

ESTRUCTURA DE DATOS

# PILAS

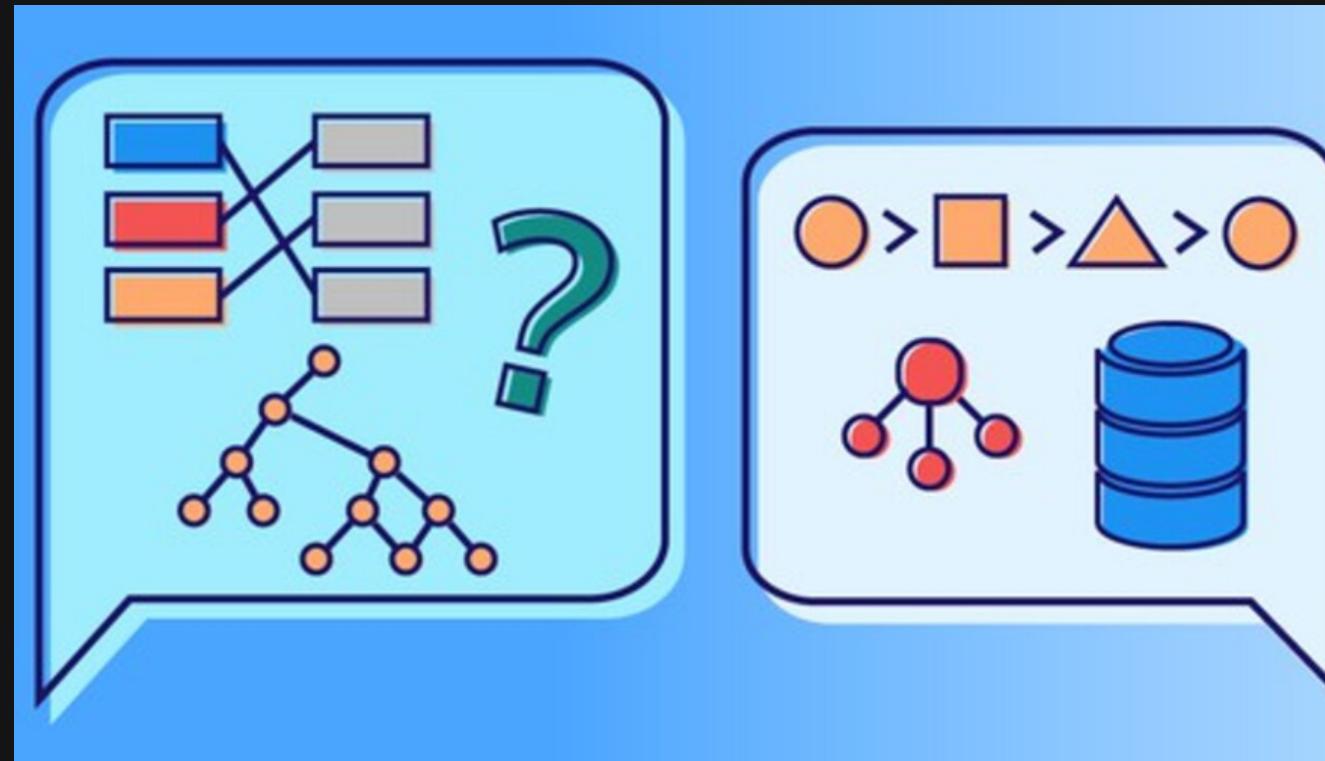
2023 - ROMARIO TOLA QUISPE



# Preguntas teóricas

# 1. ¿A que se refiere cuando se habla de ESTRUCTURA DE DATOS?

Son un modo de representar información en una computadora, aunque además, cuentan con un comportamiento interno. ¿Qué significa? Que se rige por determinadas reglas/restricciones que han sido dadas por la forma en que está construida internamente.



How many shortest-length paths are there to get from your house to the doughnut shop?

$${n \choose k} = \frac{n!}{(n-k)!k!}$$
$$e^{\pi i} + 1 = 0$$

4 up's  
7 right's

$${11 \choose 7} = {11 \choose 4} = 330 \text{ paths}$$

$B_7$   
 $B_9$

$P \quad Q \quad R \quad P \vee Q \quad P \vee R \quad (P \vee Q) \wedge (P \vee R)$

T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	F	T	F
F	F	F	F	F	F

$\exists, \forall, 11, 15, 19, 23, \dots$   
 $A_1 - A_0 = 4$   
 $A_2 - A_1 = 4$   
 $A_3 - A_2 = 4$   
 $\vdots$   
 $A_n - A_{n-1} = 4$   
 $A_n = A_0 + 4n$

$\sum_{n=0}^{\infty} 4^n = 342$   
 $S_n = 7 + 12 + 17 + 22 + \dots + 342$   
 $+ S_n = 342 + 337 + 332 + 327 + \dots + 7$   
 $2S_n = 349 + 349 + 349 + 349 + \dots + 349$   
 $2S_n = 349 \cdot 68$   
 $S_n = \frac{349 \cdot 68}{2}$   
 $S_n = 11866$

Original:  $\exists x \forall y (x \geq 2y \rightarrow x > y + 1)$   
Converse:  $\exists x \forall y (x > y + 1 \rightarrow x \geq 2y)$   
Negation:  $\neg [\exists x \forall y (\neg(x \geq 2y) \vee x > y + 1)]$   
 $\forall x \exists y (x \geq 2y \wedge x \leq y + 1)$   
Contrapositive:  $\exists x \forall y (x \leq y + 1 \rightarrow x < 2y)$

$v - e + f = 2$

P.I.E. Example:

$$6! - \left[ \binom{6}{1} 5! - \binom{6}{2} 4! + \binom{6}{3} 3! - \binom{6}{4} 2! + \binom{6}{5} 1! \right]$$

$(A \cup B \cup C) \cap (A \cap B \cap C)$

## 2. ¿Cuáles son los TIPOS DE ESTRUCTURA QUE EXISTE?

Estructura de dato estática:

- Arrays
- Matrices
- Enumeraciones

Estructura de dato dinámica:

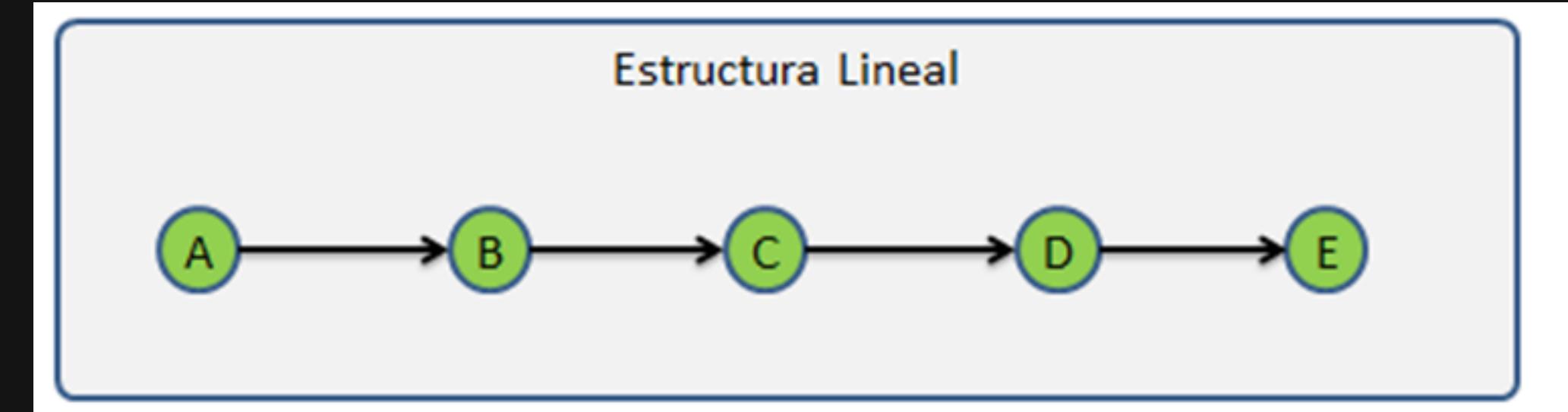
Lineales

No lineales



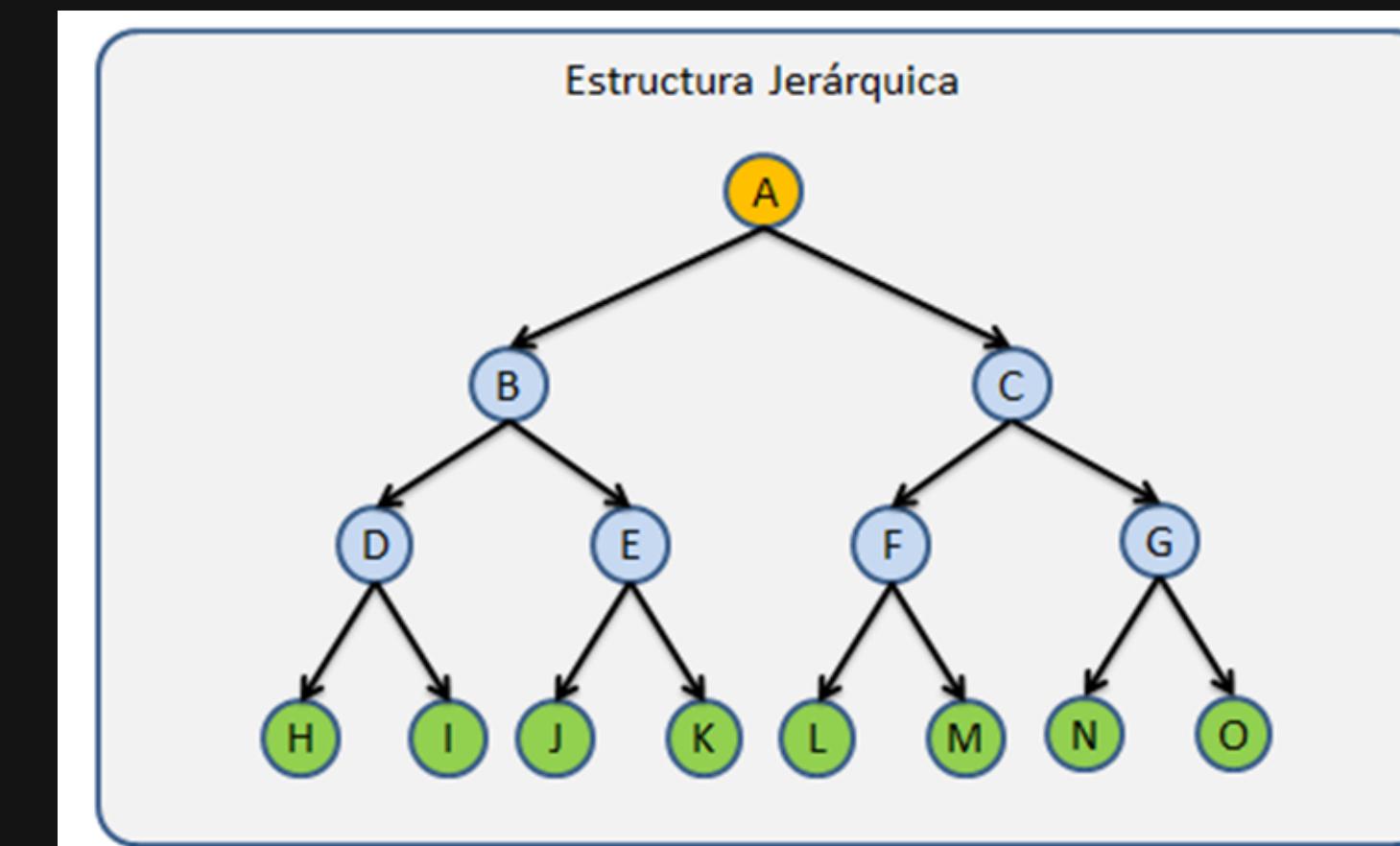
# Lineales:

- Listas Enlazadas
- Pilas
- Colas

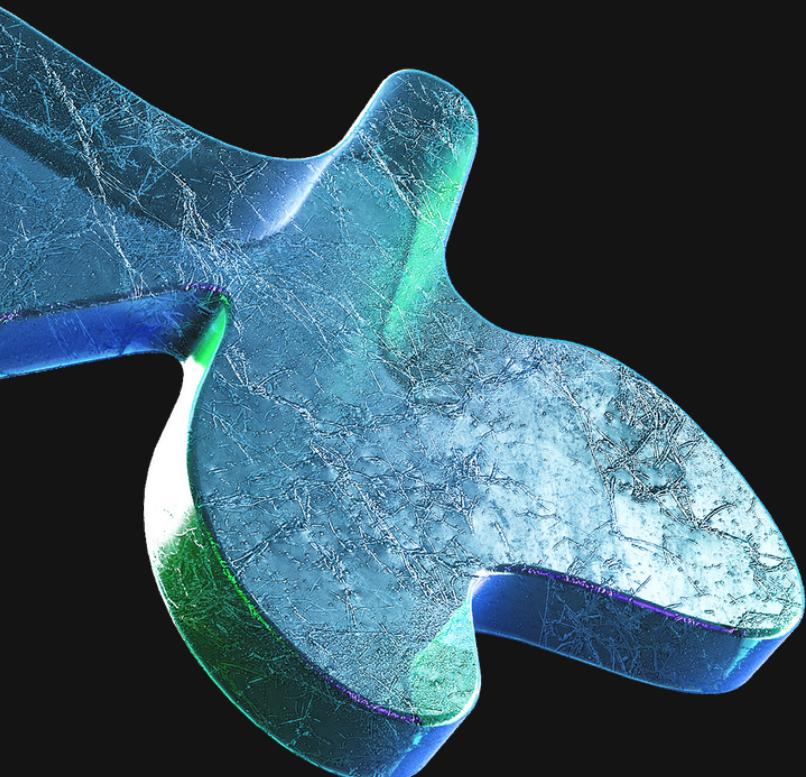


# No lineales:

- Arboles
- Grafos



### 3. ¿Por qué son útiles las estructuras de datos?.

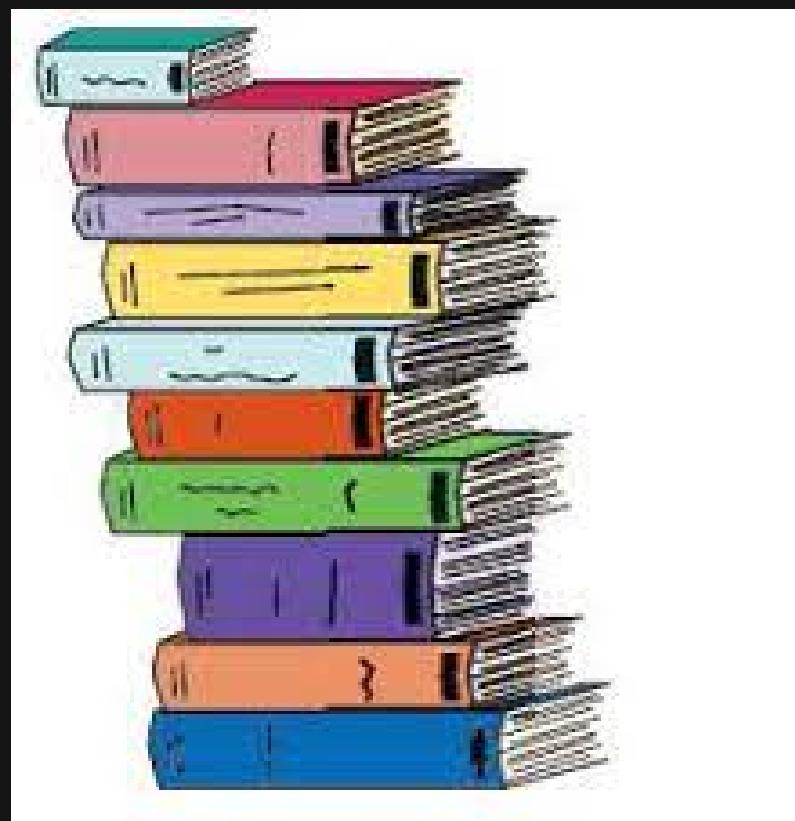
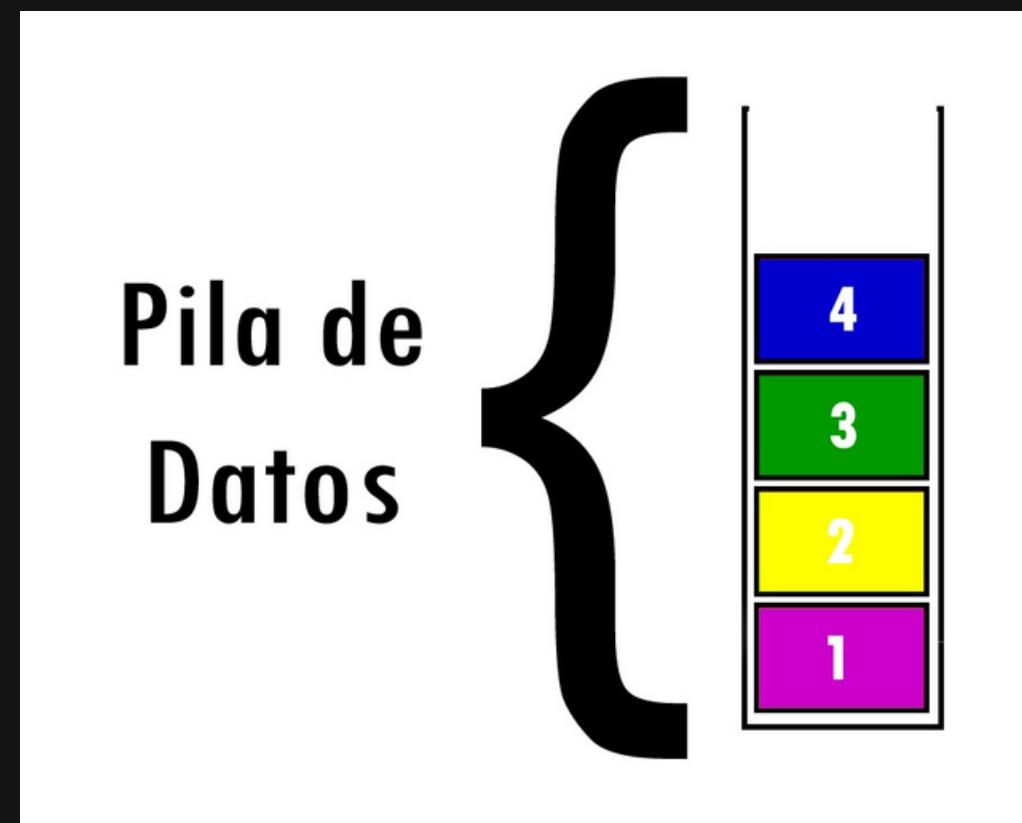


Te permitirá hacer software mas eficiente optimizando recursos, algo muy útil para entornos que trabajan con grandes cantidades de datos.

Proporcionan unas series de herramientas para solucionar ciertos tipos de problemas.

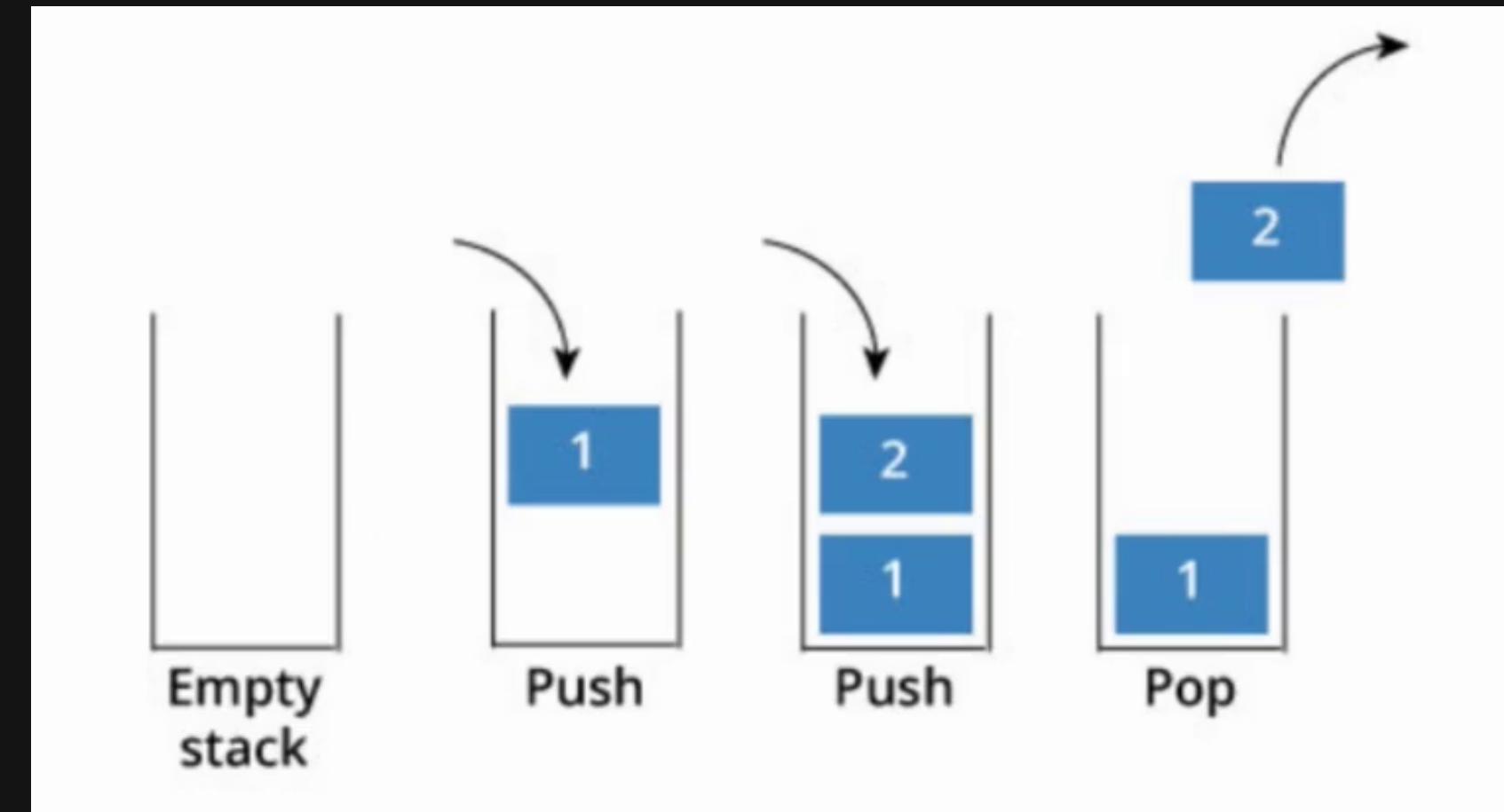
# 4. ¿Qué es una PILA?

Las pilas son un tipo de listas que tienen la particularidad de sólo poder eliminar o insertar en la cima de la lista. A estas acciones se le conocen como apilar y desapilar y conlleva a que el último elemento que ingresa a la pila sea el primero en salir a lo cual se le conoce como LIFO (Last in First out).



# 5. ¿Qué es STACK en JAVA, una STACK será lo mismo que una PILA?

Una Stack es una lista ordenada o estructura de datos que permite almacenar y recuperar datos, siendo el modo de acceso a sus elementos de tipo LIFO (del inglés Last In, First Out, «último en entrar, primero en salir»)



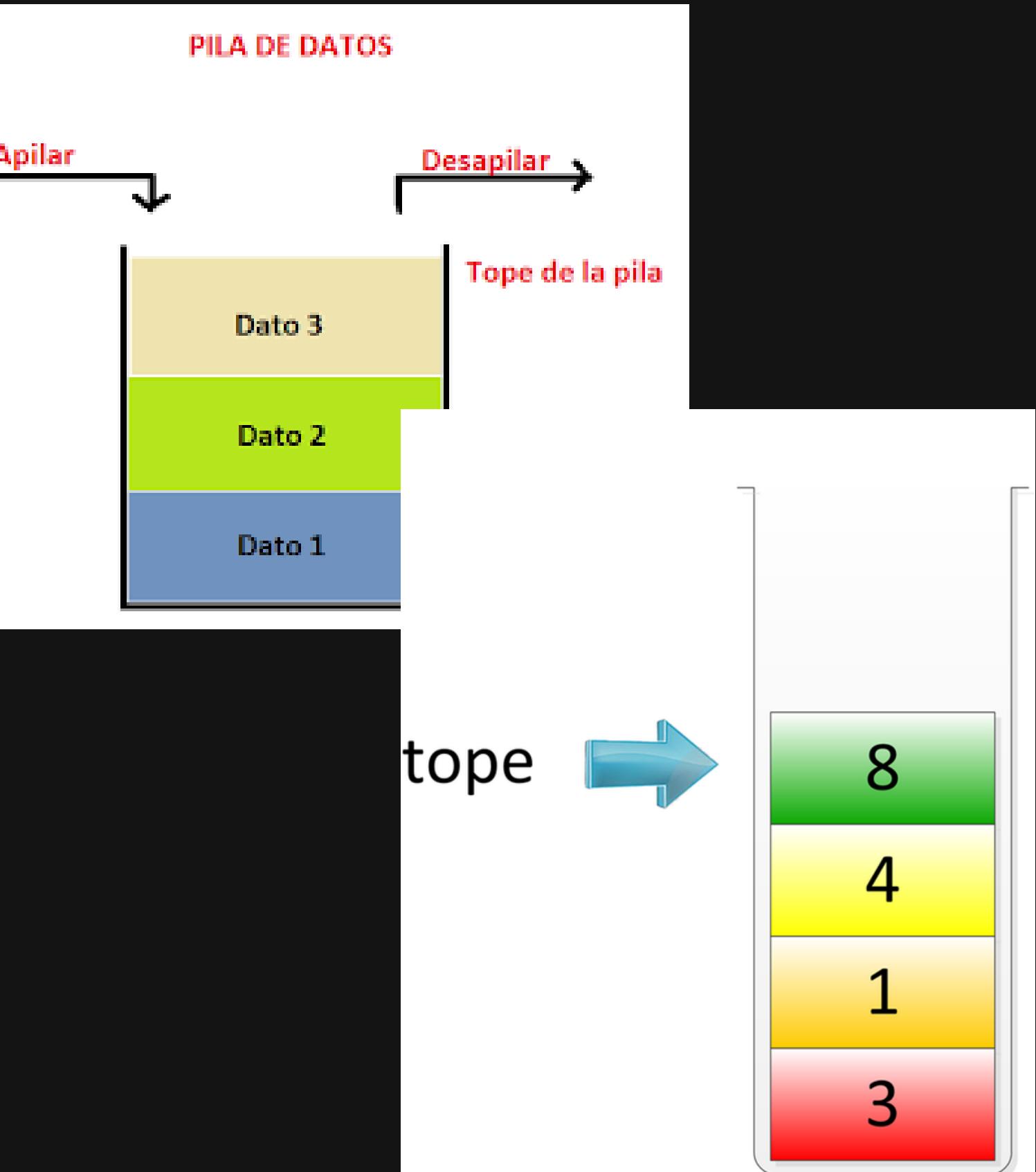
# 6. ¿Qué es TOPE en una PILA?

Es el elemento más recientemente insertado en la pila

Se encuentra en la cima de la misma.

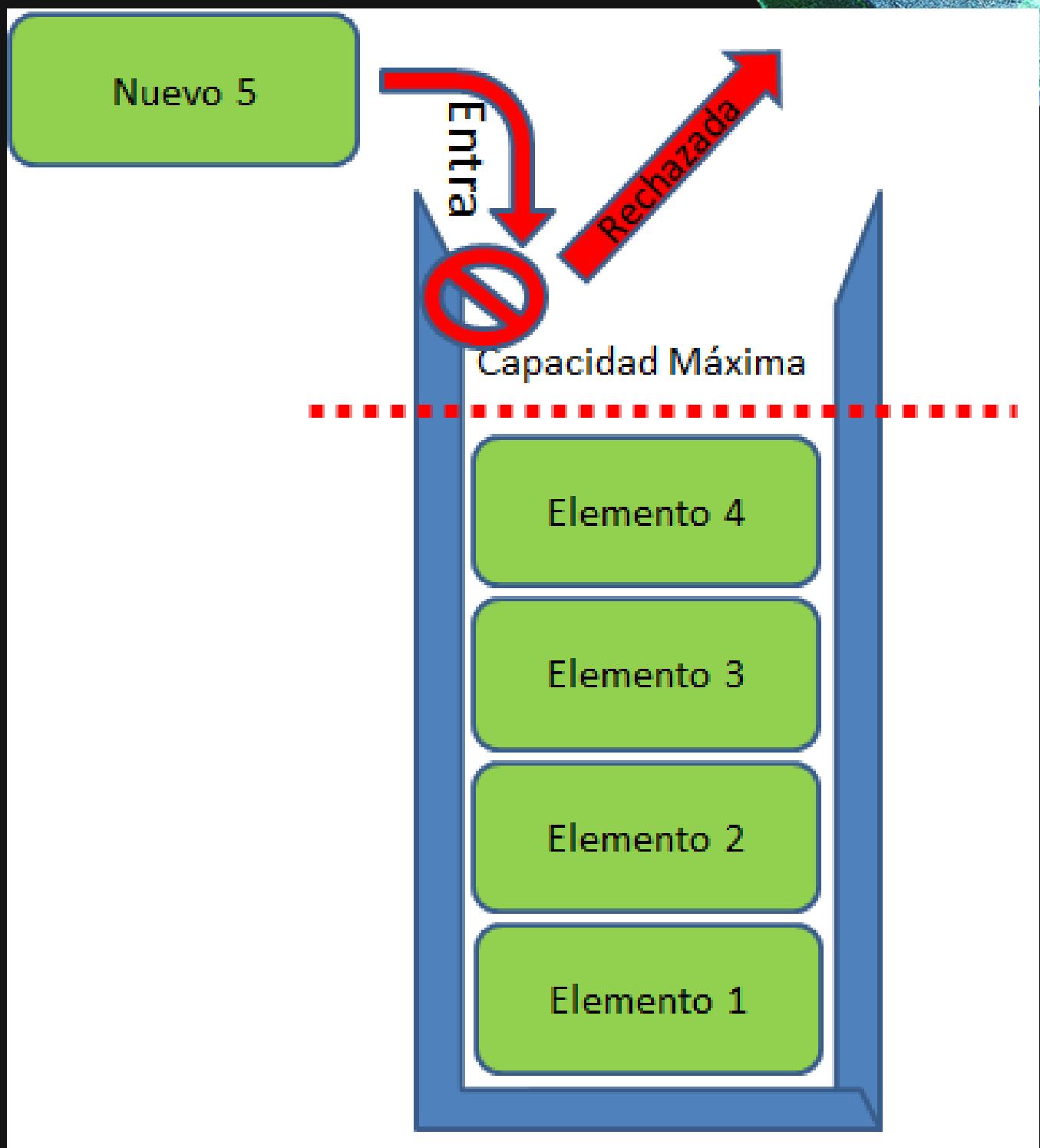
Es el único elemento que se puede eliminar de la pila en un momento dado, siguiendo el principio LIFO .

Es el elemento que se puede acceder o modificar en cualquier momento sin afectar a los demás elementos de la pila.



# 7. ¿Qué es MAX en una PILA?

Se refiere al tamaño máximo que puede alcanzar la pila. En algunos lenguajes de programación, se establece un tamaño máximo predefinido para las pilas, lo que significa que si se intenta agregar más elementos de los que la pila puede manejar, se producirá un desbordamiento o error de pila.



## 8. ¿A que se refiere los métodos esVacia() y esLlena() en una PILA?

```
8 usages  
public boolean esVacio() {  
    if (tope == 0) {  
        return true;  
    } else {  
        return false;  
    }  
}
```

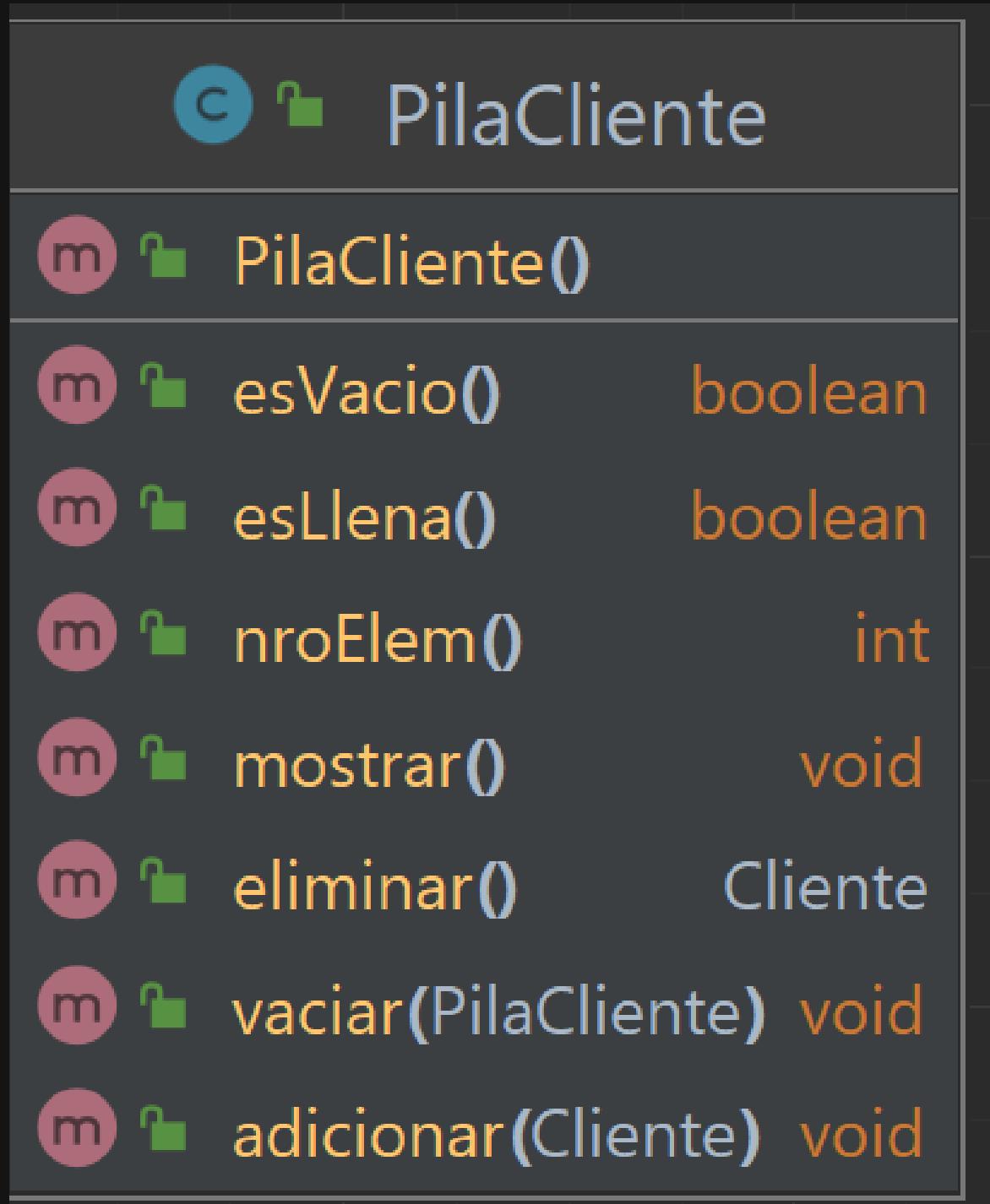
```
1 usage  
public boolean esLlena() {  
    if (tope == max) {  
        return true;  
    } else {  
        return false;  
    }  
}
```

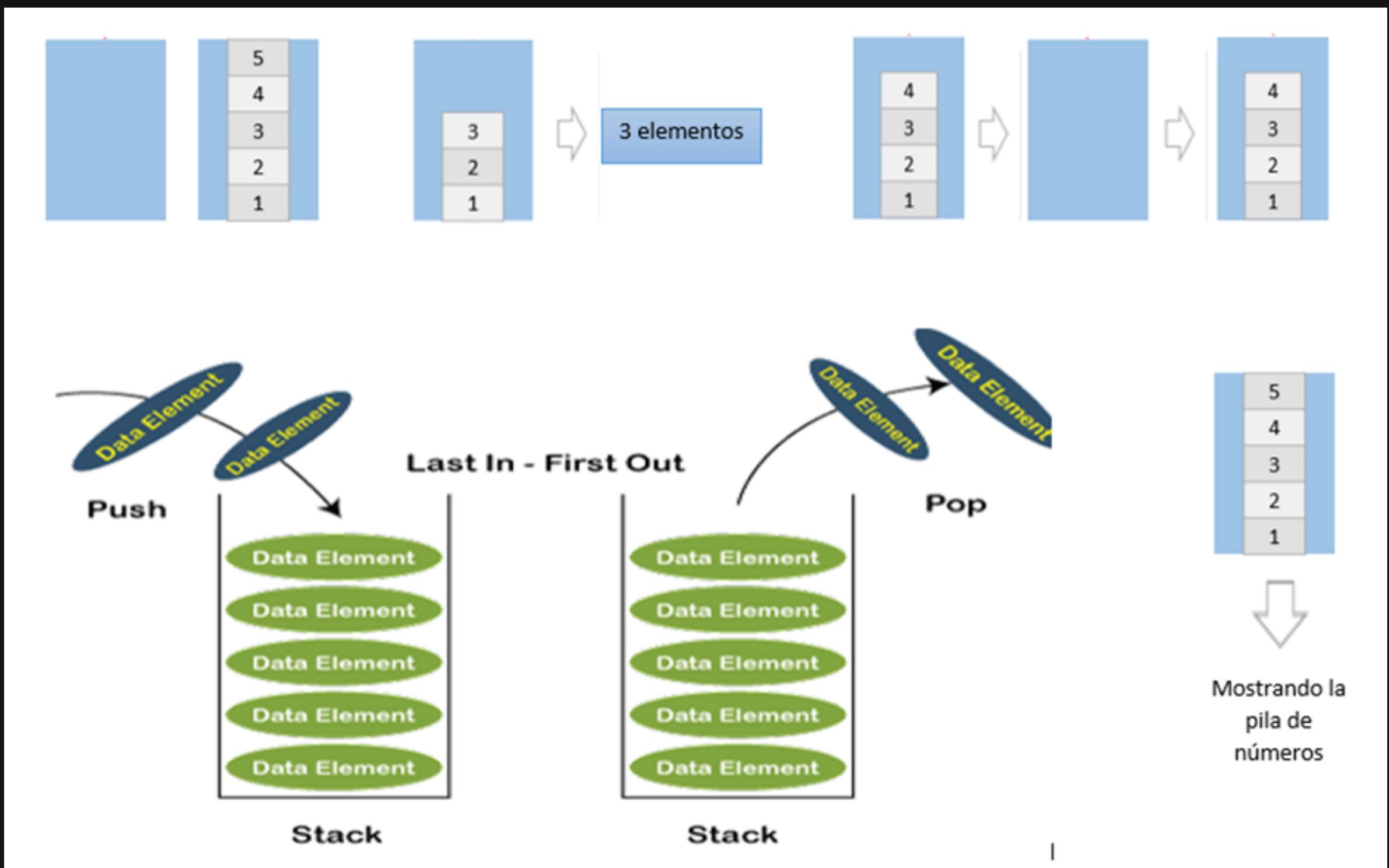
# 9. ¿Qué son los métodos estáticos en JAVA?

En Java, los métodos estáticos son aquellos que se definen en una clase y se pueden llamar sin tener que crear una instancia (objeto) de la clase en la que se definen. En otras palabras, un método estático pertenece a la clase en sí y no a una instancia específica de la clase.



# 10. ¿A través de un gráfico, muestre los métodos mínimos que debería de tener una PILA?



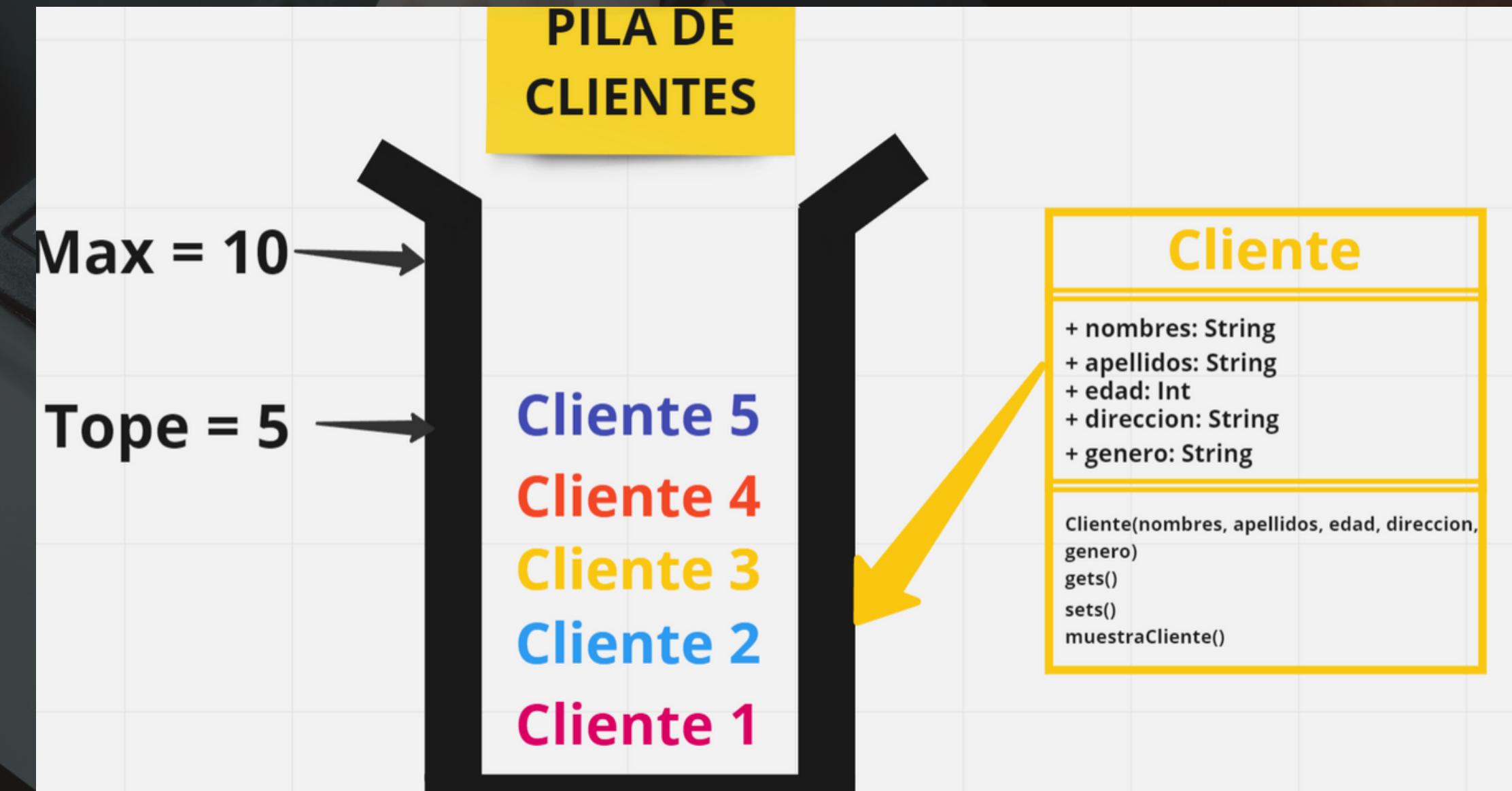


C PilaCliente		
m	PilaCliente()	
m	esVacio()	boolean
m	esLlena()	boolean
m	nroElem()	int
m	mostrar()	void
m	eliminar()	Cliente
m	vaciar(PilaCliente)	void
m	adicionar(Cliente)	void

# Parte Practica

# 11. Crear las clases necesarias para la PILA DE CLIENTES.

- Crear la clase Cliente
- Crear la clase PilaCliente
- Crear la clase Main.
- Crear un paquete de nombre PilaDeClientes (todas las clases deberán de estar dentro de este paquete)

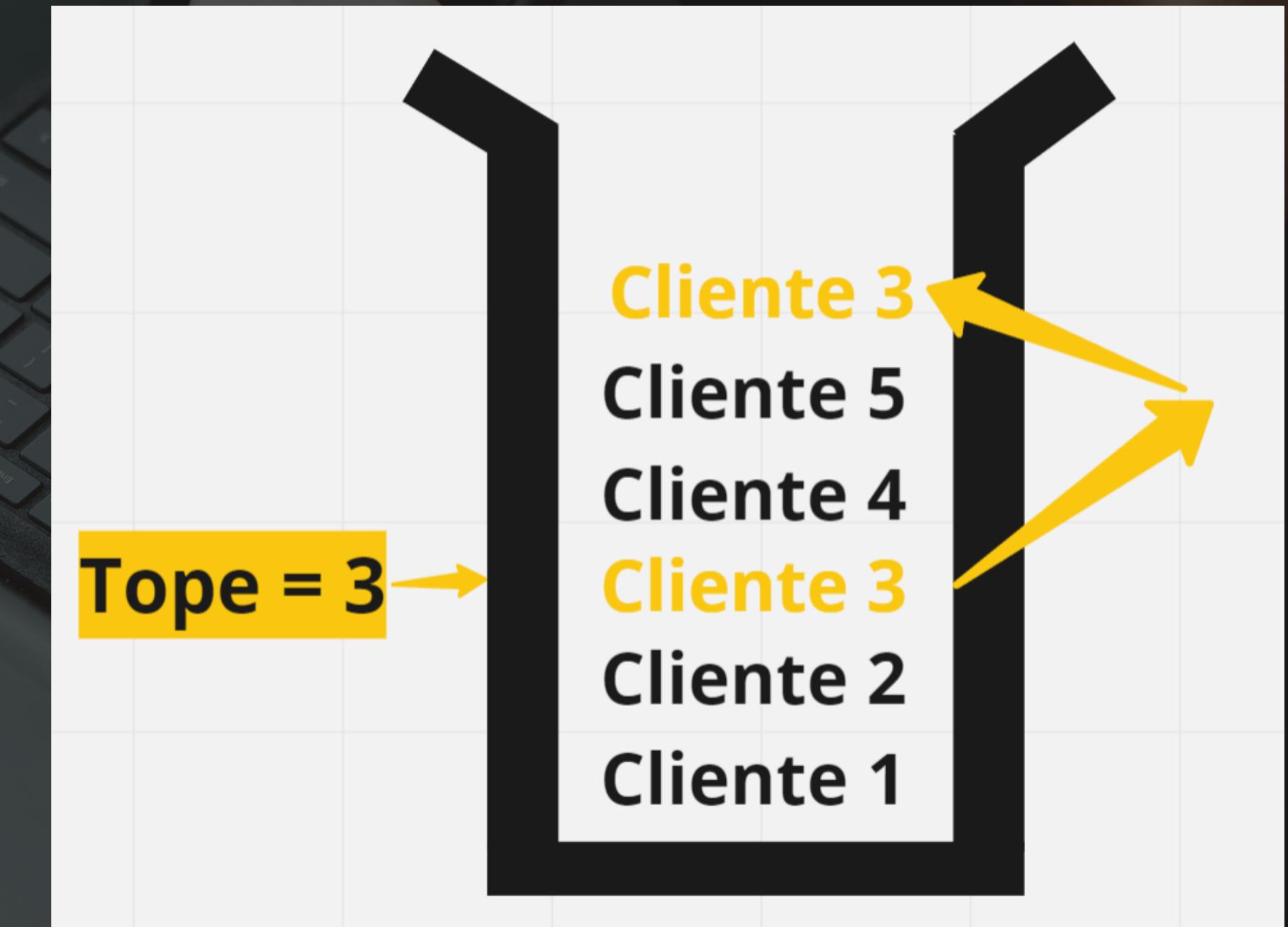


# 12.Determinar cuántos CLIENTES son mayores de 20 años.

- El método deberá llamarse mayoresCiertaEdad(Pila, edadMayor)
- El método debe ser creado en la clase MAIN como un método estático.
- El método recibe 2 parámetros
  - La Pila de Clientes
  - El valor de la edad.

# 13. Mover el k-ésimo elemento al final de la pila.

- El método deberá llamarse `kEsimoPosicion(Pila, valorTope)`
- El método debe ser creado en la clase MAIN como un método estático.
- El método recibe 2 parámetros
  - La Pila de Clientes
  - El valor(int) de la posición que moverá al final de la pila.

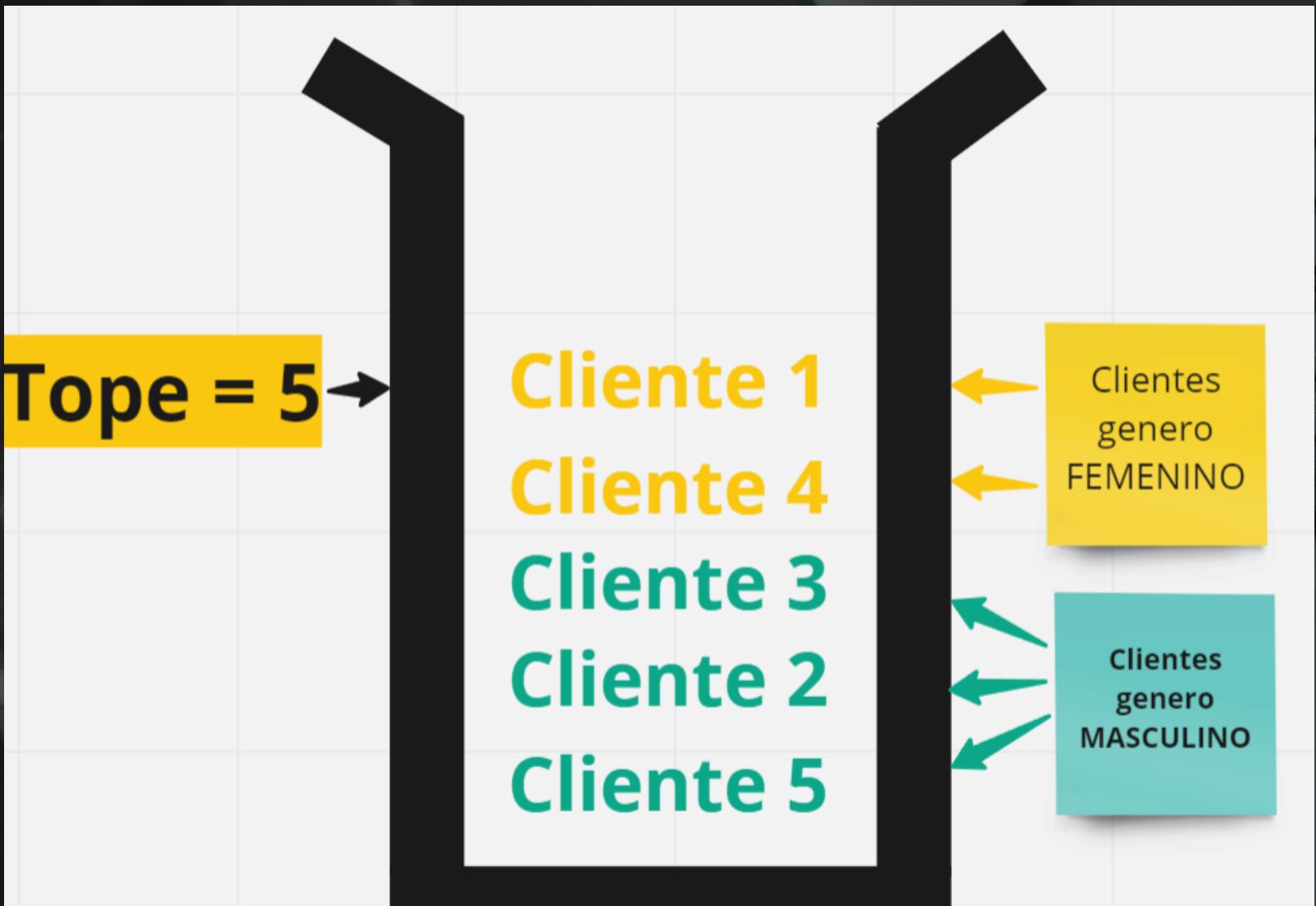


# 14.Cambiar la dirección de algunos CLIENTES de la PILA.



- El método deberá llamarse asignaDireccion(Pila, nuevaDireccion)
- El método debe ser creado en la clase MAIN como un método estático.
- El método recibe 2 parámetros
  - La Pila de Clientes
  - El valor(String) de la nueva dirección.
- Cambiar la dirección del cliente siempre y cuando el género sea FEMENINO.

# 15. Mover ÍTEMS de la PILA.



- El método deberá llamarse reordenaPila(Pila)
- El método debe ser creado en la clase MAIN como un método estático.
- El método recibe 1 parámetro  
■ La Pila de Clientes
- Mover a la base todos los clientes del género masculino y los del género femenino moverlos al final.

**GRACIAS !!**