

Programación

TP 8 – Interfaces y Excepciones

Santiago Nicolas Nievas

Pagable.java

```
1 // Pagable.java
2 package com.utn.tp8;
3
4 public interface Pagable {
5     double calcularTotal();
6 }
```

Producto.java

```
1 // Producto.java
2 package com.utn.tp8;
3
4 public class Producto implements Pagable {
5     private final String nombre;
6     private final double precio;
7
8     public Producto(String nombre, double precio) {
9         this.nombre = nombre;
10        this.precio = precio;
11    }
12
13    public String getNombre() { return nombre; }
14    public double getPrecio() { return precio; }
15
16    @Override
17    public double calcularTotal() {
18        return precio;
19    }
20
21    @Override
22    public String toString() {
23        return nombre + " ($" + precio + ")";
24    }
25 }
```

Pago.java

```
1 // Pago.java
2 package com.utn.tp8;
3
4 public interface Pago {
5     boolean procesarPago(double monto);
6 }
```

PagoConDescuento.java

```
1 // PagoConDescuento.java
2 package com.utn.tp8;
3
4 public interface PagoConDescuento extends Pago {
5     double aplicarDescuento(double monto);
6 }
```

TarjetaDeCredito.java

```
1 // TarjetaCredito.java
2 package com.utn.tp8;
3
4 public class TarjetaCredito implements PagoConDescuento {
5     private final String titular;
6     private final String numero;
7     private final double porcentajeDescuento; // 0.10 = 10%
8
9     public TarjetaCredito(String titular, String numero, double porcentajeDescuento) {
10        this.titular = titular;
11        this.numero = numero;
12        this.porcentajeDescuento = porcentajeDescuento;
13    }
14
15    @Override
16    public double aplicarDescuento(double monto) {
17        return monto * (1.0 - porcentajeDescuento);
18    }
19
20    @Override
21    public boolean procesarPago(double monto) {
22        System.out.println("Pago con Tarjeta de " + titular +
23            " por $" + String.format("%.2f", monto) + " autorizado.");
24        return true;
25    }
26 }
```

PayPal.java

```
1 // PayPal.java
2 package com.utn.tp8;
3
4 public class PayPal implements PagoConDescuento {
5     private final String email;
6     private final double porcentajeDescuento; // 0.05 = 5%
7
8     public PayPal(String email, double porcentajeDescuento) {
9         this.email = email;
10        this.porcentajeDescuento = porcentajeDescuento;
11    }
12
13    @Override
14    public double aplicarDescuento(double monto) {
15        return monto * (1.0 - porcentajeDescuento);
16    }
17
18    @Override
19    public boolean procesarPago(double monto) {
20        System.out.println("Pago con PayPal (" + email + ") por $" +
21            String.format("%.2f", monto) + " completado.");
22        return true;
23    }
24 }
```

Notificable.java

```
1 // Notificable.java
2 package com.utn.tp8;
3
4 public interface Notificable {
5     void notificar(String mensaje);
6 }
```

Cliente.java

```
1 // Cliente.java
2 package com.utn.tp8;
3
4 public class Cliente implements Notificable {
5     private final String nombre;
6     private final String email;
7
8     public Cliente(String nombre, String email) {
9         this.nombre = nombre;
10        this.email = email;
11    }
12
13    public String getNombre() { return nombre; }
14    public String getEmail() { return email; }
15
16    @Override
17    public void notificar(String mensaje) {
18        System.out.println("Notificación a " + nombre + " (" + email + "): " + mensaje);
19    }
20 }
```

Pedido.java

```
1 // Pedido.java
2 package com.utn.tp8;
3
4 import java.util.ArrayList;
5 import java.util.List;
6
7 public class Pedido implements Pagable {
8     public enum Estado { CREADO, PAGADO, ENVIADO, ENTREGADO }
9
10    private final List<Producto> productos = new ArrayList<>();
11    private Estado estado = Estado.CREADO;
12    private final Notificable cliente;
13
14    public Pedido(Notificable cliente) {
15        this.cliente = cliente;
16        notificarEstado();
17    }
18
19    public void agregarProducto(Producto p) {
20        productos.add(p);
21    }
22
23    public List<Producto> getProductos() {
24        return new ArrayList<>(productos);
25    }
26
27    public Estado getEstado() {
28        return estado;
29    }
30
31    public void setEstado(Estado nuevo) {
32        this.estado = nuevo;
33        notificarEstado();
34    }
35
36    private void notificarEstado() {
37        if (cliente != null) {
38            cliente.notificar("Estado del pedido: " + estado);
39        }
40    }
41
42    @Override
43    public double calcularTotal() {
44        double total = 0.0;
45        for (Producto p : productos) {
46            total += p.calcularTotal();
47        }
48        return total;
49    }
50 }
```

MainEcommerce.java

```
1 // MainEcommerce.java
2 package com.utn.tp8;
3
4 public class MainEcommerce {
5     public static void main(String[] args) {
6         Cliente cli = new Cliente("Santiago", "santi@example.com");
7         Pedido pedido = new Pedido(cli);
8
9         pedido.agregarProducto(new Producto("Teclado", 25_000));
10        pedido.agregarProducto(new Producto("Mouse", 12_000));
11        pedido.agregarProducto(new Producto("Auriculares", 30_000));
12
13        double total = pedido.calcularTotal();
14        System.out.println("Total del pedido: $" + String.format("%.2f", total));
15
16        PagoConDescuento medio = new TarjetaCredito("Santiago",
17            "4111-1111-1111-1111", 0.10); // 10% off
18        double totalConDescuento = medio.aplicarDescuento(total);
19        System.out.println("Total con descuento: $" + String.format("%.2f", totalConDescuento));
20
21        if (medio.procesarPago(totalConDescuento)) {
22            pedido.setEstado(Pedido.Estado.PAGADO);
23            pedido.setEstado(Pedido.Estado.ENVIADO);
24            pedido.setEstado(Pedido.Estado.ENTREGADO);
25        }
26    }
27 }
```

DivisionSegura.java

```
1 // DivisionSegura.java
2 package com.utn.tp8;
3
4 import java.util.InputMismatchException;
5 import java.util.Scanner;
6
7 public class DivisionSegura {
8     public static void main(String[] args) {
9         try (Scanner sc = new Scanner(System.in)) {
10            System.out.print("Ingrese numerador: ");
11            int numerador = sc.nextInt();
12            System.out.print("Ingrese denominador: ");
13            int denominador = sc.nextInt();
14
15            int resultado = numerador / denominador;
16            System.out.println("Resultado: " + resultado);
17        } catch (ArithmeticException e) {
18            System.out.println("Error: No se puede dividir por cero.");
19        } catch (InputMismatchException e) {
20            System.out.println("Error: Debe ingresar enteros válidos.");
21        }
22    }
23 }
```

ConversionCadenaANumero.java

```
1 // ConversionCadenaANumero.java
2 package com.utn.tp8;
3
4 public class ConversionCadenaANumero {
5     public static void main(String[] args) {
6         String texto = "123x"; // ejemplo
7         try {
8             int valor = Integer.parseInt(texto); // puede lanzar NumberFormatException
9             System.out.println("Entero: " + valor);
10        } catch (NumberFormatException e) {
11            System.out.println("Error: '" + texto + "' no es un número entero válido.");
12        }
13    }
14 }
```

LecturaArchivo.java

```
1 // LecturaArchivo.java
2 package com.utn.tp8;
3
4 import java.io.BufferedReader;
5 import java.io.FileNotFoundException;
6 import java.io.FileReader;
7 import java.io.IOException;
8
9 public class LecturaArchivo {
10    public static void main(String[] args) {
11        String ruta = "archivo.txt"; // ajustar si lo ubicás en otro lado
12        try (BufferedReader br = new BufferedReader(new FileReader(ruta))) {
13            String linea;
14            while ((linea = br.readLine()) != null) {
15                System.out.println(linea);
16            }
17        } catch (FileNotFoundException e) {
18            System.out.println("Error: El archivo no existe (" + ruta + ").");
19        } catch (IOException e) {
20            System.out.println("Error de E/S: " + e.getMessage());
21        }
22    }
23 }
```

EdadInvalidaException.java

```
1 // EdadInvalidaException.java
2 package com.utm.tp8;
3
4 public class EdadInvalidaException extends Exception {
5     public EdadInvalidaException(String mensaje) {
6         super(mensaje);
7     }
8 }
```

ValidadorEdad.java

```
1 // ValidadorEdad.java
2 package com.utm.tp8;
3
4 public class ValidadorEdad {
5     public static void validar(int edad) throws EdadInvalidaException {
6         if (edad < 0 || edad > 120) {
7             throw new EdadInvalidaException("Edad inválida: " + edad);
8         }
9     }
10
11     public static void main(String[] args) {
12         try {
13             validar(30);
14             System.out.println("Edad válida.");
15             validar(130); // dispara la excepción personalizada
16         } catch (EdadInvalidaException e) {
17             System.out.println("Capturada excepción personalizada: " + e.getMessage());
18         }
19     }
20 }
```

TryWithResourcesLectura.java

```
1 // TryWithResourcesLectura.java
2 package com.utm.tp8;
3
4 import java.io.BufferedReader;
5 import java.io.FileReader;
6 import java.io.IOException;
7
8 public class TryWithResourcesLectura {
9     public static void main(String[] args) {
10         String ruta = "archivo.txt";
11         try (BufferedReader br = new BufferedReader(new FileReader(ruta))) {
12             String linea;
13             while ((linea = br.readLine()) != null) {
14                 System.out.println(linea);
15             }
16         } catch (IOException e) {
17             System.out.println("Ocurrió un error al leer el archivo: " + e.getMessage());
18         }
19     }
20 }
```