

**UNIVERSIDAD PRIVADA FRANZ
TAMAYO
“def-hito-4”**

Estudiante: Univ. Kevin Oscar Mamani
Laura

Asignatura: BASE DE DATOS II

Carrera: INGENIERÍA DE SISTEMAS

Paralelo: BDA (1)

Docente: Lic. William Barra Paredes

fecha: 10/4/2022

GITHUB: <https://github.com/oscar999ml/B>

[aseDeDatos1](#)

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

son aquellas que nos permiten, como desarrolladores, organizar la información de manera eficiente, y en definitiva diseñar la solución correcta para un determinado problema.

2. ¿Que significa FIFO?

FIFO: Primero En Entrar, Primero en Salir

3. ¿Muestra la diferencia entre LIFO y FIFO?

El el método FIFO las primeras mercancías en entrar serán las primeras en salir, mientras que en el método LIFO, las ultimas mercancías en entrar serán las primeras en salir.

4. ¿Qué es una COLA?

Una cola (queue) es una estructura de datos donde la inserción y eliminación de los elementos se hace según el principio FIFO (First In First Out). Esto es, el primer elemento en salir es el primero en entrar en la cola. La cola se representa como una lista enlazada (también llamada lista encadenada).

5. ¿Qué es QUEUE en JAVA, una QUEUE será lo mismo que una COLA?

Una Cola o Queue es una estructura de datos que sigue la Filosofía FIFO del ingles First In – First Out que en español seria “Primero en entrar primero en salir”. Esto quiere decir que el elemento que entre primero a la Cola sera el primero que salga y el último que entre sera el último en salir.

6. ¿Qué es INI o REAR en una COLA?

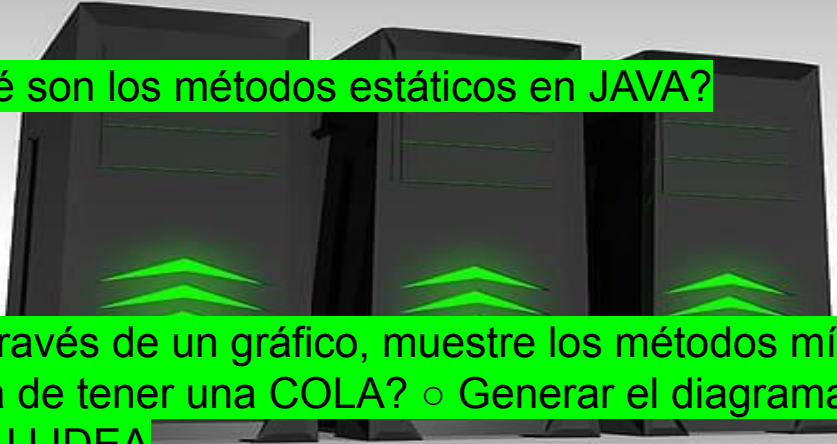
¿Qué es FIN o FRONT en una COLA?

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

8. ¿A que se refiere los métodos esVacia() y esLLena() en una COLA? ○ Adjunte los métodos

9. ¿Qué son los métodos estáticos en JAVA?

10. ¿A través de un gráfico, muestre los métodos mínimos que debería de tener una COLA? ○ Generar el diagrama con el editor INTELLIJ IDEA





continuaremos con el video xfabor


```
public class Main {  
    public static void main(String[] args) {  
        Cliente cl1 = new Cliente( Nombres: "kevin", Apellidos: "mamani", Edad: 22, Pais: "Bolivia", Genero: "Masculino", tipo: "SILVER");  
        Cliente cl2 = new Cliente( Nombres: "saul", Apellidos: "laura", Edad: 17, Pais: "Bolivia", Genero: "Masculino", tipo: "GOLD");  
        Cliente cl3 = new Cliente( Nombres: "justina", Apellidos: "mamani", Edad: 37, Pais: "Peru", Genero: "Femenino", tipo: "VIP");  
        Cliente cl4 = new Cliente( Nombres: "willy", Apellidos: "fabian", Edad: 35, Pais: "Ecuador", Genero: "Masculino", tipo: "GOLD");  
        Cliente cl5 = new Cliente( Nombres: "miguel", Apellidos: "herera", Edad: 62, Pais: "Brazil", Genero: "Masculino", tipo: "VIP");  
  
        ColaClientes Cola = new ColaClientes( max: 100);  
        Cola.adicionar(cl1);  
        Cola.adicionar(cl2);  
        Cola.adicionar(cl3);  
        Cola.adicionar(cl4);  
        Cola.adicionar(cl5);  
  
        Cliente cl6 = new Cliente( Nombres: "hanabi", Apellidos: "fokito", Edad: 23, Pais: "China", Genero: "Femenino", tipo: "SILVER");  
        Cliente cl7 = new Cliente( Nombres: "baxia", Apellidos: "lopez", Edad: 36, Pais: "Bolivia", Genero: "Masculino", tipo: "GOLD");  
        Cliente cl8 = new Cliente( Nombres: "jhonson", Apellidos: "fugimori", Edad: 22, Pais: "Peru", Genero: "Masculino", tipo: "VIP");  
        Cliente cl9 = new Cliente( Nombres: "martis", Apellidos: "rodriguez", Edad: 18, Pais: "mexico", Genero: "Masculino", tipo: "GOLD");  
        Cliente cl10 = new Cliente( Nombres: "rafaela", Apellidos: "marick", Edad: 37, Pais: "francia", Genero: "Femenino", tipo: "VIP");  
  
        ColaClientes Cola2 = new ColaClientes( max: 100);  
        Cola2.adicionar(cl6);  
        Cola2.adicionar(cl7);  
        Cola2.adicionar(cl8);  
        Cola2.adicionar(cl9);  
        Cola2.adicionar(cl10);  
    }  
}
```

```
//      camVIPbolivia(Cola, "Bolivia", "GOLD","VIP");  
public static void camVIPbolivia(ColaClientes cola, String p, String t,String nt) {  
    ColaClientes aux = new ColaClientes( max: 10);  
    Cliente Valorextraido = null;  
    while (!cola.esVacia()) {  
        Valorextraido = cola.eliminar();  
        if (Valorextraido.getPais().equals(p) && Valorextraido.getTipo().equals(t)) {  
            Valorextraido.setTipo(nt);  
        }  
        aux.adicionar(Valorextraido);  
    }  
    cola.vaciar(aux);  
    cola.mostrar();  
}
```

```
// 60
public static void mayoresalInicio(ColaClientes cola, int edad60) {
    ColaClientes aux = new ColaClientes( max: 100);
    ColaClientes aux2 = new ColaClientes( max: 100);
    Cliente Valorextraido = null;
    while (!cola.esVacia()) {
        Valorextraido = cola.eliminar();
        if (Valorextraido.getEdad() >= edad60) {
            aux2.adicionar(Valorextraido);
        }
        if(Valorextraido.getEdad()<= edad60){
            aux.adicionar(Valorextraido);
        }
    }
    cola.vaciar(aux2);
    cola.vaciar(aux);
    cola.mostrar();
}
```

```

        //CambiarSaul(Cola,Cola2, "saul");
1 usage
public static void CambiarSaul(ColaClientes colaA, ColaClientes colaB,String Nombre) {
    int nroElemColaA = colaA.nroElem();
    int nroElemColaB = colaB.nroElem();
    ColaClientes aux = new ColaClientes( max: 100);
    ColaClientes aux2 = new ColaClientes( max: 100);

    Cliente valorEliminado = null;
    for (int i = 1; i <= nroElemColaA; i++) {
        valorEliminado = colaA.eliminar();
        if (valorEliminado.getNombres().equals(Nombre)) {
            colaB.adicionar(valorEliminado);
        }
        if (!valorEliminado.getNombres().equals(Nombre)){
            colaA.adicionar(valorEliminado);
        }
    }
    for (int i = 1; i <= nroElemColaB; i++) {
        valorEliminado = colaB.eliminar();
        if (valorEliminado.getNombres().equals(Nombre)) {
            colaB.adicionar(valorEliminado);
        }
        if (!valorEliminado.getNombres().equals(Nombre)) {
            aux2.adicionar(valorEliminado);
        }
    }
    colaA.vaciar(aux);
}

```

```

        aux2.adicionar(valorEliminado);
    }
}
colaA.vaciar(aux);
colaB.vaciar(aux2);
colaA.mostrar();
colaB.mostrar();
}
}

```