

# UNIDAD 2

## PRÁCTICA 5

<b>ASIGNATURA</b>	Programación Orientada a Objetos
<b>CICLO</b>	Segundo
<b>PARALELO</b>	A
<b>PERIODO ACADÉMICO</b>	abril - septiembre 2023
<b>INTEGRANTES</b>	<ul style="list-style-type: none"><li>• Miguel Ángel Luna Yunga</li><li>• Fernando Sebastián Vásquez Reyes</li></ul>

### 1. TEMA:

Implementación de un programa en un lenguaje de programación orientada a objetos.

### 2. OBJETIVOS:

- Familiarizarse con un lenguaje de programación orientada a objetos (como Java o Python).
- Comprender la sintaxis y estructuras fundamentales del lenguaje.
- Implementar un programa sencillo utilizando el lenguaje.

### 3. RECURSOS Y MATERIALES

- Computadoras con el lenguaje de programación instalado (por ejemplo, JDK para Java o Python).
- Entorno de desarrollo integrado (IDE) para el lenguaje (como Eclipse para Java o PyCharm para Python).
- Ejercicios de práctica y problemas.

### 4. PROCEDIMIENTO:

- a. Introducción: El docente introduce el lenguaje de programación orientada a objetos y sus características, explicando su sintaxis y estructuras fundamentales.
- b. Implementación de un programa sencillo: Los estudiantes implementan un programa sencillo utilizando el lenguaje de programación, como una aplicación de consola que realice operaciones aritméticas o una aplicación que muestre mensajes en pantalla.
- c. Análisis y depuración del programa: Los estudiantes analizan y depuran el programa para identificar y solucionar errores y problemas en el código.

- d. Ejercicios de práctica: Los estudiantes resuelven ejercicios y problemas relacionados con el lenguaje de programación aprendido, con el apoyo y retroalimentación del docente.

### Ejercicios:

- Implementar un programa en Java que calcule la suma de dos números ingresados por el usuario.
- Implementar un programa en Java que calcule el área de un triángulo utilizando la fórmula de Herón.
- Implementar un programa en Java que permita al usuario ingresar un número entero y muestre en pantalla si es un número par o impar.
- Implementar un programa en Java que genere una lista de números aleatorios y muestre en pantalla los números pares de la lista.
- Implementar un programa en el lenguaje de programación orientada a objetos de elección del estudiante que realice una función específica, como ordenar una lista o calcular el promedio de una serie de números.

## 5. RESULTADOS OBTENIDOS

### - Ejercicio 1.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project View:** Shows a project named "SumaNumeros" with a "src" directory containing a "Main.java" file.
- Main.java Content:**

```
//suma de dos numeros
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int num1, num2, suma;
        System.out.println("Ingrese el primer numero: ");
        num1 = scanner.nextInt();
        System.out.println("Ingrese el segundo numero: ");
        num2 = scanner.nextInt();
        suma = num1 + num2;
        System.out.println("La suma es: " + suma);
    }
}
```
- Run Tab Output:**

```
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.1.2\lib\idea_rt.jar=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.1.2\bin\idea.jar" Main
Ingresé el primer numero:
6
Ingresé el segundo numero:
8
La suma es: 14
Process finished with exit code 0
```

## - Ejercicio 2.

```

import java.util.Scanner;

new
public class Main {
    new
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Ingrese la longitud del primer lado del triángulo:");
        double lado1 = scanner.nextDouble();

        System.out.println("Ingrese la longitud del segundo lado del triángulo:");
        double lado2 = scanner.nextDouble();

        System.out.println("Ingrese la longitud del tercer lado del triángulo:");
        double lado3 = scanner.nextDouble();

        Triangulo triangulo = new Triangulo(lado1, lado2, lado3);
        double area = triangulo.calcularArea();

        String areaFormatada = String.format("%.2f", area);

        System.out.println("El área del triángulo es: " + areaFormatada);
    }
}

```

```

3 usages
private double lado1;
3 usages
private double lado2;
3 usages
private double lado3;

1 usage new
public Triangulo(double lado1, double lado2, double lado3) {
    this.lado1 = lado1;
    this.lado2 = lado2;
    this.lado3 = lado3;
}

1 usage new
public double calcularArea() {
    double semiperimetro = (lado1 + lado2 + lado3) / 2;
    double area = Math.sqrt(semiperimetro * (semiperimetro - lado1) * (semiperimetro - lado2) * (semiperimetro - lado3));
    return area;
}

```

```

Run: Main
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.1.2\lib\idea_rt.jar=C:\Program Files\Java\jdk-15.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.1.2\lib\idea_rt.jar=C:\Program Files\Java\jdk-15.0.2\bin\java.exe"
Ingresé la longitud del primer lado del triángulo:
4
Ingresé la longitud del segundo lado del triángulo:
6
Ingresé la longitud del tercer lado del triángulo:
5
El área del triángulo es: 6,93

Process finished with exit code 0

```



- Ejercicio 3.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project:** Numeropar
- File:** Main.java
- Code Content:**

```
new*
public class Main{
    new*
    public static void main(String[] args) {
        try (Scanner scanner = new Scanner(System.in)) {
            System.out.print("Ingrese un número entero: ");
            int numero = scanner.nextInt();

            if (esPar(numero)) {
                System.out.println("El número ingresado es par.");
            } else {
                System.out.println("El número ingresado es impar.");
            }
        }
    }

    usage new*
    public static boolean esPar(int numero) {
        return numero % 2 == 0;
    }
}
```

- Run Tab:** Shows the output of running the program:

```
Ingrese un número entero: 4
El número ingresado es par.

Process finished with exit code 0
```

- Ejercicio 4.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project:** Listanumeros
- File:** Main.java
- Code Content:**

```
import java.util.ArrayList;
import java.util.List;

new*
public class Main {
    new*
    public static void main(String[] args) {
        int cantidadNumeros = 10; // Cantidad de números a generar

        GenerarNumeros generador = new GenerarNumeros();
        List<Integer> numeros = generador.generarNumerosAleatorios(cantidadNumeros);

        System.out.println("Números generados: " + numeros);

        VerificadorPar verificador = new VerificadorPar();
        System.out.println("Números pares:");
        for (int numero : numeros) {
            if (verificador.esPar(numero)) {
                System.out.println(numero);
            }
        }
    }
}
```

- Run Tab:** Shows the output of running the program:

```
Números generados: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Números pares:
2
4
6
8
10

Process finished with exit code 0
```



UNL

Universidad  
Nacional  
de Loja

FACULTAD DE LA ENERGÍA, LAS INDUSTRIAS Y  
LOS RECURSOS NATURALES NO RENOVABLES  
CARRERA DE INGENIERÍA EN COMPUTACIÓN

The screenshot shows two Java files in an IntelliJ IDEA interface:

**GenerarNumeros.java:**

```
import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class GenerarNumeros {
    public List<Integer> generarNumerosAleatorios(int cantidad) {
        List<Integer> numeros = new ArrayList<>();
        Random random = new Random();

        for (int i = 0; i < cantidad; i++) {
            int numero = random.nextInt( bound: 100 ); // Genera un número aleatorio entre 0 y 99
            numeros.add(numero);
        }

        return numeros;
    }
}
```

**VerificadorPar.java:**

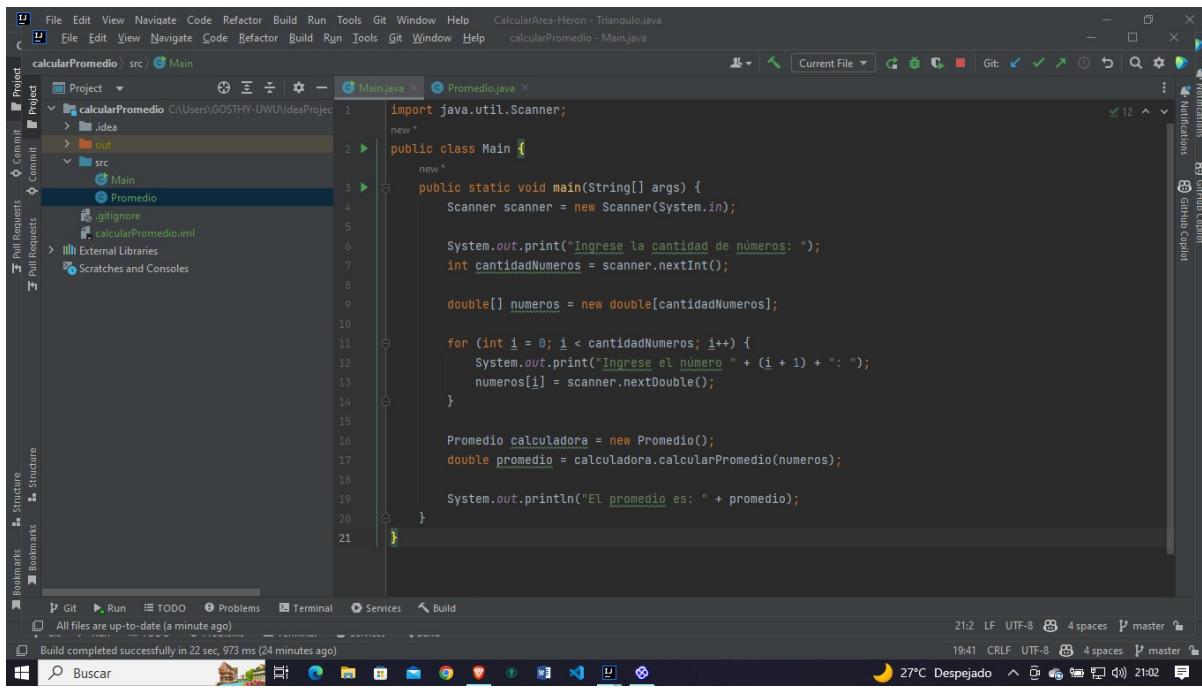
```
public class VerificadorPar {
    public boolean esPar(int numero) {
        return numero % 2 == 0;
    }
}
```

**Run Output:**

```
Números generados: [85, 77, 24, 88, 87, 62, 11, 99, 4, 14]
Números pares:
24
88
62
4
14

Process finished with exit code 0
```

### - Ejercicio 5.



```

File Edit View Navigate Code Refactor Build Run Tools Git Window Help CalcularArea-Heron - Triangulo.java
File Edit View Navigate Code Refactor Build Run Tools Git Window Help calcularPromedio - Main.java
calculatorPromedio src Main
Project Commit Pull Requests GitHub Copilot Notifications
calculatorPromedio C:\Users\GOSTHY-UWU\IdeaProject .idea out src Main Promedio .gitignore calcularPromedio.iml External Libraries Scratches and Consoles
Main.java x Promedio.java x
1 import java.util.Scanner;
new *
2 public class Main {
new *
3     public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
4
System.out.print("Ingrese la cantidad de números: ");
int cantidadNumeros = scanner.nextInt();
5
6
double[] numeros = new double[cantidadNumeros];
7
8
for (int i = 0; i < cantidadNumeros; i++) {
System.out.print("Ingrese el número " + (i + 1) + ": ");
numeros[i] = scanner.nextDouble();
9
}
10
11
Promedio calculadora = new Promedio();
double promedio = calculadora.calcularPromedio(numeros);
12
13
System.out.println("El promedio es: " + promedio);
14
}
15
16
17
18
19
20
21

```

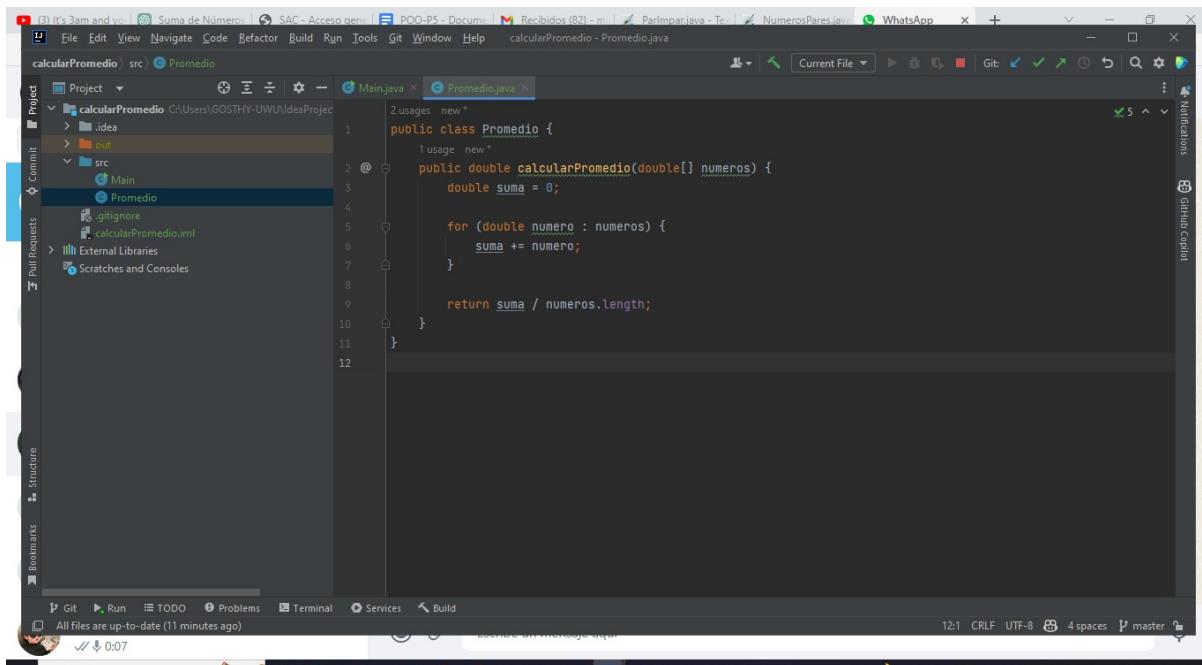
Git Run TODO Problems Terminal Services Build

All files are up-to-date (a minute ago)

Build completed successfully in 22 sec, 973 ms (24 minutes ago)

Buscar

21:2 LF UTF-8 8 spaces master



```

File Edit View Navigate Code Refactor Build Run Tools Git Window Help calcularPromedio - Promedio.java
calculatorPromedio src Promedio
Project Commit Pull Requests GitHub Copilot Notifications
calculatorPromedio C:\Users\GOSTHY-UWU\IdeaProject .idea out src Main Promedio .gitignore calcularPromedio.iml External Libraries Scratches and Consoles
Main.java x Promedio.java x
1 2 usages new "
public class Promedio {
1 usage new "
2     public double calcularPromedio(double[] numeros) {
double suma = 0;
3
4         for (double numero : numeros) {
suma += numero;
5
6
7             return suma / numeros.length;
8
9
}
10
11
}
12

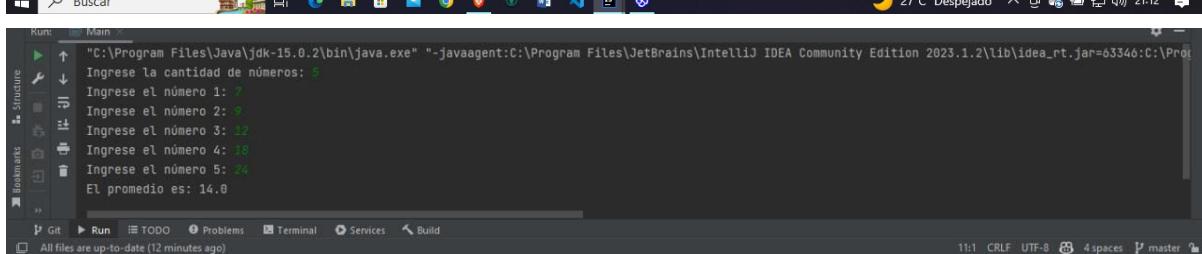
```

Git Run TODO Problems Terminal Services Build

All files are up-to-date (11 minutes ago)

Buscar

12:1 CRLF UTF-8 8 spaces master



```

Kun: Main
" C:\Program Files\Java\jdk-15.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.1.2\lib\idea_rt.jar=63346:C:\Program
Ingrese la cantidad de números: 5
Ingrese el número 1: 7
Ingrese el número 2: 9
Ingrese el número 3: 12
Ingrese el número 4: 18
Ingrese el número 5: 24
El promedio es: 14.0

```

Git Run TODO Problems Terminal Services Build

All files are up-to-date (12 minutes ago)

11:1 CRLF UTF-8 8 spaces master

## 6. CONCLUSIONES

- Al aprender un lenguaje orientado a objetos, hemos adquirido los conceptos clave de la programación orientada a objetos, como clases, objetos, herencia y polimorfismo. Esto permite organizar y estructurar el código de manera más eficiente y modular.
- Así mismo al comprender la sintaxis, se adquiere conocimiento sobre cómo se deben formar las instrucciones, cómo declarar variables, cómo utilizar estructuras de control (como bucles y condicionales) y cómo definir y llamar funciones o métodos. Esto proporciona una base sólida para desarrollar programas complejos y comprensión de cómo el lenguaje opera.
- Al aplicar los conocimientos adquiridos sobre la sintaxis y las estructuras del lenguaje, se puede construir un programa funcional que realice una tarea específica. Al hacerlo, se pone en práctica la resolución de problemas, la lógica de programación y la capacidad de traducir las ideas en código ejecutable. Implementar un programa también ayuda a comprender cómo se aplican los conceptos de la programación orientada a objetos en la práctica.
- En conclusión, hemos entendido de mejor manera la programación orientada a objetos y así mismo hemos comprendido los diferentes conceptos que son puntos clave al momento de desarrollar los códigos de los diferentes programas.

## 7. REFERENCIAS

Para diferentes sugerencias, contamos con la ayuda de Chat Gpt, una herramienta que nos brindó una amplia variedad de ideas, que fue de gran ayuda para mejorar el código de cada uno de los programas. Además, recibimos una explicación detallada sobre el código, lo cual nos permitió entender mejor cómo funciona el programa y cada una de sus clases.

Aparte de esto, nos ayudamos en algunos vídeos de Youtube para obtener los mejores resultados en nuestros códigos.

## 8. RÚBRICA

<b>Informe:</b>	3 pts
<b>Resultados:</b>	4 pts
<b>Conclusiones:</b>	3 pts
<b>Total</b>	10 pts