



# DEFENSA HITO 3

## ESTRUCTURA DE DATOS



# ESTUDIANTE



JHONATAN DAVID  
ALANOCA BLANCO

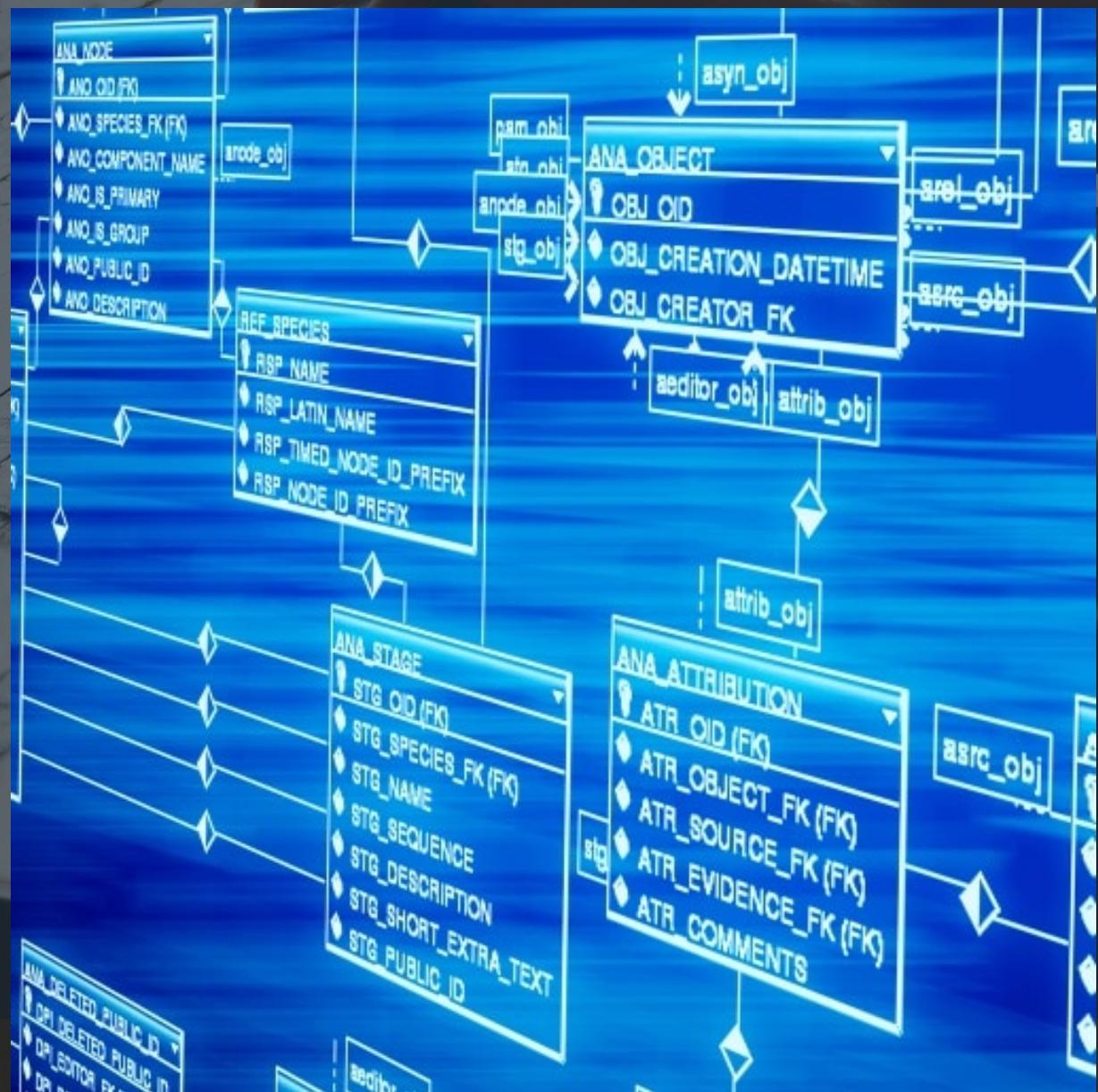


# MANEJO DE CONCEPTOS



# 1. ¿A QUE SE REFIERE CUANDO SE HABLA DE ESTRUCTURA DE DATOS?

Se refiere a una manera de organizar y almacenar datos en una computadora, para que puedan ser usados y manipulados de una manera eficiente.



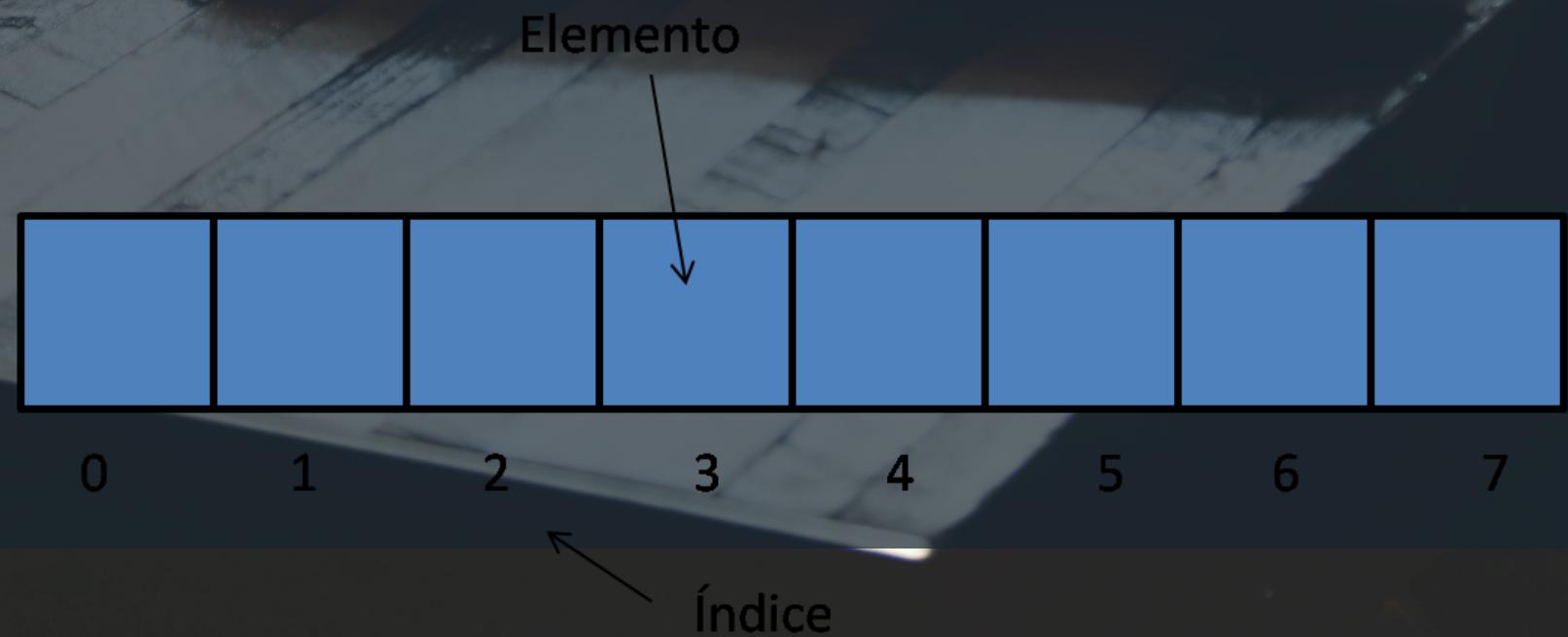
## 2. ¿CUÁLES SON LOS TIPOS DE ESTRUCTURA QUE EXISTE?

- **Externas:** Manejan grandes cantidades de datos que no pueden manejar la memoria principal de la computadora. Tienen la capacidad de almacenar grandes cantidades de datos.
  - **Base de datos:** Se utilizan para almacenar grandes cantidades de datos en una forma estructurada y organizada.
  - **Sistemas de almacenado en la nube:** Permiten almacenar y acceder a los datos en línea.
  - **Sistema de archivos:** Se utilizan para almacenar y organizar datos en un disco duro.

## 2. ¿CUÁLES SON LOS TIPOS DE ESTRUCTURA QUE EXISTE?

Las estructuras de datos se dividen en 2:

- **Internas:** Son utilizadas dentro de un objeto o componente de software. Se utilizan para almacenar y organizar los datos necesarios para que el objeto o componente funcione correctamente. Se dividen en 2 estáticas y dinámicas:
  - **Estáticas:** Son aquellas donde el tamaño no cambia.
  - **Arrays:** Almacena un conjunto fijo de elementos del mismo tipo.
  - **Matrices:** Es un array multidimensional donde los elementos se organizan en filas y columnas.



## 2. ¿CUÁLES SON LOS TIPOS DE ESTRUCTURA QUE EXISTE?

- **Dinámicas:** Son aquellas donde el tamaño puede cambiar durante la ejecución del programa.

- **Lineales:**

- listas
- pilas
- colas

- **No lineales:**

- árboles
- grafos

### 3. ¿POR QUÉ SON ÚTILES LAS ESTRUCTURAS DE DATOS?

Permiten a los programadores manejar y procesar grandes cantidades de datos de manera eficiente.

Permite hacer software mas eficiente optimizando recursos.

Tendrás una serie de herramientas para solucionar ciertos tipos de problemas

Mejoran el rendimiento y la eficiencia de los programas.

Ayudan a organizar el código de manera lógica y clara.

Reducen el tiempo y los costos asociados con el desarrollo y mantenimiento de software.

## **4. ¿QUÉ ES UNA PILA?**

Es una estructura de datos lineal.

Sigue el principio de "El ultimo en entrar, primero en salir"(LIFO).

Se asemeja a una pila de objetos en la vida real.

## **5. ¿QUÉ ES STACK EN JAVA, UNA STACK SERÁ LO MISMO QUE UNA PILA?**

Un stack en java es la forma inglesa de referirse a una pila, ambos son lo mismo pero Stack y pila son diferentes terminos que se usan para referirse a la misma accion.

## 6. ¿QUÉ ES TOPE EN UNA PILA?

El tope o "top" es el elemento que se encuentra en la posición más alta de una pila, es el último elemento que se inserta en la pila y es el primer elemento que se elimina si se requiere.

Se actualiza cada vez que se agrega o se elimina un elemento de la pila.

## 7. ¿QUÉ ES MAX EN UNA PILA?

Indica la cantidad máxima de elementos que puede contener la pila.

Es una variable que establece el límite superior del tamaño de la pila y se utiliza para comprobar si la pila está llena.

## 8. ¿A QUE SE REFIERE LOS MÉTODOS ESVACIA() Y ESLLENA() EN UNA PILA?

Las funciones "EsVacio()" y "EsLleno()" son utilizadas para verificar si la pila está vacía o llena, respectivamente.

La función EsVacio() verifica si el valor de top es igual a cero, lo que indica que no hay elementos en la pila. En ese caso, la función devuelve true, lo que significa que la pila está vacía. En caso contrario, devuelve false.

La función EsLleno() verifica si el valor de top es igual al valor máximo de la pila (max). Si top es igual a max, significa que la pila está llena y, por lo tanto, la función devuelve true. En caso contrario, devuelve false.

## 9. ¿QUÉ SON LOS MÉTODOS ESTÁTICOS EN JAVA?

Los métodos estáticos llevan la palabra "static" se utiliza para indicar que un campo, método o clase es compartido por todas las instancias de la clase y puede ser accedido sin necesidad de crear una instancia de la clase.

Esto significa que el método está disponible para ser llamado en cualquier momento, incluso antes de que se cree una instancia de la clase.

Los métodos estáticos se utilizan comúnmente para proporcionar una funcionalidad global y reutilizable

```
public static void Descuento(PilaDeLibros pila, String categoria){
```

```
    public static double mayor (PilaDeLibros pilaA ) {
```

# 10.¿A TRAVÉS DE UN GRÁFICO, MUESTRE LOS MÉTODOS MÍNIMOS QUE DEBERÍA DE TENER UNA PILA?

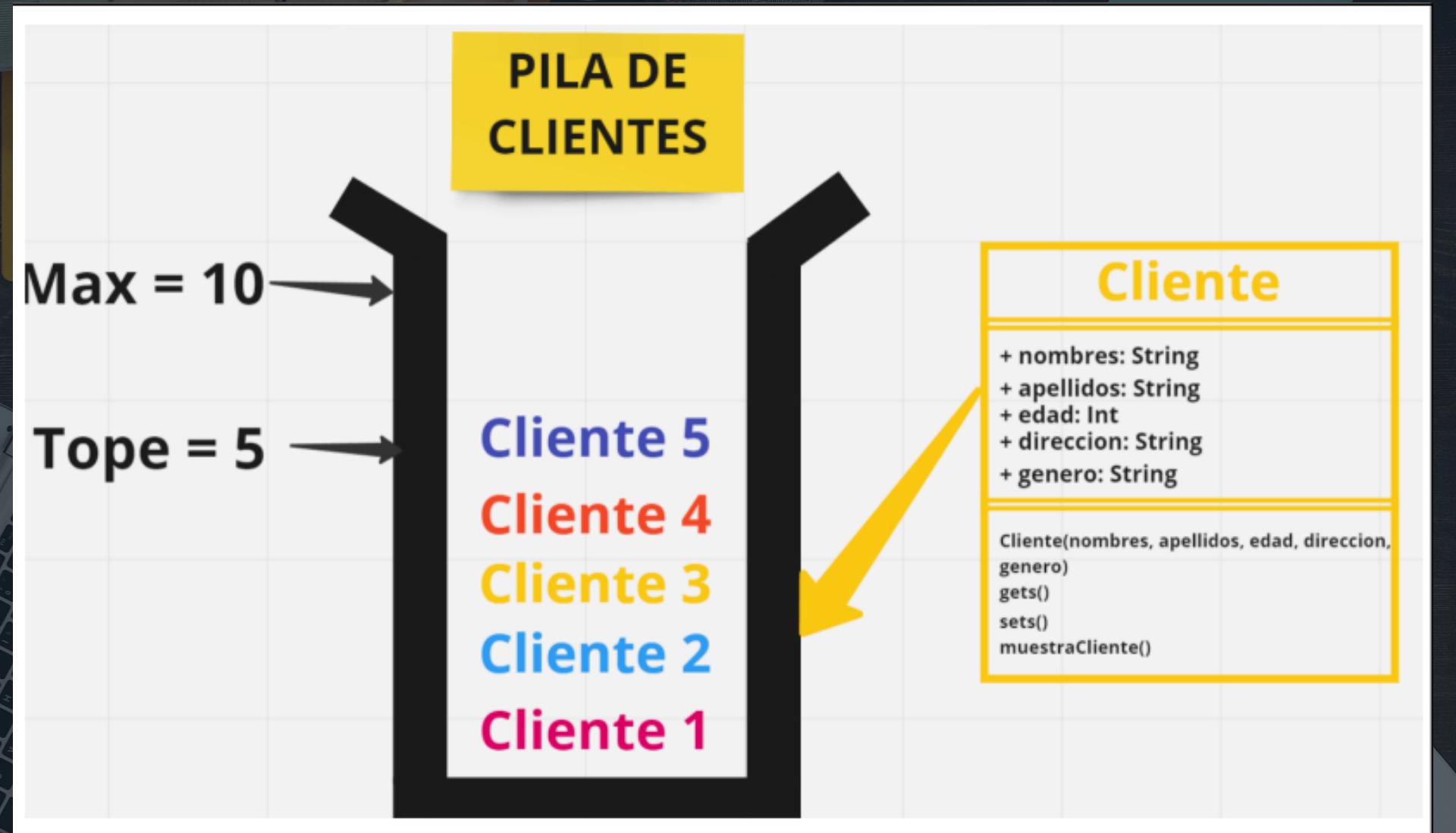
PilaDeCadenas		
m	PilaDeCadenas ()	
f	top	int
f	p	String []
f	max	int
m	EsLleno ()	boolean
m	EsVacio ()	boolean
m	NroElementos ()	int
m	Adicionar (String)	String
m	llenar (int)	void
m	Vaciar (PilaDeCadenas )	void
m	eliminar ()	String
m	Mostrar()	void



# PARTE PRACTICA

## 11. CREAR LAS CLASES NECESARIAS PARA LA PILA DE CLIENTES.

○ Diseño.



- 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)

## 11. CREAR LAS CLASES NECESARIAS PARA LA PILA DE CLIENTES.

- Adjuntar los siguientes.
- La clase MAIN con la creación de 5 clientes y agregados a la PILA.
- Una imagen de la salida de la consola en donde se muestran todos los ítems de la pila.

```
main
main()
asignaDireccion (PilaDeClientes , String) void
main (String [])
kEsimoPosicion (PilaDeClientes , int) void
MayoresCiertaEdad (PilaDeClientes , int) void
ReordenaPila (PilaDeClientes ) void
```

```
Cliente
Cliente (String , String , int, String , String )
edad int
direccion String
apellidos String
nombres String
genero String
setGenero (String ) void
setNombres (String ) void
getGenero () String
getApellidos () String
setEdad (int) void
MuestraCliente () void
getEdad () int
getNombres () String
getDireccion () String
setApellidos (String ) void
setDireccion (String ) void
```

```
PilaDeClientes
PilaDeClientes ()
max int
top int
Cliente Cliente []
Adicionar (Cliente ) void
Mostrar () void
EsLleno () boolean
NroElementos () int
EsVacio () boolean
eliminar () Cliente
Vaciar (PilaDeClientes ) void
```

## 11. CREAR LAS CLASES NECESARIAS PARA LA PILA DE CLIENTES.

```
public static void main(String[] argv){  
    Cliente cliente1 = new Cliente( nom: "Juan Carlos", ape: "Mamani Colque", ed: 19, dir: "Av. 6 de Marzo #4720, Villa Dolores, El Alto, Bolivia", gen: "Masculino");  
    Cliente cliente2 = new Cliente( nom: "Maria", ape: "Mancora Lopez", ed: 25, dir: "Calle Hermanos Manchego #100, Zona Sur, La Paz, Bolivia", gen: "Femenino");  
    Cliente cliente3 = new Cliente( nom: "Edson ivan", ape: "Condori Condori", ed: 21, dir: "Calle Juan de la Riva #1108, Zona Central, La Paz, Bolivia", gen: "Masculino");  
    Cliente cliente4 = new Cliente( nom: "Elena", ape: "Lopez Mendoza", ed: 19, dir: "Calle México #150, Villa Exaltación, El Alto, Bolivia", gen: "Femenino");  
    Cliente cliente5 = new Cliente( nom: "Sergio Andres", ape: "Mendoza Alvarado", ed: 20, dir: "Av. Julio Cesar Valdez #1250, Zona Villa Copacabana, La Paz, Bolivia", gen: "Masculino");  
    //Instanciando clase//  
    PilaDeClientes PilaDeClientes1 = new PilaDeClientes();  
    //Agregando clientes a la pila  
    PilaDeClientes1.Agregar(cliente1);  
    PilaDeClientes1.Agregar(cliente2);  
    PilaDeClientes1.Agregar(cliente3);  
    PilaDeClientes1.Agregar(cliente4);  
    PilaDeClientes1.Agregar(cliente5);  
    PilaDeClientes1.Mostrar();
```

# 11. CREAR LAS CLASES NECESARIAS PARA LA PILA DE CLIENTES.

Mostrando la PILA DE CLIENTES

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Sergio Andres

Apellidos: Mendoza Alvarado

Edad: 20

Direccion: Av. Julio Cesar Valdez #1250, Zona Villa Copacabana, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Elena

Apellidos: Lopez Mendoza

Edad: 19

Direccion: Calle Mexico #150, Villa Exaltación, El Alto, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Edson ivan

Apellidos: Condori Condori

Edad: 21

Direccion: Calle Juan de la Riva #1108, Zona Central, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Calle Hermanos Manchego #100, Zona Sur, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

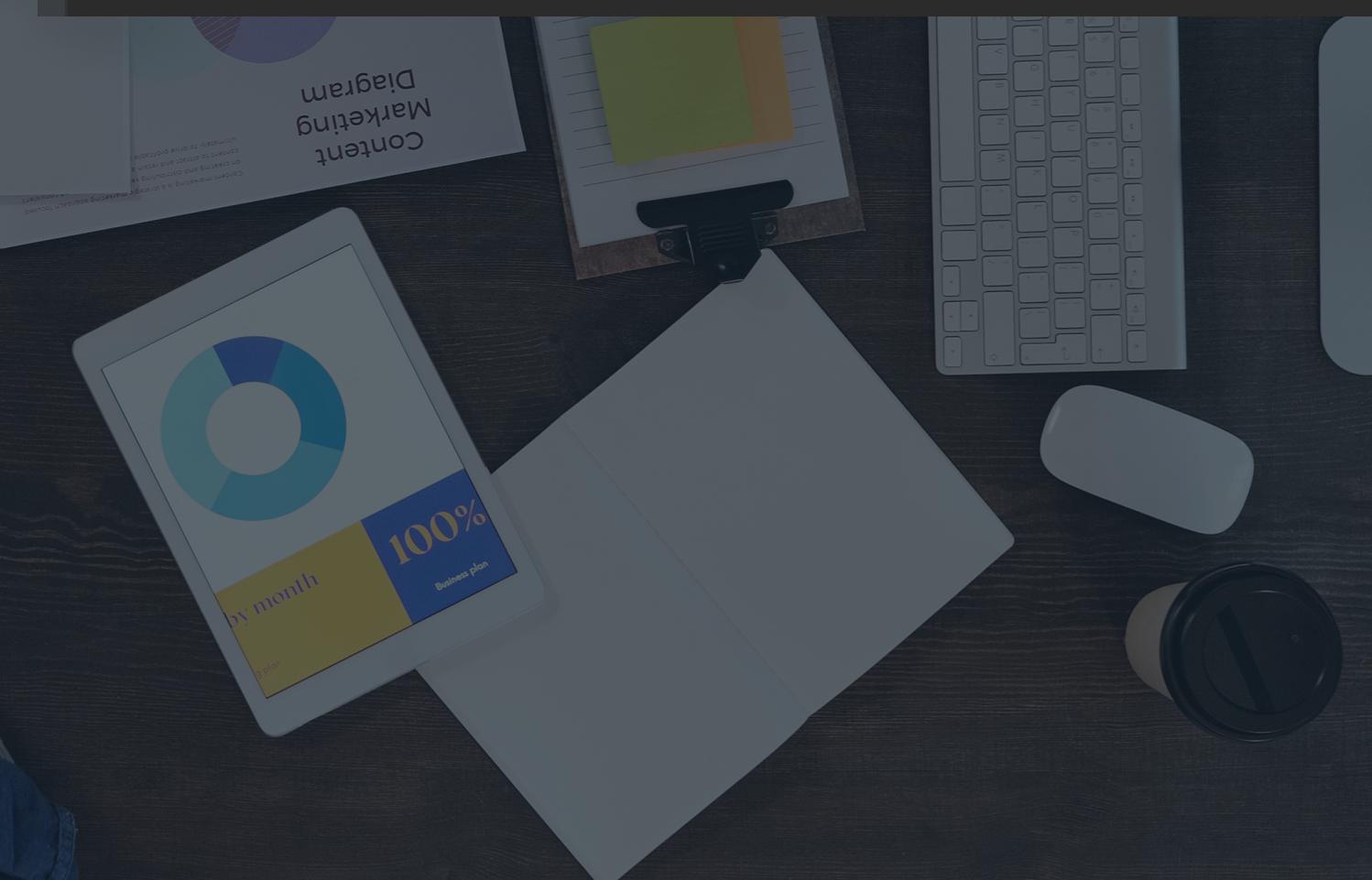
Nombre: Juan Carlos

Apellidos: Mamani Colque

Edad: 19

Direccion: Av. 6 de Marzo #4720, Villa Dolores, El Alto, Bolivia

Genero: Masculino



## 12. DETERMINAR CUÁNTOS CLIENTES SON MAYORES DE 20 AÑOS.

El método deberá llamarse `mayoresCiertaEdad(Pila, edadMayor)`

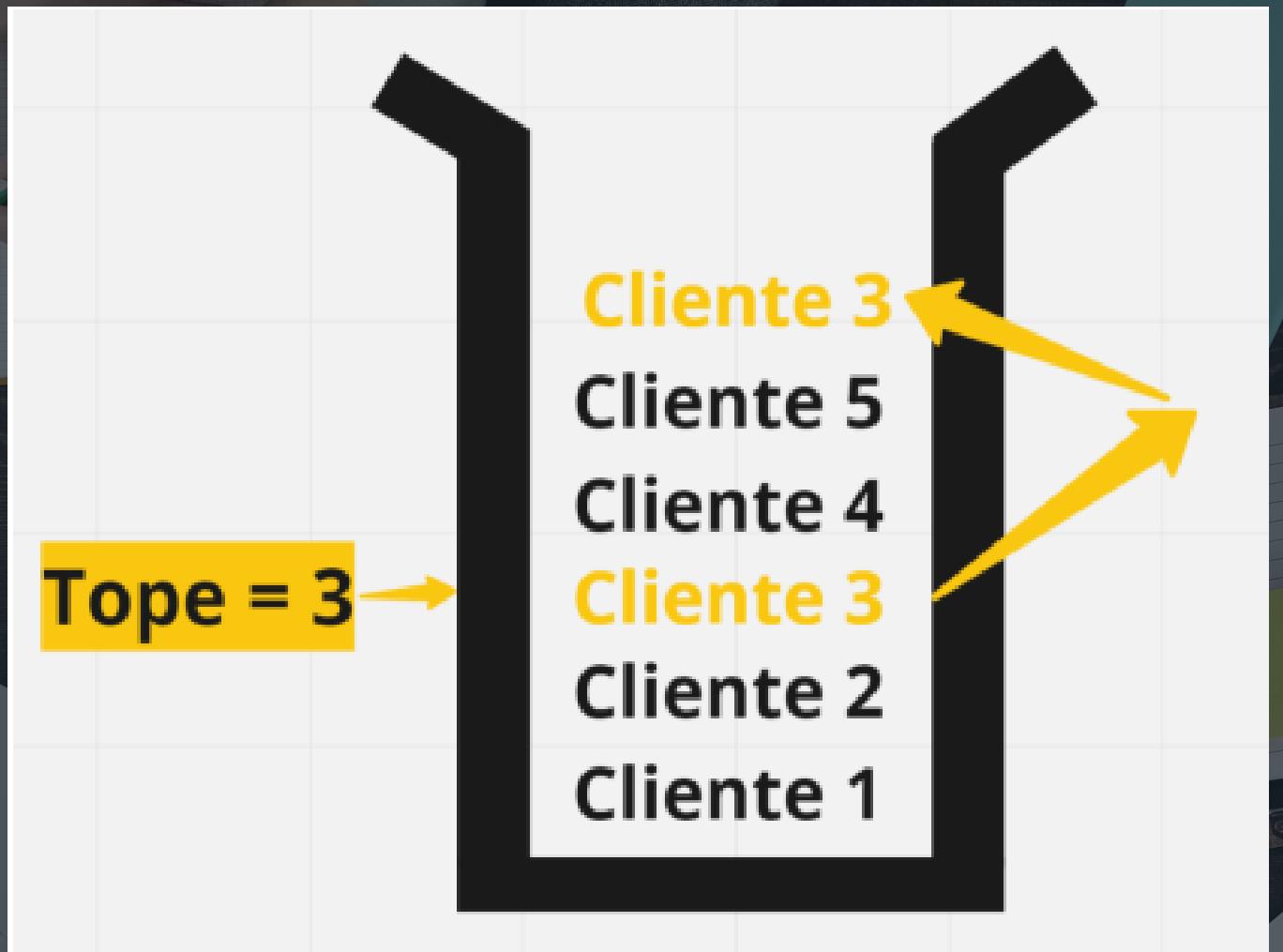
- El método debe ser creado en la clase MAIN como un método estático.
- El método recibe 2 parámetros
  - La Pila de Clientes
  - El valor de la edad.
- Adjuntar los siguientes
  - El código del método que resuelve el problema.
  - Una imagen de la salida de la consola.

## 12.DETERMINAR CUÁNTOS CLIENTES SON MAYORES DE 20 AÑOS.

```
public static void mayoresCiertaEdad(PilaDeClientes pila,int EdadMayor){  
    PilaDeClientes aux = new PilaDeClientes();  
    Cliente item = null;  
    int cont = 0;  
    while (!pila.EsVacio()){  
        item = pila.eliminar();  
        if(item.getEdad()>EdadMayor){  
            cont = cont +1 ;  
        }  
        aux.Agregar(item);  
    }  
    pila.Vaciar(aux);  
    System.out.println("El numero de clientes mayores a "+ EdadMayor +" años es: " + cont);  
}
```

El numero de clientes mayores a 20 años es: 2

## 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.

## 13.MOVER EL K-ÉSIMO ELEMENTO AL FINAL DE LA PILA.

- Adjuntar los siguientes
  - El código del método que resuelve el problema.
  - Una imagen de la salida de la consola.

```
public static void kEsimoPosicion(PilaDeClientes pila,int kesimo){  
    PilaDeClientes aux = new PilaDeClientes();  
    Cliente item = null;  
    Cliente min = null;  
  
    while (!pila.EsVacio()){  
        item = pila.eliminar();  
        if(pila.NroElementos()+1==kesimo){  
            min = item;  
        }else{  
            aux.Agregar(item);  
        }  
    }  
    pila.Vaciar(aux);  
    pila.Agregar(min);  
    System.out.println("\nMOSTRANDO NUEVA PILA DE CLIENTES\n");  
}
```

## 13. MOVER EL K-ÉSIMO ELEMENTO AL FINAL DE LA PILA.

Mostrando la PILA DE CLIENTES

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Edson ivan

Apellidos: Condori Condori

Edad: 21

Direccion: Calle Juan de la Riva #1108, Zona Central, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Sergio Andres

Apellidos: Mendoza Alvarado

Edad: 20

Direccion: Av. Julio Cesar Valdez #1250, Zona Villa Copacabana, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Elena

Apellidos: Lopez Mendoza

Edad: 19

Direccion: Calle México #150, Villa Exaltación, El Alto, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Calle Hermanos Manchego #100, Zona Sur, La Paz, Bolivia

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Calle Hermanos Manchego #100, Zona Sur, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Juan Carlos

Apellidos: Mamani Colque

Edad: 19

Direccion: Av. 6 de Marzo #4720, Villa Dolores, El Alto, Bolivia

Genero: Masculino

## 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.

PILA DE CLIENTES

Cliente 5  
Cliente 4  
Cliente 3  
Cliente 2  
Cliente 1

## 14. CAMBIAR LA DIRECCIÓN DE ALGUNOS CLIENTES DE LA PILA.

- Adjuntar los siguientes
- El código del método que resuelve el problema.
- Una imagen de la salida de la consola.

```
public static void asignaDireccion(PilaDeClientes pila, String nuevaDireccion){  
    PilaDeClientes aux = new PilaDeClientes();  
    Cliente item = null;  
    while (!pila.EsVacio()){  
        item = pila.eliminar();  
        if(item.getGenero().equals("Femenino")){  
            item.setDireccion(nuevaDireccion);  
        }  
        aux.Agregar(item);  
    }  
    pila.Vaciar(aux);  
    System.out.println("\nMOSTRANDO NUEVA PILA DE CLIENTE CON DIRECCION CAMBIADA\n");  
}
```

# 14. CAMBIAR LA DIRECCIÓN DE ALGUNOS CLIENTES DE LA PILA.

MOSTRANDO LA NUEVA PILA DE CLIENTES CON DIRECCION CAMBIADA

Mostrando la PILA DE CLIENTES

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Sergio Andres

Apellidos: Mendoza Alvarado

Edad: 20

Direccion: Av. Julio Cesar Valdez #1250, Zona Villa Copacabana, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Elena

Apellidos: Lopez Mendoza

Edad: 19

Direccion: Av. 16 de Julio No. 1496, El Alto, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Edson ivan

Apellidos: Condori Condori

Edad: 21

Direccion: Calle Juan de la Riva #1108, Zona Central, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Av. 16 de Julio No. 1496, El Alto, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Av. 16 de Julio No. 1496, El Alto, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Juan Carlos

Apellidos: Mamani Colque

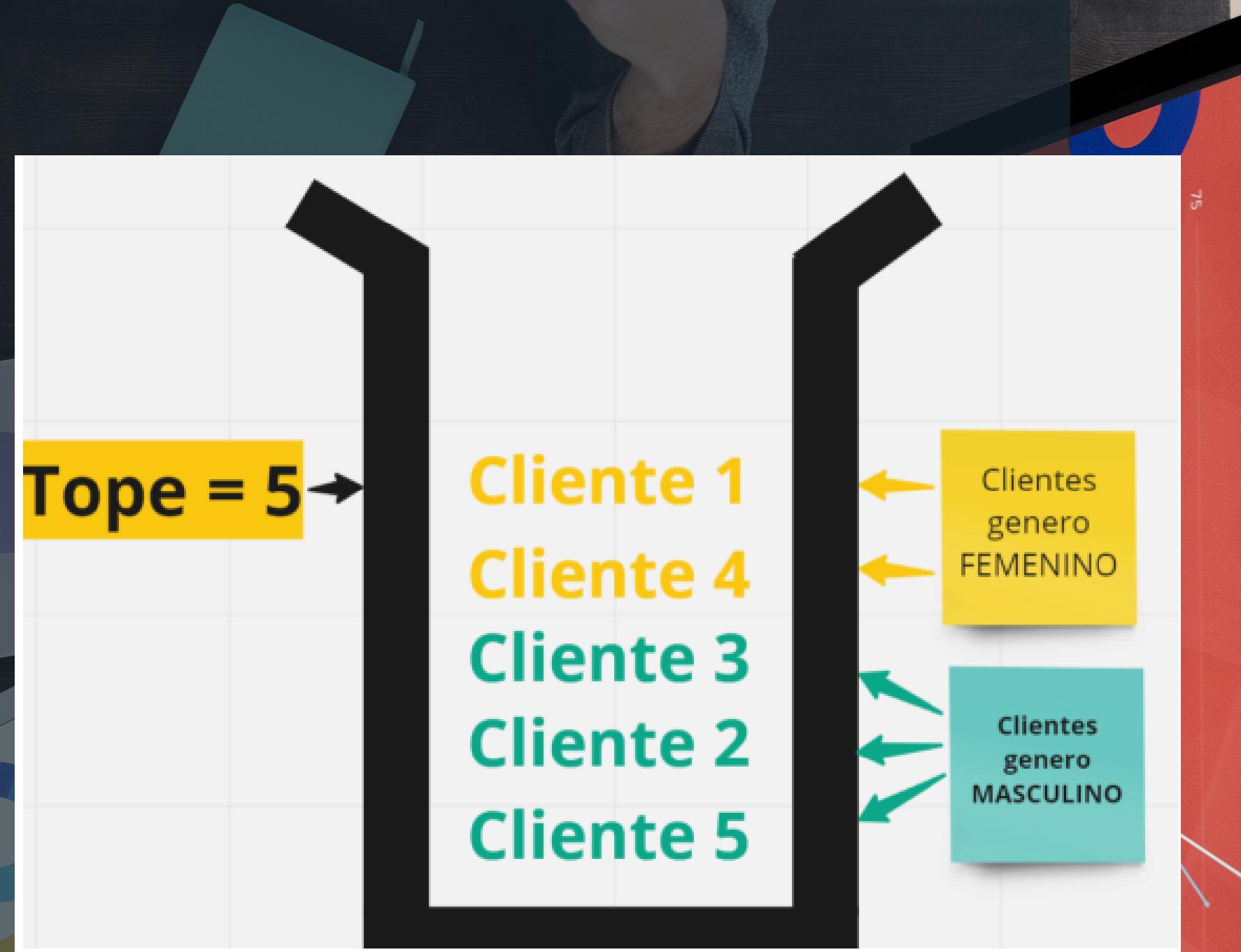
Edad: 19

Direccion: Av. 6 de Marzo #4720, Villa Dolores, El Alto, Bolivia

Genero: Masculino

## 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.



## 15.MOVER ÍTEMS DE LA PILA.

- Adjuntar los siguientes
  - El código del método que resuelve el problema.
  - Una imagen de la salida de la consola

```
public static void ReordenarPila(PilaDeClientes pila){  
    PilaDeClientes aux = new PilaDeClientes();  
    PilaDeClientes aux2 = new PilaDeClientes();  
    Cliente item = null;  
    while (!pila.EsVacio()){  
        item = pila.eliminar();  
        if(item.getGenero().equals("Masculino")){  
            aux.Agregar(item);  
        }else {  
            aux2.Agregar(item);  
        }  
    }  
    pila.Vaciar(aux);  
    pila.Vaciar(aux2);  
    System.out.println("\nMOSTRANDO NUEVA PILA DE CLIENTES\n");  
}
```

# 15. MOVER ÍTEMS DE LA PILA.

//MOSTRANDO LA NUEVA PILA DE CLIENTES//

Mostrando la PILA DE CLIENTES

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Elena

Apellidos: Lopez Mendoza

Edad: 19

Direccion: Calle México #150, Villa Exaltación, El Alto, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Maria

Apellidos: Mancora Lopez

Edad: 25

Direccion: Calle Hermanos Manchego #100, Zona Sur, La Paz, Bolivia

Genero: Femenino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Sergio Andres

Apellidos: Mendoza Alvarado

Edad: 20

Direccion: Av. Julio Cesar Valdez #1250, Zona Villa Copacabana, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Edson ivan

Apellidos: Condori Condori

Edad: 21

Direccion: Calle Juan de la Riva #1108, Zona Central, La Paz, Bolivia

Genero: Masculino

MOSTRANDO DATOS DEL CLIENTE:

Nombre: Juan Carlos

Apellidos: Mamani Colque

Edad: 19

Direccion: Av. 6 de Marzo #4720, Villa Dolores, El Alto, Bolivia

Genero: Masculino

