PROCESUAL HITO 4 **ESTRUCTURA DE DATOS** JOEL REYNALDO CONDORI TUMIRI

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

 las estructuras 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. ¿QUÉ SIGNIFICA FIFO?

FIFO: Primero En Entrar, Primero en Salir

Del inglés: "First In, First Out" La definición del FIFO es simple: el lote de stock que primera entra, es el que primero sale. Es el método más utilizado para los productos perecederos como los alimentos, que tienen fecha de caducidad.

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 es una estructura de datos que almacena elementos en una lista y
permite acceder a los datos por uno de los dos extremos de la lista. Un
elemento se inserta en la cola (parte final) de la lista y se suprime o elimina por
la frente (parte inicial, cabeza) de la lista.

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".

Queue es lo mismo que Cola.

6. ¿QUÉ ES INI O REAR EN UNA COLA?

Un INI es una variable utilizada para saber la posición real del primer elemento.

7. ¿Qué es INI o REAR en una COLA?

Un fin es la variable que marca la siguiente posición en entrar a la cola

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

El método es Vacia() es cuando la cola no contiene elementos almacenados, de forma que para realizar cambios, primero habrá que almacenar información en la cola.

El método esLlena() es cuando FIN es igual a INI, impidiendo poder seguir llenando la cola.

9. ¿QUÉ SON LOS MÉTODOS ESTÁTICOS EN JAVA?

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

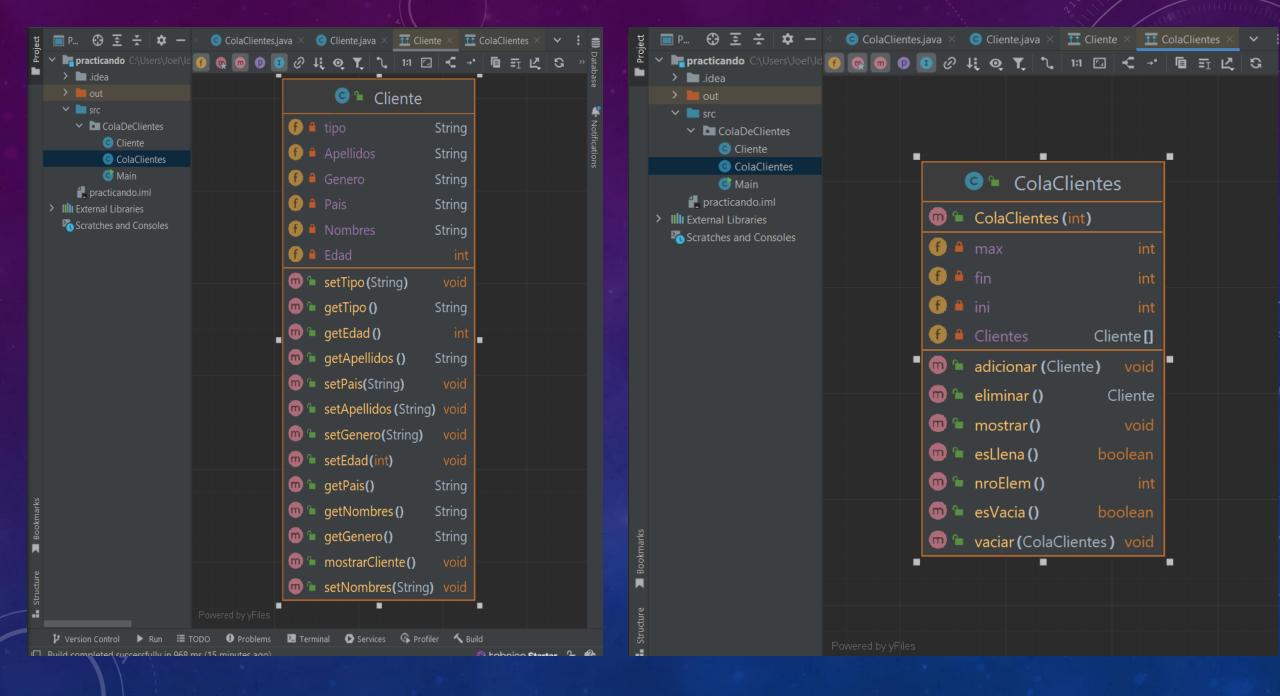
10. A TRAVÉS DE UN GRÁFICO, MUESTRE LOS MÉTODOS MÍNIMOS QUE DEBERÍA DE TENER UNA COLA.

```
public class Cliente {
   private String Nombres;
   private String Apellidos;
    private String Pais;
   private String Genero;
    private String tipo;
    public Cliente(String Nombres, String Apellidos, int Edad, String Pais, String Genero, String tipo) {
       this.Nombres = Nombres:
       this.Genero = Genero;
    public String getNombres() { return Nombres; }
    public String getApellidos() { return Apellidos; }
    public int getEdad() { return Edad; }
    public String getPais() { return Pais; }
    public String getGenero() { return Genero; }
```

```
public int getEdad() { return Edad; }
public String getPais() { return Pais; }
public String getGenero() { return Genero; }
public String getTipo() { return tipo; }
public void setNombres(String nombres) { Nombres = nombres; }
public void setApellidos(String apellidos) { Apellidos = apellidos; }
public void setPais(String pais) { Pais = pais; }
public void setGenero(String genero) { Genero = genero; }
public void setTipo(String tipo) { this.tipo = tipo; }
public void mostrarCliente() {
    System.out.println("Nombre: " + this.getNombres());
    System.out.println("Apellidos: " + this.getApellidos());
    System.out.println("Edad: " + this.getEdad());
    System.out.println("Pais: " + this.getPais());
    System.out.println("Genero: " + this.getGenero());
    System.out.println("\n");
```

```
public class ColaClientes {
    private int max;
   private Cliente[] Clientes;
    public ColaClientes(int max) {
        this.max = max;
        this.Clientes = new Cliente[this.max + 1];
    public boolean esVacia() {
        if (this.ini == 0 & this.fin == 0) {
            return true;
            return false;
    public boolean esLlena() {
        if (this.fin == this.max) {
        } else {
            return false;
    public int nroElem() { return fin - ini; }
```

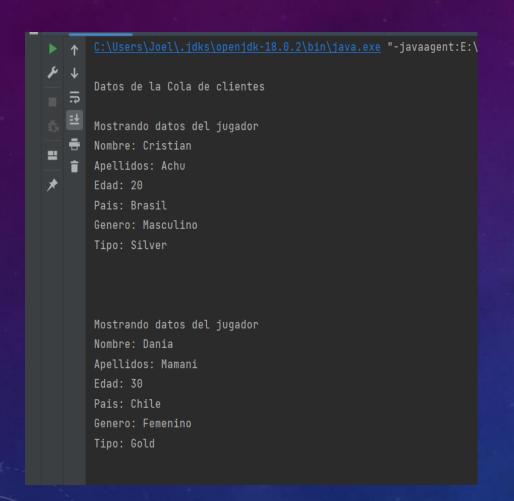
```
public Cliente eliminar() {
    Cliente elementoEliminado = null;
    if (!esVacia()) {
        elementoEliminado = this.Clientes[ini];
        if (ini == fin) {
        return elementoEliminado;
        System.out.println("Cola de numeros vacía");
   return elementoEliminado;
public void mostrar () {
    Cliente elem = null;
    if (esVacia())
        System.out.println("Cola Vacia");
        System.out.println("\nDatos de la Cola de clientes");
        ColaClientes aux = new ColaClientes(fin);
        while (!esVacia()) {
            elem = this.eliminar();
            aux.adicionar (elem);
            elem.mostrarCliente();
        vaciar(aux);
public void vaciar (ColaClientes cola) {
    while (!cola.esVacia())
        adicionar(cola.eliminar());
```

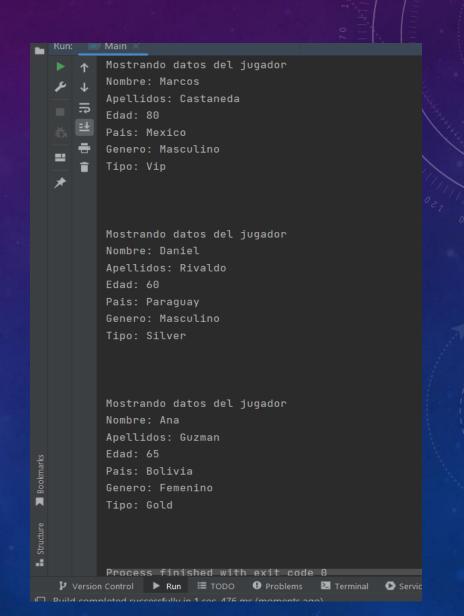


11. CREAR LAS CLASES NECESARIAS PARA LA PILA DE CLIENTES

```
ColaClientes.java ×
Main.java
                               Cliente.java
       package ColaDeClientes:
           public static void main(String[] args) {
               Cliente c1 = new Cliente( Nombres: "Cristian", Apellidos: "Achu", Edad: 20, Pais: "Brasil", Genero: "Masculino", tipo: "Silver");
               Cliente c2 = new Cliente( Nombres: "Dania", Apellidos: "Mamani", Edad: 30, Pais: "Chile", Genero: "Femenino", tipo: "Gold");
               Cliente c3 = new Cliente( Nombres: "Marcos", Apellidos: "Castaneda", Edad: 80, Pais: "Mexico", Genero: "Masculino", tipo: "Vip");
               Cliente c4 = new Cliente( Nombres: "Daniel", Apellidos: "Rivaldo", Edad: 60, Pais: "Paraquay", Genero: "Masculino", tipo: "Silver");
               Cliente c5 = new Cliente( Nombres: "Ana", Apellidos: "Guzman", Edad: 65, Pais: "Bolivia", Genero: "Femenino", tipo: "Gold");
               ColaClientes colaCli = new ColaClientes( max: 100);
               colaCli.adicionar(c1);
               colaCli.adicionar(c2):
               colaCli.adicionar(c3);
               colaCli.adicionar(c4):
               colaCli.adicionar(c5);
```

12.INICIALIZAR LA COLA DE CLIENTES





13.PROMOCION PARA USUARIOS DE BOLIVIA

```
✓ ColaDeClientes

         ColaClientes
    practicando.iml
> III External Libraries
  Scratches and Consoles
                                        public static void cambiarTipo(ColaClientes Cola, String Pais, String tipo, String NewTipo) {
                                            ColaClientes aux = new ColaClientes( max: 10);
                                            Cliente Valorextraido = null;
                                                Valorextraido = Cola.eliminar();
                                                if (Valorextraido.getPais().equals(Pais) && Valorextraido.getTipo().equals(tipo)) {
                                                    Valorextraido.setTipo(NewTipo);
                                                     aux.adicionar(Valorextraido);
                                                aux.adicionar(Valorextraido);
                                            Cola.vaciar(aux);
                                            Cola.mostrar();
```

```
Cliente c5 = new Cliente( Nombres: "Ana", Apellidos: "Guzman", Edad: 65, Pais: "Bolivia", Genero: "Femenino", tipo: "Gold");
```

```
Mostrando datos del jugador
Nombre: Ana
Apellidos: Guzman
Edad: 65
Pais: Bolivia
Genero: Femenino
Tipo: Vip

Process finished with exit code 0
```

14.MOVIENDO CLIENTES EN LA COLA

```
Cliente
         ColaClientes
         @ Main
    # practicando.iml
> IIII External Libraries
  Scratches and Consoles
                                       public static void MoverClientesMayores(ColaClientes cola, int edadMayor) {
                                           ColaClientes aux = new ColaClientes( max: 100);
                                           ColaClientes aux2 = new ColaClientes( 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();
```

