```
[13/04, 11:11 pm] Ferozi Amrita: class Shape {
    public double calculateArea(double side) {
        return side * side; // Square area
    }

    public double calculateArea(double length, double breadth) {
        return length * breadth; // Rectangle area
    }
}

class Circle extends Shape {
    @Override
    public double calculateArea(double radius) {
        return 3.14 * radius * radius; // Circle area
    }
}

public class findAreaShape {
    public static void main(String[] args) {
        Shape s = new Shape();
        Circle c = new Circle();

        System.out.println("Area of square is: " + s.calculateArea(5.0));
        System.out.println("Area of the rectangle is: " + s.calculateArea(3.5, 7.8));
        System.out.println("Area of the circle is: " + c.calculateArea(5.0));
        System.out.println("FEROZI BEGUM");

    }
}
[13/04, 11:12 pm] Ferozi Amrita: abstract class Shape3D{
    abstract double calculateVolume();
    abstract double calculateSurfaceArea();
}
class Sphere extends Shape3D{
    int radius;
    public Sphere(int radius){
        this.radius = radius;
    }
    @Override
    public double calculateVolume(){
        return (4.0/3.0) * Math.PI * Math.pow(radius, 3);
       }
    @Override
    public double calculateSurfaceArea(){
        return 4 * Math.PI * Math.pow(radius, 2);
    }
}
class cube extends Shape3D{
    int side;
    public cube(int side){
        this.side = side;
    }
    @Override
    public double calculateVolume(){
        return Math.pow(side,3);
    }
    @Override
    public double calculateSurfaceArea(){
        return 6 * Math.pow(side, 2);
    }
}
class Shapein3D{
    public static void main(String[] args) {
       cube c = new cube(5);
       System.out.println("Volume: " + c.calculateVolume());
       System.err.println("Surface Area="+ c.calculateSurfaceArea());
       Sphere s = new Sphere(5);
       System.out.println("Volume: " + s.calculateVolume());
       System.err.println("Surface Area="+ s.calculateSurfaceArea());
       System.out.println("FEROZI BEGAM");

    }
}
[13/04, 11:12 pm] Ferozi Amrita: interface Playable {
    void play();
}

class Football implements Playable {
    @Override
    public void play() {
        System.out.println("Running with the ball...");
        System.out.println("Scoring a goal!");
    }
}

class Volleyball implements Playable {
    @Override
    public void play() {
        System.out.println("Serving the ball...");
        System.out.println("Blocking the opponent's attack!");
    }
}

class Basketball implements Playable {
    @Override
    public void play() {
        System.out.println("Dribbling the ball...");
        System.out.println("Shooting a three-pointer...");
        System.out.println("Slam dunking!");
    }
}

public class OutdoorGame {
    public static void main(String[] args) {
        Football football = new Football();
        Volleyball volleyball = new Volleyball();
        Basketball basketball = new Basketball();

        System.out.println("\n--- Playing Sports ---\n");
        football.play();
        System.out.println("\n------------------------\n");
        volleyball.play();
        System.out.println("\n------------------------\n");
        basketball.play();
    }
}
[13/04, 11:13 pm] Ferozi Amrita: abstract class Animal{
    abstract void sound();
}
class lion extends Animal{
    @Override
    public void sound(){
        System.out.println("the lion is roaring!!");
    }
}
class tiger extends Animal{
    @Override
    public void sound(){
     System.out.println("the tiger is rearing!!");
    }
}
class landtSounds{
    public static void main(String[] args) {
        lion l = new lion();
        l.sound();
        tiger t =new tiger();
        t.sound();
        System.out.println("FEROZI BEGAM");
    }
}
[13/04, 11:13 pm] Ferozi Amrita: abstract class patternPrinting{
    public abstract void printPattern(int n);
    public void displayTitle(String title){
        System.out.println("The pattern name ="+title);
    }
}
class StarPattern extends patternPrinting{
    public void printPattern(int n) {
        for(int i=1;i<=n;i++){
        for(int j=1;j<=i;j++){
        System.out.print("*");
    } System.out.println();
}
    }}
class numberPattern extends patternPrinting{
 public void printPattern(int n){
for(int i=1;i<=n;i++){
for (int j=1;j<=i;j++){
System.out.print(j);
} System.out.println();
}
}}
public class patterns{
    public static void main(String[] args) {
     patternPrinting p = new StarPattern();
     p.displayTitle("Star Pattern");
     p.printPattern(5);
     numberPattern k=new numberPattern();
     k.displayTitle("Number Pattern");
     k.printPattern(5);
    }
```