PRACTICA DE STRING

1. **Creación de la Clase Alumno:**
   * Definimos una clase llamada **Alumno** para representar a cada estudiante.
   * Cada alumno tendrá tres atributos: **nombre** (String), **apellido** (String) y **dni** (String).
   * Usaremos el constructor de la clase para inicializar estos atributos cuando creemos un nuevo objeto **Alumno**.
   * Usaremos los **metodos get** para poder obtener los datos de los alumnos.
2. **Uso de la Clase String:**
   * La clase **String** se utiliza para almacenar y manipular cadenas de caracteres (texto).
   * En este programa, la utilizamos para almacenar el nombre, apellido y DNI de cada alumno.
   * Usaremos algunos métodos de la clase String para mostrar datos de los alumnos.
3. **Creación de la Clase Main:**
   * La clase **Main** contiene el método **main**, que es el punto de entrada del programa.
   * Creamos algunos objetos **Alumno** para representar a diferentes estudiantes.
   * Mostramos la información de los alumnos utilizando los métodos de la clase String.

**Código Java:**

**public class Alumno {**

**// Atributos de la clase Alumno**

**private** **String** **nombre**;

**private** **String** **apellido**;

**private** **String** **dni**;

**// Constructor de la clase Alumno**

**public** **Alumno**(**String** **nombre**, **String** **apellido**, **String** **dni**) {

**this**.**nombre** = **nombre**;

**this**.**apellido** = **apellido**;

**this**.**dni** = **dni**;

}

**// Metodos get para obtener los datos del alumno.**

**public** **String** **getNombre()** {

**return** **nombre**;

}

**public** **String** **getApellido**() {

**return** **apellido;**

}

**public** **String** **getDni**() {

**return** **dni;**

}

**// Método para mostrar la información del alumno**

**public void mostrarInformacion() {**

**System.out.println**("Nombre completo: " + nombre + " " + apellido);

**System.out.println**("DNI: " + dni);

}

}

public class Main {

public static void main(String[] args) {

**// Creación de objetos Alumno**

Alumno **alumno1** = **new** Alumno("Juan", "Pérez", "12345678A");

Alumno **alumno2** = **new** Alumno("María", "García", "98765432B");

Alumno **alumno3** = **new** Alumno("Carlos", "López", "55555555C");

**// Mostrar información de los alumnos**

System.out.println("Información de los alumnos:");

System.out.println("-------------------------");

System.out.println("Alumno 1:");

alumno1.mostrarInformacion();

System.out.println("-------------------------");

System.out.println("Alumno 2:");

alumno2.mostrarInformacion();

System.out.println("-------------------------");

System.out.println("Alumno 3:");

alumno3.mostrarInformacion();

System.out.println("-------------------------");

// Ejemplo de métodos de la clase String

String nombreCompleto = alumno1.getNombre() + " " + alumno1.getApellido();

System.out.println("Nombre completo en mayúsculas: " + **nombreCompleto.toUpperCase());**

System.out.println("Longitud del nombre completo: " + **nombreCompleto.length());**

System.out.println("Inicial del nombre: " + alumno1.**getNombre().charAt(0));**

}

}

**Puntos Clave para Tus Alumnos:**

* La clase String es fundamental para trabajar con texto en Java.
* Los objetos String son inmutables, lo que significa que no se pueden modificar una vez creados.
* Java proporciona una amplia gama de métodos para manipular cadenas de caracteres, como toUpperCase(), length() y charAt().
* El uso de los metodos get para poder obtener información de los atributos privados de una clase.

**Class RAndom**

**Explicación Paso a Paso:**

* **Importar la Clase Random:**
  + Necesitamos importar la clase java.util.Random para poder usarla.
* **Crear un Objeto Random:**
  + Creamos un objeto de la clase Random para generar números aleatorios.
* **Generar Números Aleatorios:**
  + Utilizamos los métodos de la clase Random para generar diferentes tipos de números aleatorios:
    - nextInt(int bound): Genera un número entero aleatorio entre 0 (inclusive) y bound (exclusivo).
    - nextDouble(): Genera un número decimal aleatorio entre 0.0 (inclusive) y 1.0 (exclusivo).
    - nextBoolean(): Genera un valor booleano aleatorio (verdadero o falso).
* **Generar Contraseñas Aleatorias:**
  + Podemos usar la clase Random para generar contraseñas aleatorias, combinando letras, números y símbolos.