

Programación sobre Redes - Guía de ejercicios 3

Nota: Todas las clases e interfaces que se creen a partir de ahora deben hacerse en un mismo package.

Compilador online: <https://www.online-java.com/>

Instalador: <https://www.eclipse.org/downloads/packages/>
<https://sourceforge.net/projects/eclipse-neon-portable/>

Ejercicio 1

Crear una clase llamada Alumnos que tenga:

- Como atributos Nombre, Edad, DNI privados
- Constructor que inicialice los datos al generar una instancia
- Setters y Getters para modificar los atributos
- Método que muestre "El alumno **X** con DNI **X** tiene **X** años", *siendo X los atributos creados*.

Luego, en una clase Escuela, generar 2 instancias(con datos iniciales que decidan) de Alumnos para:

- Ver su información completa
- Modificar sus edades
- Ver nuevamente la información, actualizada

```
int[] vector = new vector[2];
```

```
Alumnos vector = new Alumnos();
```

Ejercicio 2

Retomar el ejercicio anterior. En vez de que se pase la edad al constructor, que se envíe la fecha en que nació el alumno y hacer el cálculo en un método dentro de la clase Alumnos. Luego, mostrar toda la información de un alumno en la clase Escuela.

Resolución: [Escuela.java](#) - [Alumnos.java](#)

Ejercicio 3 - <https://www.online-java.com/WCoyRS7vEm>

Crear una clase llamada Calculadora que permita ingresar dos números(atributos) y sepa responder a las siguientes consultas (utilizar solo una instancia para la comprobación):

- ¿Cuál es la suma entre 56 y 102?
- ¿Cuál es la resta entre 73 y 21?
- ¿Cuál es la multiplicación entre 12 y 128?

- ¿Cuál es la división de 146 entre 18?

Ejercicio 4 - <https://www.online-java.com/1Dsl3VgJEW>

Crear una clase llamada Rectangle que tiene como atributos 2 pares de lados. Las instancias de la clase Rectangle deben saber responder los siguientes mensajes:

public double getArea();

Devuelve el área del rectángulo.

public double getPerimeter();

Devuelve el perímetro del rectángulo.

Ejercicio 5 - <https://www.online-java.com/KYbqzWVsiU>

Crear una clase llamada Vector3D que representa un vector en el espacio tridimensional. Las instancias de la clase Vector3D deben saber responder los siguientes mensajes:

public double getX();

Devuelve la coordenada x del vector.

public double getY();

Devuelve la coordenada y del vector.

public double getZ();

Devuelve la coordenada Z del vector.

public double norm();

Devuelve la norma del vector.

Ejercicio 6

Crear una clase llamada Matrix que representa una matriz de números reales. Las instancias de la clase

Matrix deben saber responder los siguientes mensajes:

public int rows();

Devuelve la cantidad de filas de la matriz.

public int columns();

Devuelve la cantidad de columnas de la matriz.

public double get(int row, int column);

Devuelve el valor que se encuentra en la fila row y columna column de la matriz.

public void set(int row, int column, double value);

Establece el valor que se encuentra en la fila row y columna column de la matriz al valor de value.

La clase Matrix debe proveer al menos los siguientes constructores:

public Matrix(int rows, int columns);

Crea una matriz de rows filas y columns columnas con todos los elementos inicializados a 0.

Teoría de Clases abstracta e interfaces

Ejercicio 7

Desarrolla un programa que utilice una clase abstracta llamada Animal para representar animales genéricos. Define propiedades comunes a todos los animales, como nombre y edad, y métodos abstractos que representen acciones comunes, como comer y emitir sonidos. Luego, crea subclases concretas para representar diferentes tipos de animales, como "Perro", "Gato" y "Pájaro". Implementa los métodos abstractos en cada subclase para definir el comportamiento específico de cada animal. En el programa principal, crea instancias de estas subclases y muestra cómo interactúan con las acciones comunes de la clase abstracta Animal.

Ejercicio 8

Crea un programa en Java que utilice interfaces para realizar cálculos matemáticos y calcular el área de un círculo. Debes seguir estos pasos:

Define una interfaz llamada Calculable con un método double calcular() que permita realizar cálculos matemáticos.

Implementa la interfaz Calculable en una clase llamada Calculadora que realice operaciones matemáticas básicas, como suma, resta, multiplicación y división. La clase Calculadora debe recibir dos operandos y un operador (puede ser +, -, *, o /) y devolver el resultado de la operación. Asegúrate de manejar la división por cero.

Implementa la interfaz Calculable en una segunda clase llamada CalculoCirculo que calcule el área de un círculo. La clase CalculoCirculo debe recibir el radio del círculo y devolver el área ($\pi * \text{radio}^2$).

En el programa principal, crea instancias de la clase Calculadora para realizar cálculos matemáticos y una instancia de la clase CalculoCirculo para calcular el área de un círculo. Muestra los resultados de los cálculos en la consola.