

1. Escribe un programa en Java con una clase llamada **Semaforo** con un atributo **color** y otro **duracion** que se pueden consultar y modificar. Además habrá un constructor a partir de un color y una duración pasados por parámetro. Crea 2 métodos en la clase: uno devolverá un valor booleano que valdrá true si el color del semáforo es verde y false si no; el otro método hará lo mismo pero para el color rojo.

2. Escribe un programa en Java con una clase llamada **Validaciones** que tendrá los siguientes **métodos estáticos**: (1 punto)

- Método que comprueba si la variable **numérica entera** pasado por parámetro es mayor que 10 y menor o igual que 100. Si no es así, lanzará una excepción indicando el error.
- Método que comprueba si una variable de **texto** pasada por parámetro tiene una longitud mayor que 2 y que NO comienza por el texto "ex". Si no es así, lanzará una excepción indicando el error.
- Método que comprueba si un variable de tipo carácter es una letra mayúscula. Si no es así, lanzará una excepción indicando el error.

3. Crea un programa en Java con una clase llamada **Avion** con los atributos **numeroVuelo**, **fechaSalida** y **retraso** que solamente se podrán consultar. Además habrá un método que devolverá un valor booleano que será true si el avión tiene retraso. Crea 2 objetos de la clase **Avion**, establece sus atributos usando el constructor (1 de ellos tendrá retraso) y muestra su **numeroVuelo**, **fechaSalida** y si tiene **retraso** o no. La salida esperada sería como:

***El avión JB3455 con fecha de salida 2024-12-03 tiene retraso.***

***El avión XG3789 con fecha de salida 2024-10-10 NO tiene retraso.***

4. Escribe un programa en Java que te pregunte tu color favorito y que compruebe si el texto introducido tiene el texto "verde" (ignorando mayúsculas y minúsculas). Si es así, se mostrará por la consola un mensaje diciendo que el lenguaje Java inicialmente se llamaba green. Si no el programa acaba. La salida esperada sería algo así:

***Dime tu color favorito: El color VeRdE.***

***El lenguaje Java inicialmente se llamaba green.***

5. Escribe un método en Java que compruebe si el parámetro de texto **numeroVuelo** sigue el formato de un número de vuelo y devuelva true si es así, o false si no. El formato de un número de vuelo es un texto de 7 caracteres formado por 2 letras mayúsculas, seguidos de 4 números que pueden representar un rango de valores de 1000 a 4999 y seguido de otra letra mayúscula. **Por ejemplo AZ1390G sería un número de vuelo válido pero bG0399a no.**

6. Escribe un método en Java que se le pase 2 números enteros: 1 llamado **numElementos** y otro llamado **numerolnicial**. El método devolverá un array de números enteros con tantos elementos como indique numElementos y estará inicializado con números enteros con el primer elemento igual a **numerolnicial**, el segundo igual a numerolnicial más 1, el tercero igual a numerolnicial + 2 y así sucesivamente.

7. Escribe un programa en Java con una clase llamada **Animal** con un método **hacerRuido** que muestre por consola el texto "el animal hace ruido". Crea una subclase llamada **Gato** que sobrescriba el método **hacerRuido** y muestre por consola el texto "miau".

8. Crea un programa en Java con una clase abstracta llamada **Forma2D** con 2 métodos abstractos llamados **calcularArea** y **calcularPerimetro** que devuelvan un valor numérico con decimales y que no tengan parámetros. Crea la subclase **Circulo** que hereda de **Forma2D** e implementa dichos métodos. La clase **Circulo** necesitará un campo radio para ello. El área de un círculo se pueden calcular como **PI\*radio al cuadrado** y el perímetro como **2\*PI\*radio**. El valor de PI se obtiene escribiendo Math.PI.

9. Crea un programa en Java con una interfaz llamada **Redimensionable** que tendrá los métodos **void redimensionarAncho(int ancho)** y **void redimensionarAlto(int alto)**. Crea una clase llamada **Rectangulo** que implemente la interfaz **Redimensionable**. Para ello la clase **Rectangulo** necesitará un campo alto y otro ancho.