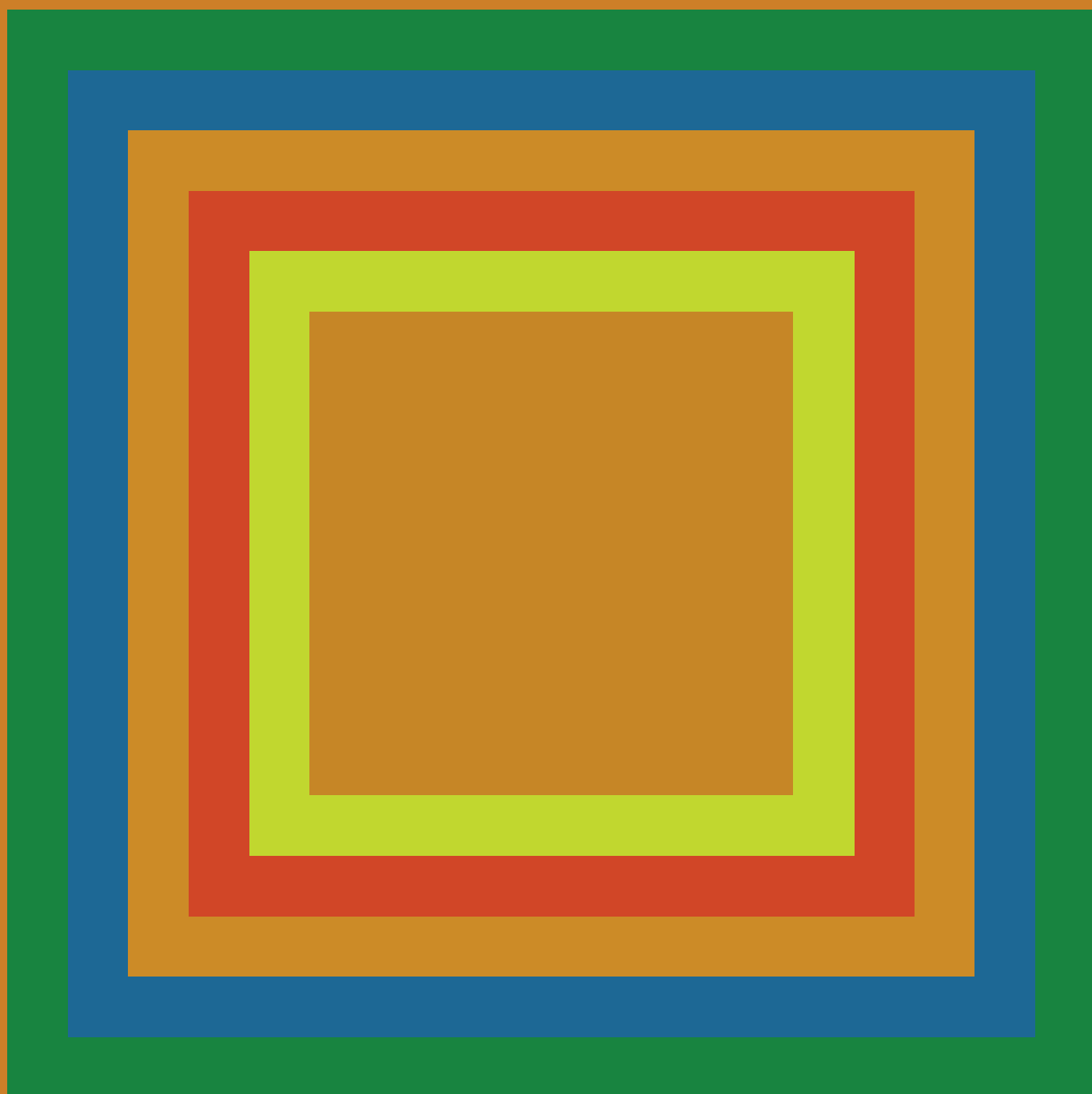


VACACIONES SANTALLANA

PROGRAMACIÓN



Bloque 1

Esta práctica consiste en la realización de 5 ejercicios sobre desarrollo de algoritmos. Deberás realizarlas de acuerdo a las especificaciones.

Ejercicio 1

Necesitamos contar de 5 en 5, comenzando por 50 y terminando por 100. Realiza el código necesario para que pinte ese resultado por la terminal.

Ejercicio 2

Realiza el ejercicio 1 con codificaciones diferentes. (for, while, do while)

Ejercicio 3

Diseña el algoritmo para un programa que permita introducir varios números y los vaya sumando. El programa terminará cuando el usuario introduzca un número negativo, en cuyo caso no debe incluirse en la suma.

Ejercicio 4

Codifica un algoritmo para un programa que te permita introducir por teclado la edad de 10 personas y muestre al final los siguientes datos:

Nº de mayores de edad.

Nº de menores de edad.

Media de edades.

Ejercicio 5

Codifica un algoritmo para un programa que te permita introducir por teclado una calificación numérica y muestre un texto en pantalla indicando si es un aprobado o un suspenso. El algoritmo principal debe invocar a una función que recibe la calificación numérica y retorna un texto con los valores "Aprobado" o "Suspenso". Debes realizar el ordinograma y el pseudocódigo.

Ejercicio 6

Mejora el pseudocódigo del ejercicio 5 para que no solo muestre "Aprobado" o "Suspenso", sino que además indique si es "Muy deficiente", "Insuficiente", "Bien", "Notable", etc.

Bloque 2

Esta práctica consiste en la realización de cinco pequeños programas en Java que deberás desarrollar y entregar a tu tutor con las siguientes pautas:

Ejercicio 1

Declara una variable llamada num1 y asígnale un valor al azar del 1 al 20. Luego declara otra variable llamada num2. Si num1 está comprendido entre 5 y 15 (ambos inclusive) asigna a num2 el resultado de la operación num1+15, de lo contrario asigna a num2 el resultado de la operación num1+5. Al final muestra en pantalla el valor de ambas variables.

Ejercicio 2

Necesitamos contar en orden inverso de 10 en 10 desde 100 hasta 10 (100, 90, 80, ..., 10). Realiza dos programas en Java, cada uno de ellos con dos estructuras de control distintas, que realicen la tarea indicada.

Ejercicio 3

Realiza un programa en Java que genere un número al azar entre 1 y 10. Una vez generado, si el número es mayor de 5 mostrará en pantalla el mensaje "Hoy va a ser un gran día", de lo contrario mostrará el mensaje "Hoy va a ser el mejor día del mundo".

Ejercicio 4

Realiza un programa en Java que vaya generando números al azar entre 1 y 10. Cada número generado debe mostrarse en pantalla, terminando cuando se genere el número 7. El programa también deberá informar sobre cuántos números se han tenido que generar para obtener el 7.

Ejercicio 5

Realiza un programa en Java que genere un número al azar entre 1 y 5. Una vez generado deberás utilizar una estructura alternativa múltiple (switch) para escribir en pantalla una cadena de texto diferente, según el número generado.



Para realizar los ejercicios que te vamos a proponer necesitas implementar la entrada de datos por teclado, es decir, permitir al usuario que introduzca un valor y recoger dicho valor en el programa. Puedes basarte en este pequeño ejemplo:

```
import java.util.Scanner;

public class Prueba {

    public static void main(String[] args) {

        Scanner lector = new Scanner(System.in);

        String nombre;

        System.out.println("¿Cómo te llamas? ");

        nombre = lector.nextLine();

        System.out.println("Hola "+nombre);

        lector.close();

    }

}
```

La ejecución desde Eclipse quedaría así:

```
Scanner lector = new Scanner(System.in);
```

La clase Scanner sirve para construir objetos que representan flujos de datos de entrada o lectura. En este caso hemos creado el objeto pasando como argumento System.in (flujo de entrada por teclado).

```
nombre = lector.nextLine();
```

Permite al usuario introducir una línea de texto por teclado y la almacena en la variable nombre.

```
lector.close();
```

Debemos cerrar el objeto Scanner cuando ya no lo necesitamos.

La clase Scanner tiene muchas más funciones que estudiarás más adelante. Ahora lo importante es que puedas implementar la entrada de datos por teclado.

Esta práctica consiste en la realización de dos programas que deberás desarrollar:

Ejercicio 1

Escribe un programa Java que sirva para realizar un pequeño test de inglés al usuario. El test consistirá en la traducción de 20 palabras del inglés al español. Las palabras del test se extraerán de una matriz de 20 filas y 2 columnas. Comienza por este código y luego completa lo que falta.

Ejercicio 2

Escribe un programa Java que funcione de acuerdo a las siguientes especificaciones:

- Debe permitir al usuario introducir un texto por teclado.
- Se mostrará en pantalla la longitud del texto (número de caracteres).
- Se mostrará el texto en mayúsculas.
- Se mostrará el texto separado por palabras (cada palabra en una línea) usando el método `split()`.
- Se contará el número de letras minúsculas.
- Se mostrará un triángulo con el texto introducido.

Ejercicio 3

Escribe un programa Java que funcione de acuerdo a las siguientes especificaciones:

- Debe permitir al usuario introducir varios textos terminando cuando escriba la palabra FIN. Cada texto debe guardarse en un objeto de tipo `String`.
- Por cada texto introducido (menos la palabra FIN) el programa debe informar del número de caracteres.
- Se deben ir concatenando todos los textos introducidos convertidos a mayúsculas (menos la palabra FIN) en un objeto de tipo `StringBuilder`.
- Dentro del objeto `StringBuilder` cada texto nuevo debe terminar con el carácter punto y coma.
- Por último, debemos utilizar un objeto de la clase `StringTokenizer` para volver a mostrar los textos introducidos en líneas separadas.

Bloque 3

Codifica las siguientes clases y genera una clase main para probar las diferentes operaciones.

Clase 1: Suma. Esta clase implementará todos los métodos que están relacionados con la suma de la calculadora. Esta clase tendrá los siguientes métodos:

- Suma de dos números reales, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Suma de dos números enteros, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Suma de tres números reales, tendrá 3 parámetros de entrada y uno de salida que será la solución.
- Suma con valor acumulado, tendrá un parámetro de entrada y la clase deberá de guardar el valor acumulado.

Clase 2: Resta. Esta clase implementará todos los métodos que están relacionados con la resta de la calculadora. Esta clase tendrá los siguientes métodos:

- Resta de dos números reales, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Resta de dos números enteros, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Resta de tres números reales, tendrá 3 parámetros de entrada y uno de salida que será la solución.
- Resta con valor acumulado, tendrá un parámetro de entrada y la clase deberá de guardar el valor acumulado.

Clase 3: Producto. Esta clase implementará todos los métodos que están relacionados con la multiplicación de la calculadora. Esta clase tendrá los siguientes métodos:

- Producto de dos números reales, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Producto de dos números enteros, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Producto de tres números reales, tendrá 3 parámetros de entrada y uno de salida que será la solución.
- Potencia, tendrá dos parámetro de entrada (base y exponente) y uno de salida que será la solución

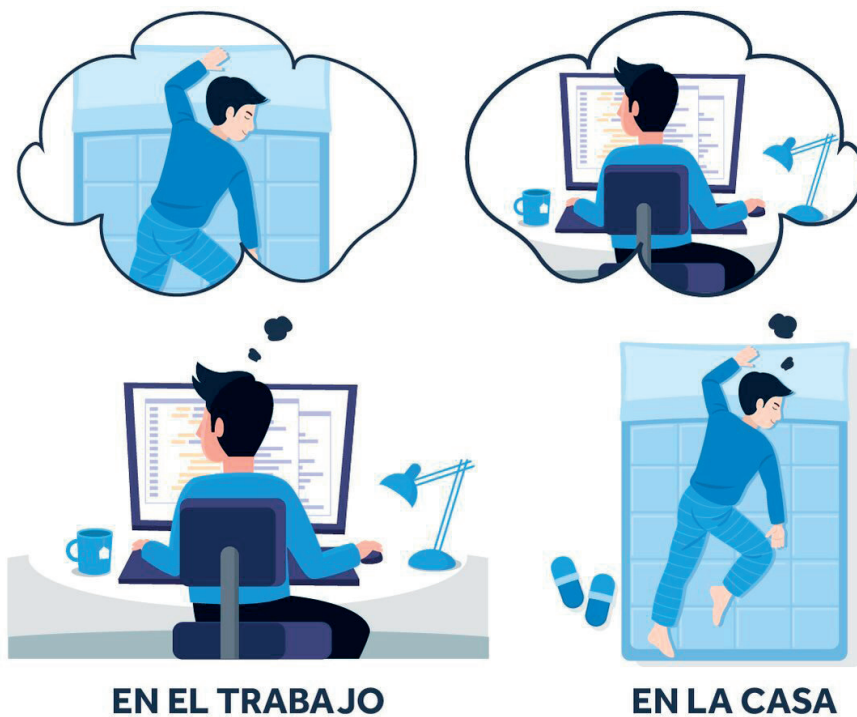
Clase 4: Cociente. Esta clase implementará todos los métodos que están relacionados con la división de la calculadora. Esta clase tendrá los siguientes métodos:

- División de dos números reales, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- División de dos números enteros, tendrá 2 parámetros de entrada y uno de salida que será la solución.
- Inverso de un número real, tendrá un parámetro de entrada y uno de salida que será la solución.
- Raíz de un número, tendrá un parámetro de entrada y uno de salida que será la solución.

Documentación de casos especiales, unos ejemplos podrían ser:

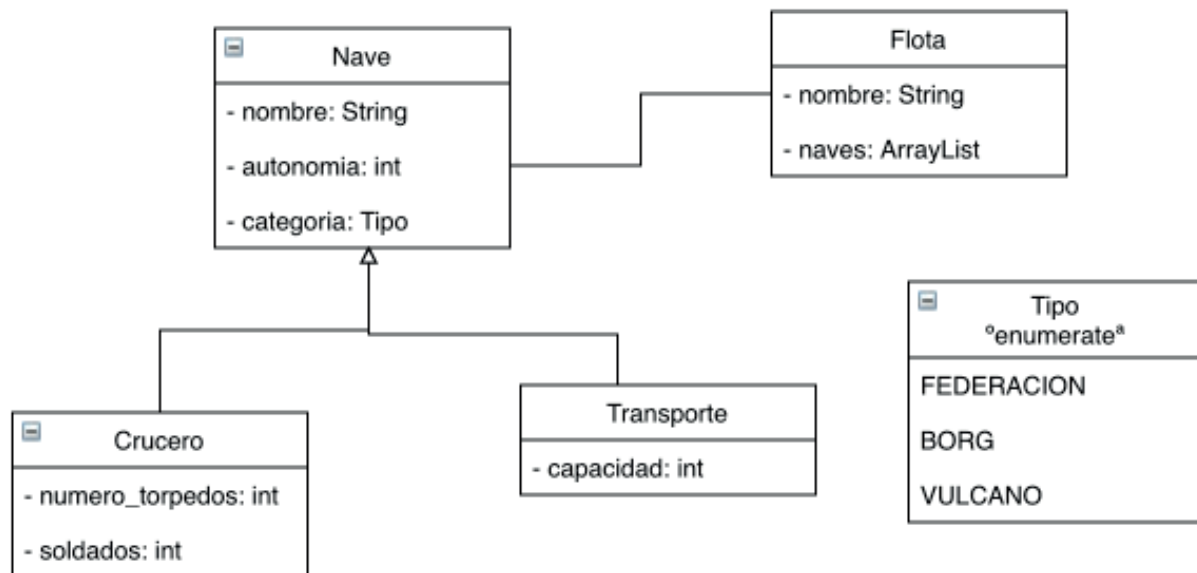
- ¿Qué ocurre cuando alguno de los parámetros de entrada es cero?
- Supongamos que la calculadora no puede utilizar números negativos ¿Cómo lo documentaríamos?
- ¿Qué ocurriría si la división no da un número exacto?
- ¿Qué ocurriría con la potencia si los números pasados son muy grandes?

LA VIDA DE UN PROGRAMADOR:



Bloque 4

Códifica en Java las siguientes clases para el modelo descrito en el siguiente diagrama.

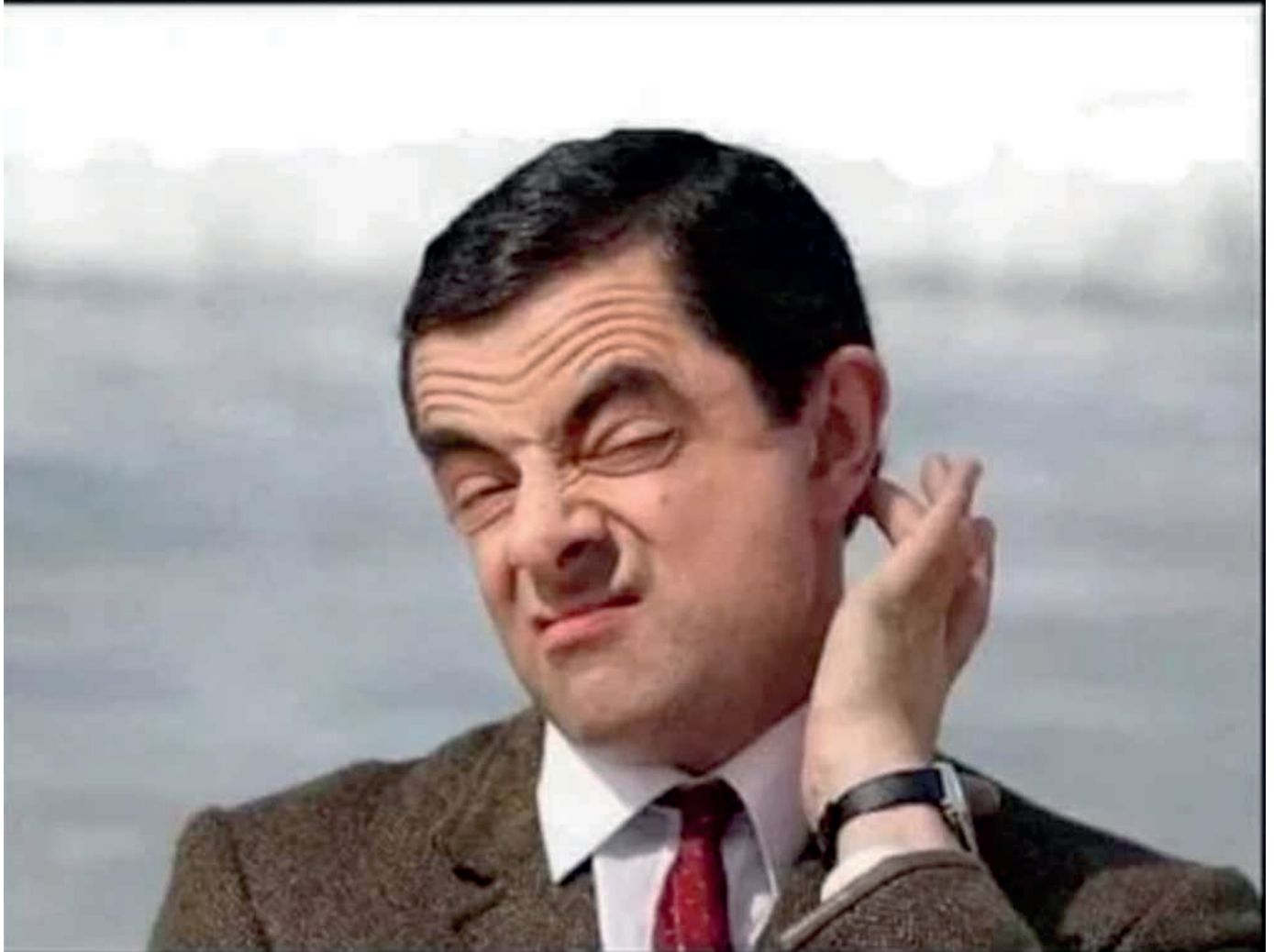


Crea los métodos y constructores que consideres necesarios para la correcta ejecución de las siguientes operaciones y restricciones.

- El programa no puede crear un objeto de tipo Nave.
- El programa de poder imprimir por la terminal todos los atributos de un objeto incluyendo el string de Tipo.
- El programa debe poder imprimir la suma del total de años luz que pueden recorrer sus naves. (la suma del atributo autonomía de todos los objetos creados).

Crea un clase main donde instanciaras varios objetos con datos inventados y prueba los metodos solicitados.

**Todavía no he decidido dónde pasar
Semana Santa ,**



si en el cuarto o en la sala.