

## Ejercicios interfaces y clases internas

1. Crea una interfaz llamada **Figura** que tenga métodos para calcular el área y el perímetro de una figura geométrica. Luego, implementa esta interfaz en clases concretas como **Círculo**, **Triángulo** y **Rectángulo**. Calcula el área y el perímetro de cada figura y muestra los resultados.

```
package interfaces;

public interface Figura {

    double calcularArea();
    double calcularPerimetro();

}

package interfaces;

public class Ejercicio1 {

    public static void main(String[] args) {

        Figura circulo = new Circulo(5);
        System.out.println("Área del círculo: " +
            circulo.calcularArea());
        System.out.println("Perímetro del círculo: " +
            circulo.calcularPerimetro());

        Figura triangulo = new Triangulo(4, 6);
        System.out.println("Área del triángulo: " +
            triangulo.calcularArea());
        System.out.println("Perímetro del triángulo: " +
            triangulo.calcularPerimetro());

        Figura rectangulo = new Rectangulo(3, 7);
        System.out.println("Área del rectángulo: " +
            rectangulo.calcularArea());
        System.out.println("Perímetro del rectángulo: " +
            rectangulo.calcularPerimetro());

    }

}

class Circulo implements Figura {
    private double radio;

    public Circulo(double radio) {
        this.radio = radio;
    }

    @Override
    public double calcularArea() {
        return Math.PI * radio * radio;
    }

    @Override
    public double calcularPerimetro() {
        return 2 * Math.PI * radio;
    }

}
```

```
}  
class Triangulo implements Figura {  
    private double base;  
    private double altura;  
  
    public Triangulo(double base, double altura) {  
        this.base = base;  
        this.altura = altura;  
    }  
  
    @Override  
    public double calcularArea() {  
        return (base * altura) / 2;  
    }  
  
    @Override  
    public double calcularPerimetro() {  
        // No se puede calcular el perimetro solo con base y altura  
        return Double.NaN;  
    }  
}  
  
class Rectangulo implements Figura {  
    private double ladoA;  
    private double ladoB;  
  
    public Rectangulo(double ladoA, double ladoB) {  
        this.ladoA = ladoA;  
        this.ladoB = ladoB;  
    }  
  
    @Override  
    public double calcularArea() {  
        return ladoA * ladoB;  
    }  
  
    @Override  
    public double calcularPerimetro() {  
        return 2 * (ladoA + ladoB);  
    }  
}
```

2. Crea una interfaz llamada **Motor** que tenga métodos para encender y apagar un motor. Luego, implementa esta interfaz en clases concretas como **MotorGasolina** y **MotorElectrico**. Crea una clase **Vehiculo** que tenga un atributo de tipo **Motor** y métodos para encender y apagar el motor.

```
package interfaces;  
  
public interface Motor {  
  
    void encender();  
    void apagar();  
  
}  
  
package interfaces;  
  
public class Ejercicio2 {
```

```
        public static void main(String[] args) {
            MotorGasolina motorGasolina = new MotorGasolina();
            Vehiculo vehiculoGasolina = new Vehiculo(motorGasolina);
            vehiculoGasolina.encenderMotor();
            vehiculoGasolina.apagarMotor();

            MotorElectrico motorElectrico = new MotorElectrico();
            Vehiculo vehiculoElectrico = new Vehiculo(motorElectrico);
            vehiculoElectrico.encenderMotor();
            vehiculoElectrico.apagarMotor();
        }
    }

    class MotorGasolina implements Motor {
        @Override
        public void encender() {
            System.out.println("Motor de gasolina encendido");
        }

        @Override
        public void apagar() {
            System.out.println("Motor de gasolina apagado");
        }
    }

    class MotorElectrico implements Motor {
        @Override
        public void encender() {
            System.out.println("Motor eléctrico encendido");
        }

        @Override
        public void apagar() {
            System.out.println("Motor eléctrico apagado");
        }
    }

    class Vehiculo {
        private Motor motor;

        public Vehiculo(Motor motor) {
            this.motor = motor;
        }

        public void encenderMotor() {
            motor.encender();
        }

        public void apagarMotor() {
            motor.apagar();
        }
    }
}
```

### 3. Crear una clase interna regular

```
package clas_inter;

public class Ejercicio3 {

    public static void main(String[] args) {
```

```
        ClaseExterna externa = new ClaseExterna();
        externa.usarClaseInterna();

    }

}

class ClaseExterna {
    private int datoExterno = 10;

    public class ClaseInterna {
        public void mostrarDatoExterno() {
            System.out.println("Dato externo: " + datoExterno);
        }
    }

    public void usarClaseInterna() {
        ClaseInterna interna = new ClaseInterna();
        interna.mostrarDatoExterno();
    }
}
```

#### 4. Crear una clase interna estática

```
package clas_inter;

public class Ejercicio4 {

    public static void main(String[] args) {
        ClaseExterna2.usarClaseInterna();
    }

}

class ClaseExterna2 {
    private static int datoExterno = 10;

    public static class ClaseInterna {
        public void mostrarDatoExterno() {
            System.out.println("Dato externo: " + datoExterno);
        }
    }

    public static void usarClaseInterna() {
        ClaseInterna interna = new ClaseInterna();
        interna.mostrarDatoExterno();
    }
}
```

#### 5. Crear una clase local

```
package clas_inter;

public class Ejercicio5 {

    public static void main(String[] args) {
        ClaseExterna3 externa = new ClaseExterna3();
        externa.usarClaseLocal();
    }

}
```

```
}  
class ClaseExterna3 {  
    private int datoExterno = 10;  
  
    public void usarClaseLocal() {  
        class ClaseLocal {  
            public void mostrarDatoExterno() {  
                System.out.println("Dato externo: " + datoExterno);  
            }  
        }  
  
        ClaseLocal local = new ClaseLocal();  
        local.mostrarDatoExterno();  
    }  
}
```