

TRABAJO PRÁCTICO 7

Herencia y Polimorfismo

Alumno: Ignacio Figueroa – 45.406.120

Tecnicatura Universitaria en Programación – UTN

Materia: Programación II

Comisión: 7

Objetivo:

Comprender y aplicar los conceptos de herencia y polimorfismo en la Programación Orientada a Objetos, reconociendo su importancia para la reutilización de código, la creación de jerarquías de clases y el diseño flexible de soluciones en Java.

Link al repositorio con el código completo: [tup-utn/2do-cuatrimestre/programacion-II/herencia-y-polimorfismo/tp/tp-7-herencia-y-polimorfismo](https://github.com/tup-utn/2do-cuatrimestre/programacion-II/herencia-y-polimorfismo/tp/tp-7-herencia-y-polimorfismo) at main · [figueroaignacio/tup-utn](https://github.com/figueroaignacio/tup-utn)

Ejercicio 1 – Vehículos

Vehiculo.java

```

1  package Ejercicio1;
2
3  public class Vehiculo {
4
5      private String marca;
6      private String modelo;
7
8      public Vehiculo(String marca, String modelo) {
9          this.marca = marca;
10         this.modelo = modelo;
11     }
12
13     public String getMarca() {
14         return marca;
15     }
16
17     public String getModelo() {
18         return modelo;
19     }
20
21     public void mostrarInfo() {
22         System.out.println("Vehículo - Marca: " + marca + ", Modelo: " + modelo);
23     }
24 }
25

```

Auto.java

```

1  package Ejercicio1;
2
3  public class Auto extends Vehiculo {
4
5      private int cantidadPuertas;
6
7      public Auto(String marca, String modelo, int cantidadPuertas) {
8          super(marca, modelo);
9          this.cantidadPuertas = cantidadPuertas;
10     }
11
12     @Override
13     public void mostrarInfo() {
14         System.out.println("Auto - Marca: " + getMarca() + ", Modelo: " + getModelo()
15             + ", Puertas: " + cantidadPuertas);
16     }
17 }
18

```

DemoEjercicio1.java

```

1  package Ejercicio1;
2
3
4  public class DemoEjercicio1 {
5      public static void run() {
6          System.out.println("=== Ejercicio 1: Vehículos ===");
7
8          Auto auto = new Auto("Toyota", "Corolla", 4);
9          auto.mostrarInfo();
10
11         System.out.println();
12     }
13 }
14

```

Resultado en consola

Output - tp-7-herencia-y-polimorfismo (run)

```

run:
=== Ejercicio 1: Vehículos ===
Auto - Marca: Toyota, Modelo: Corolla, Puertas: 4

BUILD SUCCESSFUL (total time: 0 seconds)

```

Ejercicio 2 – Figuras

Figura.java

```

1  package Ejercicio2;
2
3  public abstract class Figura {
4
5      protected String nombre;
6
7      public Figura(String nombre) {
8          this.nombre = nombre;
9      }
10
11     public abstract double calcularArea();
12
13     public void mostrarArea() {
14         System.out.printf("%s → Área: %.2f%n", nombre, calcularArea());
15     }
16
17 }
18

```

Circulo.java

```

1  package Ejercicio2;
2
3  public class Circulo extends Figura {
4
5      private double radio;
6
7      public Circulo(double radio) {
8          super("Circulo");
9          this.radio = radio;
10     }
11
12     @Override
13     public double calcularArea() {
14         return Math.PI * radio * radio;
15     }
16
17 }
18

```

Rectangulo.java

```

1  package Ejercicio2;
2
3  public class Rectangulo extends Figura {
4
5      private double ancho;
6      private double alto;
7
8      public Rectangulo(double ancho, double alto) {
9          super("Rectángulo");
10         this.ancho = ancho;
11         this.alto = alto;
12     }
13
14     @Override
15     public double calcularArea() {
16         return ancho * alto;
17     }
18 }
19

```

DemoEjercicio2.java

```

1  package Ejercicio2;
2
3  public class DemoEjercicio2 {
4
5      public static void run() {
6          System.out.println("≡≡≡ Ejercicio 2: Figuras ≡≡≡");
7
8          Figura[] figuras = {
9              new Circulo(3),
10             new Rectangulo(4, 6)
11         };
12
13         for (Figura f : figuras) {
14             f.mostrarArea();
15         }
16
17         System.out.println();
18     }
19 }
20

```

Resultado en consola

```
Output - tp-7-herencia-y-polimorfismo (run)

run:
=== Ejercicio 2: Figuras ===
Circulo ? area: 28.27
Rectangulo ? area: 24.00

BUILD SUCCESSFUL (total time: 0 seconds)
```

Ejercicio 3 – Empleados

Empleado.java

```
1 package Ejercicio3;
2
3 public abstract class Empleado {
4
5     protected String nombre;
6
7     public Empleado(String nombre) {
8         this.nombre = nombre;
9     }
10
11     public abstract double calcularSueldo();
12
13     public void mostrarSueldo() {
14         System.out.printf("%s -> Sueldo: %.2f%n", nombre, calcularSueldo());
15     }
16
17 }
```

EmpleadoPlanta.java

```

1  package Ejercicio3;
2
3  public class EmpleadoPlanta extends Empleado {
4
5      private final double sueldoFijo;
6
7      public EmpleadoPlanta(String nombre, double sueldoFijo) {
8          super(nombre);
9          this.sueldoFijo = sueldoFijo;
10     }
11
12     @Override
13     public double calcularSueldo() {
14         return sueldoFijo;
15     }
16
17 }
18

```

EmpleadoTemporal.java

```

1  package Ejercicio3;
2
3  public class EmpleadoTemporal extends Empleado {
4
5      private final int horasTrabajadas;
6      private final double tarifa;
7
8      public EmpleadoTemporal(String nombre, int horas, double tarifa) {
9          super(nombre);
10         this.horasTrabajadas = horas;
11         this.tarifa = tarifa;
12     }
13
14     @Override
15     public double calcularSueldo() {
16         return horasTrabajadas * tarifa;
17     }
18
19 }
20

```

DemoEjercicio3.java

```

1  package Ejercicio3;
2
3  public class DemoEjercicio3 {
4
5      public static void run() {
6          System.out.println("=== Ejercicio 3: Empleados ===");
7
8          Empleado[] empleados = {
9              new EmpleadoPlanta("Ana", 80000),
10             new EmpleadoTemporal("Luis", 120, 1500)
11         };
12
13         for (Empleado e : empleados) {
14             e.mostrarSueldo();
15         }
16
17         System.out.println();
18     }
19
20 }
21

```

Resultado en consola

```

Output - tp-7-herencia-y-polimorfismo (run)

run:
=== Ejercicio 3: Empleados ===
Ana ? Sueldo: 80000.00
Luis ? Sueldo: 180000.00

BUILD SUCCESSFUL (total time: 0 seconds)

```

Ejercicio 4 – Animales

Animal.java

```

1  package Ejercicio4;
2
3  public class Animal {
4
5      protected String nombre;
6
7      public Animal(String nombre) {
8          this.nombre = nombre;
9      }
10
11     public void hacerSonido() {
12         System.out.println(nombre + " hace un sonido...");
13     }
14
15     public void describirAnimal() {
16         System.out.println("Este es " + nombre);
17     }
18
19 }
20

```

Perro.java

```

1  package Ejercicio4;
2
3  public class Perro extends Animal {
4
5      public Perro(String nombre) {
6          super(nombre);
7      }
8
9      @Override
10     public void hacerSonido() {
11         System.out.println(nombre + " dice: Guau");
12     }
13
14 }

```

Gato.java

```

1  package Ejercicio4;
2
3  public class Gato extends Animal {
4
5      public Gato(String nombre) {
6          super(nombre);
7      }
8
9      @Override
10     public void hacerSonido() {
11         System.out.println(nombre + " dice: Miau");
12     }
13 }

```

Vaca.java

```

1  package Ejercicio4;
2
3  public class Vaca extends Animal {
4
5      public Vaca(String nombre) {
6          super(nombre);
7      }
8
9      @Override
10     public void hacerSonido() {
11         System.out.println(nombre + " dice: Muu");
12     }
13 }
14

```

DemoEjercicio4.java

```

1  package Ejercicio4;
2
3  public class DemoEjercicio4 {
4
5      public static void run() {
6          System.out.println("=== Ejercicio 4: Animales ===");
7
8          Animal[] animales = {
9              new Perro("Rex"),
10             new Gato("Michi"),
11             new Vaca("Lola")
12         };
13
14         for (Animal a : animales) {
15             a.describirAnimal();
16             a.hacerSonido();
17             System.out.println();
18         }
19     }
20 }

```

Resultado en consola

```
Output - tp-7-herencia-y-polimorfismo (run)

run:
=== Ejercicio 4: Animales ===
Este es Rex
Rex dice: Guau

Este es Michi
Michi dice: Miau

Este es Lola
Lola dice: Muu

BUILD SUCCESSFUL (total time: 0 seconds)
```

Tp7HerenciaYPolimorfismo.java

```
1 package tp.pkg7.herencia.y.polimorfismo;
2
3 import Ejercicio1.DemoEjercicio1;
4 import Ejercicio2.DemoEjercicio2;
5 import Ejercicio3.DemoEjercicio3;
6 import Ejercicio4.DemoEjercicio4;
7
8 /**
9  *
10  * @author Ignacio Figueroa - TP 7 | Herencia y Polimorfismo en Java
11  */
12 public class Tp7HerenciaYPolimorfismo {
13
14     public static void main(String[] args) {
15         DemoEjercicio1.run();
16         DemoEjercicio2.run();
17         DemoEjercicio3.run();
18         DemoEjercicio4.run();
19     }
20
21 }
22
```