**Java Trainee**

Topics:

- Variables, Control structures, Exceptions

- Loop, If, do..while

- Objects & Classes

Realizar los siguientes ejercicios utilizando Java

1. Calcular el área de un triángulo dado la base y la altura
2. Dividir dos números e imprimir el resultado, manejar la excepción de división por cero.
3. Verificar si un número es positivo, negativo o cero
4. Imprimir los números del 1 al 10 utilizando un bucle do-while
5. Crear una clase llamada "Persona" con atributos nombre y edad, y un método para imprimir los datos
6. Calcular el promedio de tres números
7. Convertir una cadena de texto en un número entero, manejar la excepción de formato incorrecto
8. Verificar si un año es bisiesto o no
9. Leer números del usuario hasta que se ingrese un valor negativo, luego imprimir la suma de los números ingresados
10. Crear una clase llamada "Rectangulo" con atributos base y altura, y un método para calcular el área.

Consideraciones:

Para lograr los objetivos el desarrollador puede:

* Utilizar la IDE que le resulte más cómodo.
* Herramientas como Docker, Git, UML.
* Organizar el proyecto como mejor le parezca.
* Los ejercicios tienen que ser ejecutables.

Teoría

1. Variables, Control structures, Exceptions:
   * ¿Qué es una variable en Java? ¿Cómo se declara y se asigna un valor a una variable?
   * Explica la diferencia entre una estructura de control if y una estructura de control switch.
   * ¿Qué es una excepción en Java? ¿Cómo se maneja una excepción utilizando try-catch?
2. Loop, If, do..while:
   * ¿Cuál es la diferencia entre un bucle for y un bucle while en Java?
   * ¿Cómo se utiliza la estructura de control if-else para tomar decisiones en un programa?
   * ¿Cuál es la diferencia entre un bucle do-while y un bucle while? ¿Cuándo usarías cada uno de ellos?
3. Objects & Classes:
   * ¿Qué es una clase en Java? ¿Cuál es la diferencia entre una clase y un objeto?
   * Explica el concepto de encapsulación y cómo se logra en Java.
   * ¿Cuál es la diferencia entre una variable de instancia y una variable de clase?
4. Variables, Control structures, Exceptions:
   * ¿Qué es el ámbito (scope) de una variable en Java? ¿Cuáles son los diferentes niveles de ámbito?
   * ¿Qué es una estructura de control try-catch-finally en Java? ¿Por qué es útil en el manejo de excepciones?
   * ¿Cuál es la diferencia entre una excepción verificada y una excepción no verificada?
5. Loop, If, do..while:
   * ¿Cómo se utiliza un bucle for-each en Java? ¿En qué casos es útil?
   * ¿Cuál es la diferencia entre un bucle while y un bucle do-while? ¿Cuándo usarías cada uno?
   * ¿Cómo se utiliza la estructura de control if-else-if en Java?
6. Objects & Classes:
   * ¿Qué es la herencia en Java? ¿Cómo se implementa?
   * Explica el concepto de polimorfismo en Java. ¿Cómo se logra?
   * ¿Cuál es la diferencia entre una clase abstracta y una interfaz en Java?
7. Variables, Control structures, Exceptions:
   * ¿Qué es una constante en Java? ¿Cómo se declara y se utiliza?
   * ¿Cuál es el propósito de la estructura de control try-with-resources en Java?
   * ¿Cómo se lanza una excepción personalizada en Java?
8. Loop, If, do..while:
   * ¿Cuál es la diferencia entre un bucle for y un bucle do-while en Java? ¿En qué casos usarías cada uno?
   * ¿Cómo se utiliza la estructura de control if-else anidada en Java?
   * ¿Cuál es la diferencia entre un bucle while y un bucle do-while? ¿Cuándo usarías cada uno?
9. Objects & Classes:
   * ¿Qué es la sobrecarga de métodos en Java? ¿Cómo se implementa?
   * Explica el concepto de encapsulación y por qué es importante en la programación orientada a objetos.
   * ¿Cuál es la diferencia entre una clase abstracta y una interfaz en Java? ¿Cuándo usarías cada una?
10. Variables, Control structures, Exceptions:
    * ¿Qué es el ámbito (scope) de una variable en Java? ¿Cuáles son los diferentes niveles de ámbito?
    * ¿Qué es una excepción en Java? ¿Cuál es la diferencia entre una excepción verificada y una excepción no verificada?
    * ¿Cómo se manejan las excepciones en Java utilizando try-catch-finally?