

**Programación Orientada a Objetos**

**Leonardo Cerda Rodriguez**

**AL07024282**

**Profesor Héctor Antonio Aguilar**

**Actividad 2**



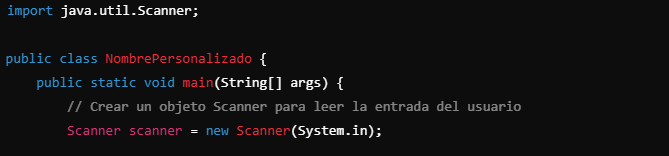
**Introducción**

En esta actividad, no adentraremos en diversos temas clave de la programación en Java. Por ejemplo, aprenderemos sobre el manejo de cadenas de texto (Strings), estructuras de selección y repetición, así como el uso de arreglos. Estos conceptos nos proporcionarán las bases necesarias para crear aplicaciones más complejas y flexibles. Podremos aplicar los conocimientos adquiridos sobre Strings, estructuras de control y arreglos en Java, además de diseñar y desarrollar programas que trabajen con cadenas de texto, tomen decisiones, realicen ciclos y utilicen arreglos para resolver problemas prácticos.

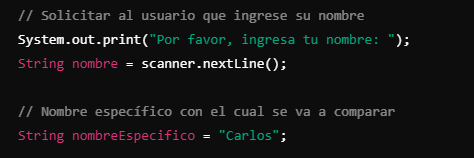
**Desarrollo**

Primeramente, abriremos un proyecto nuevo de java con ayuda de NetBeans y de los paquetes de instalación JDK de Oracle. Procedemos a hacer el primer programa de strings y estructuras de selección.

Importamos el paquete java.util.scanner para utilizar el escáner y así los usuarios puedan introducir datos por medio de la consola. Declaramos la clase Saludo la cual debe ser pública.

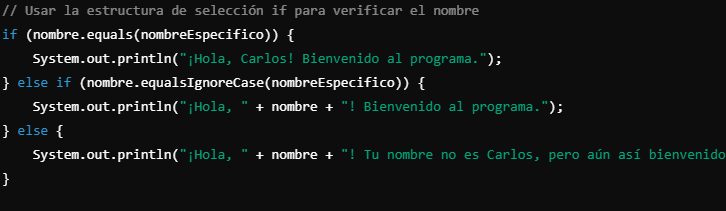


Ahora continuamos el código con un public static void main (String[] args lo cual nos ayuda a dar acceso desde cualquier parte del código, static nos menciona que dicho método es perteneciente a la clase y no requiere de una instancia para ejecutarse. Void se refiere a que no nos devuelve un valor, main es nuestro método principal para java, String[] args recibe argumentos provenientes de la línea de comandos.

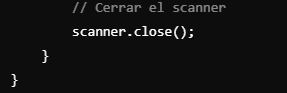


El método equals() compara dos cadenas de texto para determinar si son exactamente iguales, teniendo en cuenta las mayúsculas y minúsculas. Esto significa que diferencia entre "Carlos" y "carlos", porque uno está en mayúsculas y el otro no.

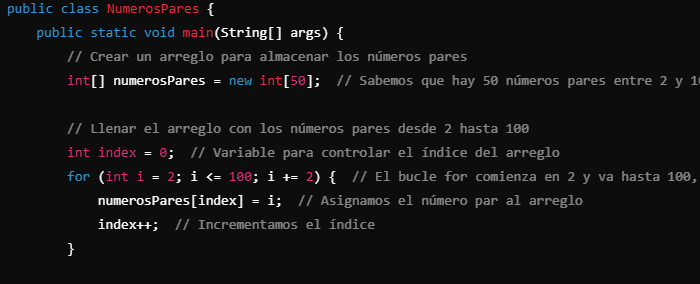
La estructura if es una estructura de control que permite ejecutar diferentes bloques de código según ciertas condiciones. La primera condición compara el nombre ingresado por el usuario (nombre) con el nombre específico (nombreEspecifico, que es "Carlos") usando el método equals(). La segunda condición else if hace que si la primera condición no se cumple, se compara el nombre con equalsIgnoreCase(), que no distingue entre mayúsculas y minúsculas. Si el nombre coincide con "Carlos" sin importar las letras en mayúscula o minúscula, se imprime un mensaje, pero con el nombre tal como lo ingresó el usuario.



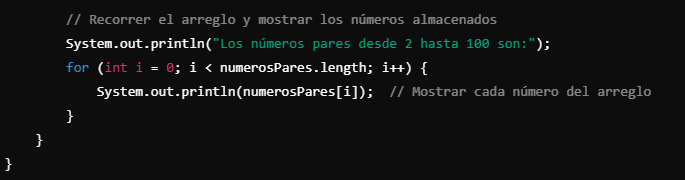
Por ultimo cerramos el scanner y cerramos los corchetes correspondientemente al espaciado de cada función del programa para que no tengamos errores de ejecución ni de sintaxis.



Para el segundo programa con referencia al tema de Estructuras de repetición y arreglos abriremos un nuevo proyecto, creamos un arreglo numerosPares de tipo int con capacidad para almacenar 50 elementos, ya que hay 50 números pares entre 2 y 100 y usamos un bucle for que comienza en 2 (i = 2) y va hasta 100, con un incremento de 2 en cada iteración (i += 2), para asegurar que solo se seleccionen los números pares.



Usamos otro bucle for para recorrer el arreglo numerosPares y mostrar cada número almacenado en él. El índice i va desde 0 hasta numerosPares.length - 1, que es la cantidad total de elementos en el arreglo y por último cerramos los corchetes y probamos el correcto funcionamiento del programa.

.