



Programación Java

➤ Introducción al lenguaje Java

// Práctica integradora

Objetivo

El objetivo de esta guía práctica es que podamos afianzar y profundizar los conceptos sobre clase abstracta, herencia y sobrescritura. Para esto vamos a plantear una serie de ejercicios simples que nos permitirán repasar los temas que estudiamos.

¡Vamos todavía!



Ejercicio 1

1. Crear una clase Password. El constructor debe recibir una regex (expresión regular) que valide el formato requerido para la contraseña. Investigar en Internet qué recursos provee Java para operar con expresiones regulares.
2. Agregar el método public void setValue(String pwd) que permite asignar una contraseña acorde a la requerida por la regex del punto anterior. En caso de no



concordar, el método debe arrojar una excepción.

- a. Discusión: ¿qué tipo de excepción consideras más adecuada?
 - b. Implementar el método según lo decidido en (i).
3. Escribir tres clases: PasswordFuerte, PasswordIntermedia, PasswordSimple, todas deben extender a Password y hardcodear una regex acorde al nivel de seguridad que el mismo nombre de la clase indica.



Ejercicio 2

1. Crear la clase FiguraGeometrica, abstracta y con el método: `public abstract double area();`
2. Sobrecribir adecuadamente el método `toString`. Discutir en grupo: ¿tiene sentido sobrecribir el método `toString` en una clase abstracta?
3. Crear las clases Circulo, Triangulo y Rectangulo, todas subclases de FiguraGeometrica. ¿Qué error de compilación aparece? ¿Cómo lo solucionarías?
4. Agregar los atributos (variables de instancia) que corresponda en cada caso y sobrecribir adecuadamente el método `area` en todas las subclases de FiguraGeometrica.
5. Crear una clase utilitaria con un método estático: `public static double areaPromedio (FiguraGeometrica arr[])`, que calcule y retorne el área promedio de las figuras contenidas en el array. Crear un programa, instanciar un `FiguraGeometrica[]`, cargarle



IT BOARDING

BOOTCAMP

datos e invocar a `areaPromedio` para obtener el área promedio. Mostrar el resultado.