }  
  
    public String toString(){  
        return "color: " + color + "\ndim: " + dim +  
            "\ndateCreated: " + dateCreated;  
    }  
  
    public abstract double area();  
    public abstract double perimeter();  
    public abstract void printObject();  
  
}
```

```

public class Rectangle extends Geoobject2{
    private double width = 1.0;
    private double height = 1.0;

    public Rectangle(double width, double height){
        this.width = width;
        this.height = height;
    }

    public Rectangle(double width, double height, String
                        color, int dim) {
        super(color, dim);
        this.width = width;
        this.height = height;
    }

    public double area() {
        return (width * height);
    }

    public double perimeter() {
        return 2 * (width + height);
    }

    public void printObject(){
        System.out.println(toString()+ "\nwidth: " + width +
                            "\nheight: " + height);
    }
}

```

```

public class Triangle extends Geoobject2{
    private double a = 1;
    private double b = 1;
    private double c = 1;

    public Triangle(double a, double b, double c) {
        this.a = a;
        this.b = b;
        this.c = c;
    }
    public Triangle(double a, double b, double c, String color,
                                                              int dim) {

        super(color, dim);
        this.a = a;
        this.b = b;
        this.c = c;
    }

    public double perimeter() {
        return a+b+c;
    }

    public double area() { // Based on Heron Rule
        double p = perimeter()/2;
        double ar=p*(p-a)*(p-b)*(p-c);
        return Math.sqrt(ar);
    }

    public void printObject(){
        System.out.println(toString()+ "\na: " + a + "\nb: " + b
                            + "\nc: " + c);
    }
}

```

```

import java.util.Scanner;
public class Geoobject {
    public static void main(String[] args) {
        Geoobject2[] object = new Geoobject2 [20];
        Scanner input = new Scanner (System.in);
        for (int i = 0; i < 10 ; i++){
            String color = input.next();
            int dim = input.nextInt();
            double width = input.nextDouble();
            double height = input.nextDouble();
            object[i] = new Rectangle(width, height, color, dim);
        }
        for (int i = 10; i < 20 ; i++){
            String color = input.next();
            int dim = input.nextInt();
            double a = input.nextDouble();
            double b = input.nextDouble();
            double c = input.nextDouble();
            object[i] = new Triangle(a, b, c, color, dim);
        }
        for (int i = 0; i < 20 ; i++){
            object[i].printObject();
        }
        for (int i = 0; i < 10 ; i++){
            if (equal (object[i], object[i+10]))
                System.out.println("The area of Rectangle
                                    ["+i+"] == The area of Triangle
                                    ["+(i+10)+"]");
        }
    }
    public static boolean equal(Geoobject2 ob1, Geoobject2 ob2) {
        return (ob1.area() == ob2.area());
    }
}

```