

PRACTICA



Estudiante:

Univ. Iris Michelle Velasco Aruquipa

Asignatura: Estructura de Datos

Docente

Lic. William Barra Paredes



[**https://github.com/mishelvelasco/estructura-de-datos-**](https://github.com/mishelvelasco/estructura-de-datos-)

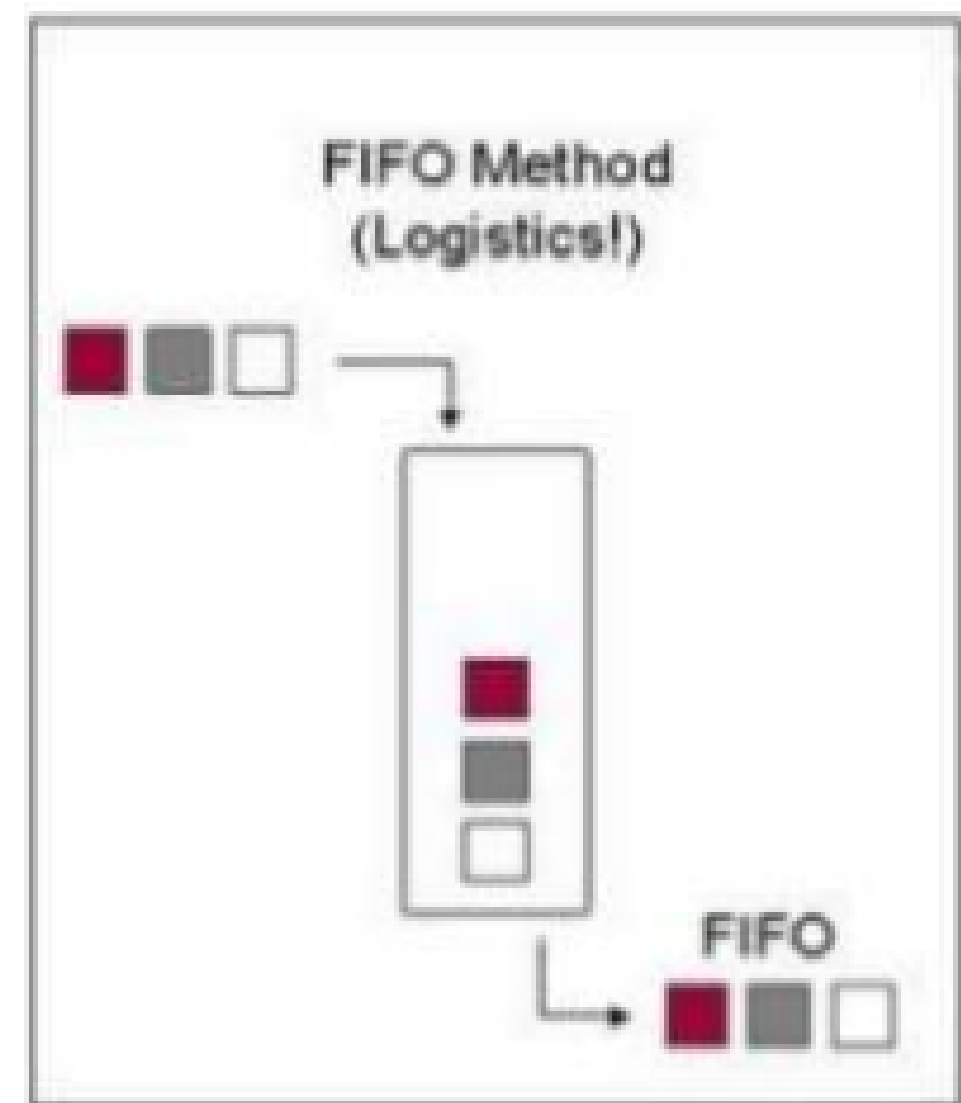


¿A QUE SE REFIERE CUANDO SE HABLA DE ESTRUCTURA DE DATOS?

Una “estructura de datos” es una colección de valores , la relación que existe entre estos valores y las operaciones que podemos hacer sobre ellos; en pocas palabras se refiere a cómo los datos están organizados y cómo se pueden administrar. Una estructura de datos describe el formato en que los valores van a ser almacenados, cómo van a ser accedidos y modificados, pudiendo así existir una gran cantidad de estructuras de datos.

¿QUE SIGNIFICA FIFO?

Significa: «Primero en entrar, primero en salir» es un concepto utilizado en estructuras de datos y teoría de colas. Guarda analogía con las personas que esperan en una cola y van siendo atendidas en el orden en que llegaron, es decir, que "la primera persona que entra es la primera persona que sale".




¿MUESTRA LA DIFERENCIA ENTRE LIFO Y FIFO?





FIFO

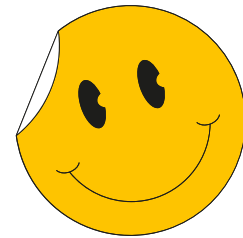
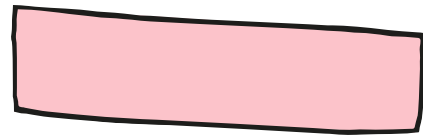
Primero en
entrar, primero
en salir



LIFO

Último En
Entrar, Primero
En Salir

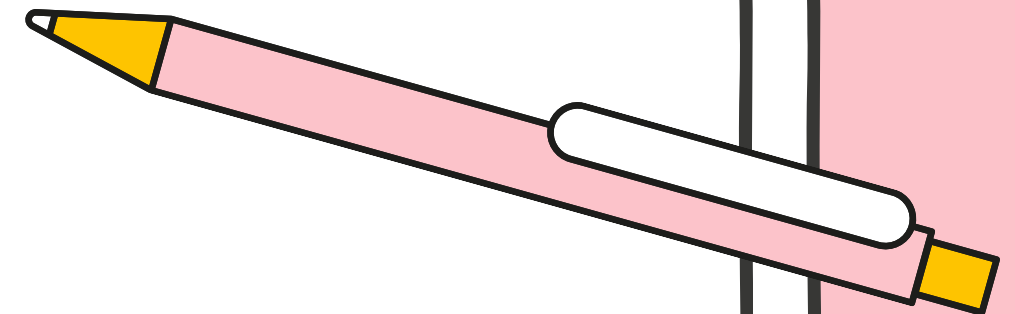


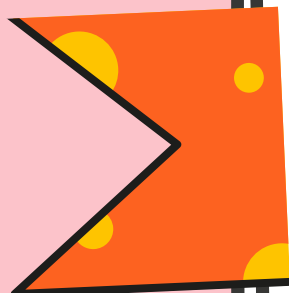
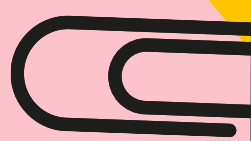



¿QUÉ ES UNA COLA?



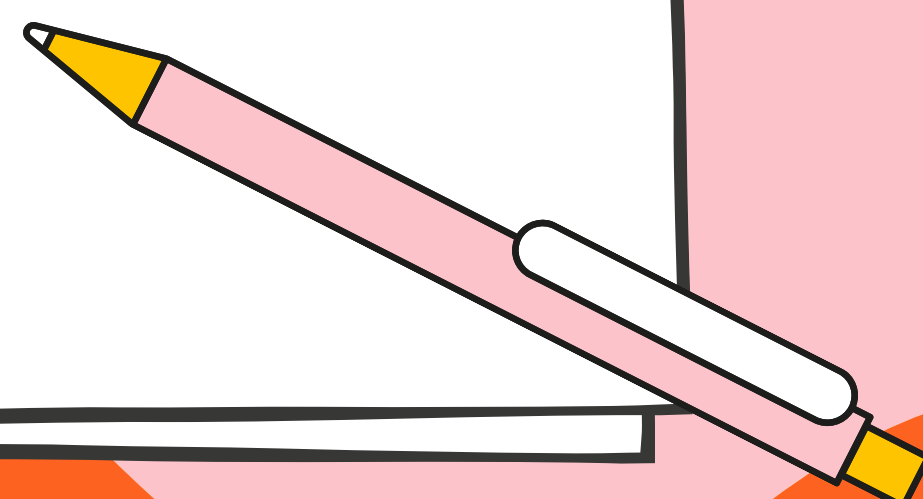
una cola es una lista lineal en la que los elementos solo pueden ser añadidos por un extremo y eliminados por el otro.





¿QUÉ ES QUEUE EN JAVA, UNA QUEUE SERA LO MISMO QUE UNA COLA?

Un objeto de la clase Queue es una cola. Permite almacenar objetos y luego recuperarlos en el orden en el cual se insertaron.
por lo cual una Queue es lo mismo que una cola ya que siguen la misma filosofia de FIFO



¿QUÉ ES INÍ O REAR EN UNA COLA?



INE

Frontal de la
cola



REAR

extremo,
final





¿QUÉ ES FÍN O FRONT EN UNA COLA?

en una cola los datos entran por un extremo llamado final y se insertan por el otro extremo llamado frente (front).

¿A QUE SE REFIERE EL MÉTODOS ESVACÍA() EN UNA COLA?

El método esVacia()
es cuando la cola
no tiene elementos
almacenados

```
public boolean esVacia() {  
    if (this.ini == 0 & this.fin == 0) {  
        return true;  
    } else {  
        return false;  
    }  
}
```

¿A QUE SE REFIERE EL MÉTODOS ESLLENA() EN UNA COLA?

El método esLlena() es cuando el tope de la cola es igual a fin

```
public boolean esLlena() {  
    if (this.fin == this.max) {  
        return true;  
    } else {  
        return false;  
    }  
}
```



¿QUÉ SON LOS MÉTODOS ESTÁTICOS EN JAVA?

Un método estático es un método que tiene sentido invocarla sin crear previamente ningún objeto.

¿A TRAVÉS DE UN GRÁFICO, MUESTRE LOS MÉTODOS MÍNIMOS QUE DEBERÍA DE TENER UNA COLA?

Cliente		
m	Cliente (String , String , int , String , String , String)	
f	Genero	String
f	pais	String
f	Apellidos	String
f	Edad	int
f	tipo	String
f	Nombres	String
m	setPais (String)	void
m	getTipo ()	String
m	getGenero ()	String
m	getEdad ()	int
m	mostrarCliente ()	void
m	setTipo (String)	void
m	setEdad (int)	void
m	getApellidos ()	String
m	getPais ()	String
m	setApellidos (String)	void
m	setDireccion (String)	void
m	getDireccion ()	String
m	setGenero (String)	void
m	setNombres (String)	void
m	getNombres ()	String

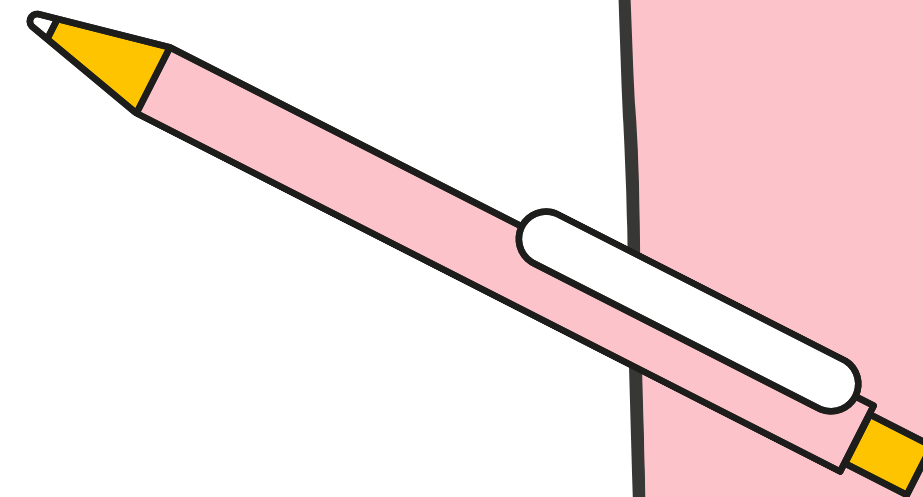
CREAR LAS CLASES NECESARIAS PARA LA COLA DE CLIENTES.

```
package procesual_hito4;
public class Cliente {

    3 usages
    private String Nombres;
    3 usages
    private String Apellidos;
    3 usages
    private int Edad;
    5 usages
    private String pais;
    3 usages
    private String Genero;
    3 usages
    private String tipo;

    10 usages
    public Cliente(String Nombres, String Apellidos, int Edad, String pais, String Genero,String tipo) {
        this.Nombres = Nombres;
        this.Apellidos = Apellidos;
        this.Edad = Edad;
        this.pais = pais;
        this.Genero = Genero;
        this.tipo=tipo;
    }
}
```

CLASE CLIENTE



CREAR LAS CLASES NECESARIAS PARA LA COLA DE CLIENTES.

```
public class ColaCliente {
    4 usages
    private int max;
    8 usages
    private int fin;
    7 usages
    private int ini;
    3 usages
    private Cliente [] clientes;

    8 usages
    public ColaCliente (int max) {
        this.max = max;
        this.clientes = new Cliente[this.max + 1];
        this.ini = 0;
        this.fin = 0;
    }

    6 usages
    public boolean esVacia() {
        if (this.ini == 0 & this.fin == 0) {
            return true;
        } else {
            return false;
        }
    }
}
```

```
    public boolean esLlena() {
        if (this.fin == this.max) {
            return true;
        } else {
            return false;
        }
    }

    2 usages
    public int nroElem() {
        return fin - ini;
    }

    21 usages
    public void adicionar(Cliente nuevoCliente) {
        if (!esLlena()) {
            fin++;
            clientes[fin] = nuevoCliente;
        } else {
            System.out.println("Cola de numeros llena");
        }
    }

    6 usages
    public Cliente eliminar() {
        Cliente elementoEliminado = null;
        if (!esVacia()) {
            this.ini++;
        }
    }
}
```

```
    public void mostrar () {
        Cliente elem = null;
        if (esVacia())
            System.out.println("Pila Vacía");
        else {
            System.out.println("\nDatos de la Pila de clientes");
            ColaCliente aux = new ColaCliente(this.max);
            while (!esVacia()) {
                elem = this.eliminar();
                aux.adicionar (elem);
                elem.mostrarCliente();
            }
            vaciar(aux);
        }
    }

    6 usages
    public void vaciar (ColaCliente pila) {
        while (!pila.esVacia())
            adicionar(pila.eliminar());
    }
}
```


CREAR LAS CLASES NECESARIAS PARA LA COLA DE CLIENTES.

```
public class Main {  
    public static void main(String[] args) {  
        Cliente cli1 = new Cliente( Nombres: "iris", Apellidos: "velasco", Edad: 20, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli2 = new Cliente( Nombres: "jorge", Apellidos: "mamani", Edad: 30, pais: "argentina", Genero: "masculino", tipo: "sirv");  
        Cliente cli3 = new Cliente( Nombres: "ilia", Apellidos: "laura", Edad: 65, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli4 = new Cliente( Nombres: "marcos", Apellidos: "calle", Edad: 30, pais: "bolivia", Genero: "masculino", tipo: "sirver");  
        Cliente cli5 = new Cliente( Nombres: "leila", Apellidos: "velasquez", Edad: 40, pais: "bolivia", Genero: "femenino", tipo: "gol");  
        ColaCliente cola = new ColaCliente( max: 100);  
        cola.adicionar(cli1);  
        cola.adicionar(cli2);  
        cola.adicionar(cli3);  
        cola.adicionar(cli4);  
        cola.adicionar(cli5);  
  
        Cliente cli6 = new Cliente( Nombres: "lulu", Apellidos: "velasco", Edad: 20, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli7 = new Cliente( Nombres: "saul", Apellidos: "mamani", Edad: 30, pais: "argentina", Genero: "masculino", tipo: "sirve");  
        Cliente cli8 = new Cliente( Nombres: "patito", Apellidos: "laura", Edad: 65, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli9 = new Cliente( Nombres: "patricio", Apellidos: "calle", Edad: 30, pais: "bolivia", Genero: "masculino", tipo: "sirv");  
        Cliente cli10 = new Cliente( Nombres: "julio", Apellidos: "velasquez", Edad: 40, pais: "bolivia", Genero: "femenino", tipo: "gol");  
        ColaCliente cola2 = new ColaCliente( max: 100);  
        cola.adicionar(cli6);  
        cola.adicionar(cli7);  
        cola.adicionar(cli8);  
    }  
}
```

12. INICIALIZAR LA COLA DE CLIENTES.

CREAR UNA COLA CON 5 CLIENTES.

- EN LA CLASE MAIN DEBERÁN ESTAR LOS 5 CLIENTES.
- MOSTRAR TODOS LOS DATOS DE LA COLA DE CLIENTES

```
class Main {  
    public static void main(String[] args) {  
        Cliente cli1 = new Cliente( Nombres: "iris", Apellidos: "velasco", Edad: 20, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli2 = new Cliente( Nombres: "jorge", Apellidos: "mamani", Edad: 30, pais: "argentina", Genero: "masculino", tipo: "sirver");  
        Cliente cli3 = new Cliente( Nombres: "ilia", Apellidos: "laura", Edad: 65, pais: "bolivia", Genero: "femenino", tipo: "vip");  
        Cliente cli4 = new Cliente( Nombres: "marcos", Apellidos: "calle", Edad: 30, pais: "bolivia", Genero: "masculino", tipo: "sirver");  
        Cliente cli5 = new Cliente( Nombres: "leila", Apellidos: "velasquez", Edad: 40, pais: "bolivia", Genero: "femenino", tipo: "gold");  
        ColaCliente cola = new ColaCliente( max: 100);  
        cola.adicionar(cli1);  
        cola.adicionar(cli2);  
        cola.adicionar(cli3);  
        cola.adicionar(cli4);  
        cola.adicionar(cli5);  
    }  
}
```

Datos de la Pila de clientes

Mostrando datos del jugador

Nombre: iris
Apellidos: velasco
Edad: 20
Direccion: bolivia
Genero: femenino
tipo: vip

Mostrando datos del jugador

Nombre: jorge
Apellidos: mamani
Edad: 30
Direccion: argentina
Genero: masculino
tipo: sirver

13.PROMOCIÓN PARA USUARIOS DE BOLÍVIA.

A TODOS LOS CLIENTES QUE SEAN DE NACIONALIDAD BOLIVIANA Y ADEMÁS EL TIPO DE CLIENTE GOLD, CONVERTIR A ESTOS CLIENTES EN VIP

```
static void asignaVip(ColaCliente cola, String NuevoTipo) {  
    ColaCliente aux = new ColaCliente( max: 10);  
    Cliente valor = null;  
    while (!cola.esVacia()) {  
        valor = cola.eliminar();  
        if (valor.getPais() == "bolivia" && valor.getTipo() == "gold") {  
            valor.setTipo(NuevoTipo);  
            aux.adicionar(valor);  
        } else {  
            aux.adicionar(valor);  
        }  
    }  
    cola.vaciar(aux);  
    cola.adicionar(valor);  
    cola.mostrar();  
}
```

```
Mostrando datos del jugador  
Nombre: lulu  
Apellidos: velasco  
Edad: 20  
Direccion: bolivia  
Genero: femenino  
tipo: vip
```

14.MOVIENDO CLIENTES EN LA COLA

MOVER AL INICIO TODOS LOS CLIENTES MAYORES A 60 AÑOS.

```
//Determinar cuantos clientes son mayores a 60 años
public static void MoverClientesMayores ( ColaCliente cola , int edadMayor ) {
    ColaCliente aux = new ColaCliente ( max: 100 );
    ColaCliente aux2 = new ColaCliente ( max: 100 );
    Cliente Valorextraido = null ;
    while (! cola.esVacia ()) {
        Valorextraido = cola.eliminar ();
        if ( Valorextraido.getEdad () >= edadMayor ) {
            aux2.adicionar (Valorextraido);
        }
        aux.adicionar (Valorextraido);
    }
    cola.vaciar (aux2);
    cola.vaciar (aux);
    cola.mostrar ();
}
```

```
Mostrando datos del jugador
Nombre: ilia
Apellidos: laura
Edad: 65
Direccion: bolivia
Genero: femenino
tipo: vip
```

```
Mostrando datos del jugador
Nombre: patito
Apellidos: laura
Edad: 65
Direccion: bolivia
Genero: femenino
tipo: vip
```

15.MOVIENDO CLIENTES ENTRE 2 COLAS.

```
public static void CambiarColaclientes ( ColaCliente colaA , ColaCliente colaB , String Nombre )
{
    int nroElemColaA = colaA.nroElem ();
    int nroElemColaB = colaB.nroElem ();
    ColaCliente aux = new ColaCliente ( max: 100 );
    ColaCliente aux2 = new ColaCliente ( max: 100 );
    Cliente valorEliminado = null ;
    for ( int i = 1 ; i <= nroElemColaA ; i ++ ) {
        valorEliminado = colaA.eliminar ();
        if ( valorEliminado.getNombres().equals(Nombre)) {
            colaB.adicionar ( valorEliminado );
        } else {
            colaA.adicionar ( valorEliminado );
        }
    }
    for ( int i = 1 ; i <= nroElemColaB ; i ++ ) {
        valorEliminado = colaB.eliminar ();
        if ( valorEliminado.getNombres () .equals ( Nombre )) {
            colaB.adicionar ( valorEliminado );
        } else {
            aux2.adicionar ( valorEliminado );
        }
    }
    colaA.vaciar ( aux );
    colaB.vaciar ( aux2 );
    colaA.mostrar ();
    colaB.mostrar ();
}
```

Activar Windows



**MUCHAS
GRACIAS**