# Introducción a Java

## 1. Introducción a Java: Bibliotecas y Fundamentos

Java es un lenguaje de programación orientado a objetos, creado para ser simple, seguro y fácil de aprender. Es popular para aplicaciones empresariales, desarrollo móvil (especialmente Android), y sistemas distribuidos. Java cuenta con una amplia biblioteca estándar (Java API) que nos permite usar funciones ya creadas, como la entrada/salida de datos, manipulación de colecciones y operaciones matemáticas. Este enfoque ahorra tiempo y mejora la eficiencia en la programación, ya que reutilizamos código probado.

## 2. Presentación de la Sintaxis de Java

La sintaxis de Java es muy similar a la de otros lenguajes como C y C++, por lo que se considera fácil de aprender si tienes nociones previas de programación. Aquí tienes algunos elementos básicos:  
- Clases y Objetos: Todo en Java gira en torno a clases y objetos. Para crear un programa en Java, se debe definir al menos una clase.  
- Métodos: Son bloques de código que realizan tareas específicas y se definen dentro de las clases.  
- Main: Es el método principal que se ejecuta al correr un programa. En Java se define así: `public static void main(String[] args) { }`.  
- Palabras clave como `public`, `static`, y `void` tienen funciones específicas.

## 3. Método de Iniciar Java

Para empezar a programar en Java, necesitas:  
- Instalar el JDK (Java Development Kit), que incluye el compilador y herramientas para programar en Java.  
- Un IDE (como IntelliJ IDEA, Eclipse o NetBeans) o simplemente un editor de texto como VS Code para escribir el código.  
- El código Java se guarda en archivos `.java` y, después de compilarlo, se convierte en archivos `.class` que pueden ejecutarse en la JVM (Java Virtual Machine).

## 4. Herencia en Java

La herencia es un concepto fundamental de la programación orientada a objetos y permite que una clase (subclase) herede atributos y métodos de otra clase (superclase). Esto nos ayuda a reutilizar código y a estructurarlo de manera más lógica y jerárquica.

Ejemplo de Herencia:

A screen shot of a computer program

Description automatically generated

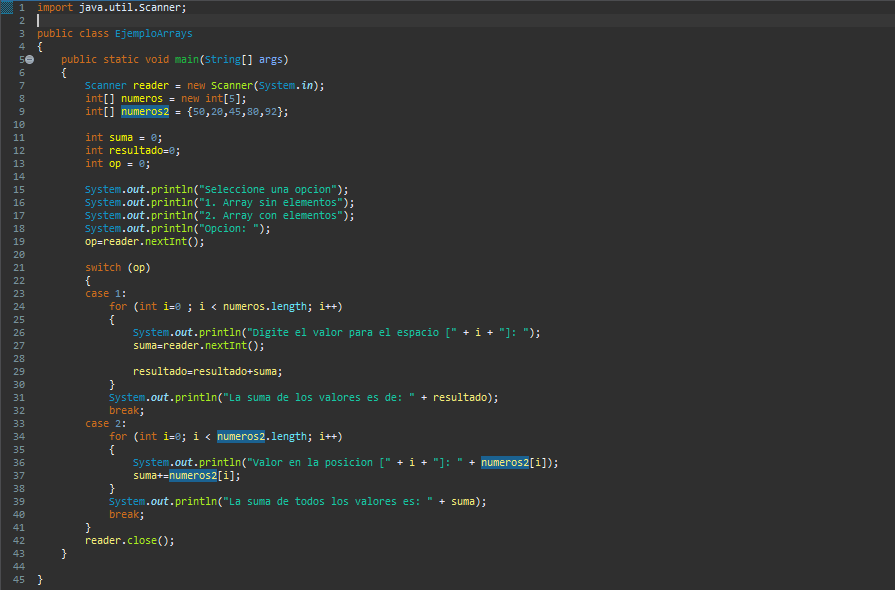
## 5. Declaración y Creación de un Array

Un array en Java es una estructura de datos que permite almacenar múltiples valores del mismo tipo en una sola variable. Se declara y se inicializa asignando un tamaño fijo o valores específicos.

## 6. Ejemplo del Array

Un ejemplo común es almacenar calificaciones en un array y calcular el promedio. Utilizamos un ciclo `for` para recorrer el array y sumar las calificaciones, luego calculamos el promedio dividiendo la suma entre el número de elementos.

Ejemplo de Array:



## 7. Errores Comunes en los Arrays

Algunos errores frecuentes incluyen:  
- Índice fuera de límites: Intentar acceder a una posición fuera del tamaño del array genera un error (`ArrayIndexOutOfBoundsException`).  
- Arrays sin inicializar: Declarar un array sin asignarle memoria (`null`) y luego intentar acceder a él genera una excepción.