

Facultad: Ingeniería en Sistemas

Universidad: Da Vinci

Curso: Estructura De Datos

Tema: Diario De Un Aventurero

Nombre: Ezry Adilia Maribeth Leal Franco

Carnet: 202200202

## **“DIARIO DE UN AVENTURERO”**

### **INTRODUCCIÓN**

Este proyecto desarrolla la aplicación de consola que permitirá al usuario gestionar las misiones en un diario del aventurero. Cada misión deberá tener un nombre, una confirmación de completada o no y una fecha. El trabajo de la consola será registrar las misiones, eliminarlas, mostrarlas en lista, buscarlas y editarlas.

### **DESCRIPCION DEL PROBLEMAS Y SOLUCION PROPUESTA**

#### ***1. PROBLEMA:***

Se requiere implementar en la consola como el usuario gestiona un diario de misiones de un aventurero. Cada una de las misiones deberá contar con un nombre, un estado y una fecha. Esta consola permitirá registrar, eliminar, mostrar, buscar y editar las misiones.

#### ***2. SOLUCION***

Se utiliza una pila, esta almacenará las misiones, esta permitirá añadir y quitar elementos de manera ordenada implementando el LIFO, se implementa la clase pila y la clase diarioaventurero que contendrá la lógica permitiendo gestionar las misiones. Así como se proporcionará un interfaz de usuario simple.

## **CODIGO FUENTE**

```
import java.util.LinkedList;
```

```
import java.util.Scanner;
```

```
class Misión {
```

```
    private String nombre;
```

```
    private boolean completada;
```

```
    private String fecha;
```

```
    public Misión(String nombre, boolean completada, String fecha) {
```

```
        this.nombre = nombre;
```

```
        this.completada = completada;
```

```
        this.fecha = fecha;
```

```
    }
```

```
    public String getNombre() {
```

```
        return nombre;
```

```
}
```

```
public boolean isCompletada() {
```

```
    return completada;
```

```
}
```

```
public String getFecha() {
```

```
    return fecha;
```

```
}
```

```
public void setCompletada(boolean completada) {
```

```
    this.completada = completada;
```

```
}
```

```
public void setFecha(String fecha) {
```

```
    this.fecha = fecha;
```

```
}
```

```
}
```

```
class Pila<T> {  
  
    private LinkedList<T> elementos;  
  
    public Pila() {  
  
        elementos = new LinkedList<>();  
  
    }  
  
    // Agregar un elemento a la pila  
  
    public void agregar(T elemento) {  
  
        elementos.addFirst(elemento);  
  
    }  
  
    // Verificar si la pila está vacía  
  
    public boolean esVacía() {  
  
        return elementos.isEmpty();  
  
    }  
}
```

```
// Quitar el elemento en la cima de la pila
```

```
public T quitar() {  
  
    if (esVacia()) {  
  
        return null;  
  
    }  
  
    return elementos.removeFirst();  
  
}  
}
```

```
class DiarioAventurero {  
  
    private Pila<Misión> pilaMisiones;  
  
    public DiarioAventurero() {  
  
        this.pilaMisiones = new Pila<>();  
  
    }  
  
}
```

```
// Registrar una nueva misión
```

```
public void registrarMisión(String nombre, boolean completada, String fecha) {
```

```
Misión misión = new Misión(nombre, completada, fecha);

pilaMisiones.agregar(misión);

}
```

// Eliminar la última misión si no está completada

```
public void eliminarÚltimaMisión() {

    if (!pilaMisiones.esVacia()) {

        Misión misión = pilaMisiones.quitar();

        if (!misión.isCompletada()) {

            pilaMisiones.agregar(misión);

        }

    }

}
```

// Mostrar todas las misiones

```
public void mostrarMisiones() {

    Pila<Misión> pilaTemporal = new Pila<>();

    while (!pilaMisiones.esVacia()) {
```

```

        Misión misión = pilaMisiones.quitar();

        System.out.println("Nombre: " + misión.getNombre() + ", Completada: " +
misión.isCompletada() + ", Fecha: " + misión.getFecha());

        pilaTemporal.agregar(misión);

    }

    while (!pilaTemporal.esVacia()) {

        pilaMisiones.agregar(pilaTemporal.quitar());

    }

}

// Buscar una misión específica

public boolean buscarMisión(String nombre) {

    Pila<Misión> pilaTemporal = new Pila<>();

    boolean encontrada = false;

    while (!pilaMisiones.esVacia()) {

        Misión misión = pilaMisiones.quitar();

        if (misión.getNombre().equals(nombre)) {

            encontrada = true;

```



```

    }

    pilaTemporal.agregar(misión);

}

while (!pilaTemporal.esVacia()) {

    pilaMisiones.agregar(pilaTemporal.quitar());

}

return encontrada;

}

```

// Editar una misión existente

```

public void editarMisión(String nombre, boolean completada, String fecha) {

    Pila<Misión> pilaTemporal = new Pila<>();

    while (!pilaMisiones.esVacia()) {

        Misión misión = pilaMisiones.quitar();

        if (misión.getNombre().equals(nombre)) {

            misión.setCompletada(completada);

            misión.setFecha(fecha);

        }
    }
}

```

```
        pilaTemporal.agregar(misión);

    }

    while (!pilaTemporal.esVacia()) {

        pilaMisiones.agregar(pilaTemporal.quitar());

    }

}

}

}

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        DiarioAventurero diario = new DiarioAventurero();

        boolean salir = false;

        while (!salir) {

            System.out.println("\nDiario Aventurero");

            System.out.println("1. Registrar misión");

            System.out.println("2. Eliminar última misión");
```

```
System.out.println("3. Mostrar misiones");

System.out.println("4. Buscar misión");

System.out.println("5. Editar misión");

System.out.println("6. Salir");

System.out.print("Elige una opción: ");

int opcion = scanner.nextInt();

scanner.nextLine(); // Consumir nueva línea


switch (opcion) {

    case 1:

        System.out.print("Nombre de la misión: ");

        String nombre = scanner.nextLine();

        System.out.print("¿Completada? (true/false): ");

        boolean completada = scanner.nextBoolean();

        scanner.nextLine(); // Consumir nueva línea

        System.out.print("Fecha (YYYY-MM-DD): ");

        String fecha = scanner.nextLine();

        diario.registrarMisión(nombre, completada, fecha);
```

```
break;
```

```
case 2:
```

```
diario.eliminarÚltimaMisión();
```

```
System.out.println("Última misión eliminada si no estaba completada.");
```

```
break;
```

```
case 3:
```

```
System.out.println("Misiones:");
```

```
diario.mostrarMisiones();
```

```
break;
```

```
case 4:
```

```
System.out.print("Nombre de la misión a buscar: ");
```

```
nombre = scanner.nextLine();
```

```
boolean encontrada = diario.buscarMisión(nombre);
```

```
if (encontrada) {
```

```
    System.out.println("La misión '" + nombre + "' ya fue completada.");
```

```
} else {
```

```
    System.out.println("La misión '" + nombre + "' no fue encontrada o no está  
completada.");
```

```
}
```

```
break;
```

```
case 5:
```

```
System.out.print("Nombre de la misión a editar: ");
```

```
nombre = scanner.nextLine();
```

```
System.out.print("¿Completada? (true/false): ");
```

```
completada = scanner.nextBoolean();
```

```
scanner.nextLine(); // Consumir nueva línea
```

```
System.out.print("Fecha (YYYY-MM-DD): ");
```

```
fecha = scanner.nextLine();
```

```
diario.editarMisión(nombre, completada, fecha);
```

```
break;
```

```
case 6:
```

```
salir = true;
```

```
break;
```

```
default:
```

```
System.out.println("Opción no válida. Por favor, elige una opción del 1 al 6.");
```

```
break;
```

```
    }  
}  
  
scanner.close();  
  
System.out.println("¡Gracias por usar el Diario Aventurero!");  
  
}  
  
}
```

## EJEMPLO DEL USO

```
Diario Aventurero  
1. Registrar misión  
2. Eliminar última misión  
3. Mostrar misiones  
4. Buscar misión  
5. Editar misión  
6. Salir  
Elige una opción: 1  
Nombre de la misión: DERROTAR GIGANTE  
¿Completada? (true/false): true  
Fecha (YYYY-MM-DD): 2025-03-05
```

Diario Aventurero

1. Registrar misión
2. Eliminar última misión
3. Mostrar misiones
4. Buscar misión
5. Editar misión
6. Salir

Elige una opción: 1

Nombre de la misión: Búsqueda del Tesoro

¿Completada? (true/false): false

Fecha (YYYY-MM-DD): 2025-04-26

Diario Aventurero

1. Registrar misión
2. Eliminar última misión
3. Mostrar misiones
4. Buscar misión
5. Editar misión
6. Salir

Elige una opción: 4

Nombre de la misión a buscar: Derrota Gigante

La misión 'Derrota Gigante' no fue encontrada o no está completada.

```
1. Registrar misión
2. Eliminar última misión
3. Mostrar misiones
4. Buscar misión
5. Editar misión
6. Salir
Elige una opción: 5
Nombre de la misión a editar: Busqueda del Tesoro
¿Completada? (true/false): false
Fecha (YYYY-MM-DD): 2025-04-27

Diario Aventurero
1. Registrar misión
2. Eliminar última misión
3. Mostrar misiones
4. Buscar misión
5. Editar misión
6. Salir
Elige una opción: 3
Misiones:
Nombre: Busqueda del Tesoro, Completada: false, Fecha: 2025-04-27
Nombre: DERROTAR GIGANTE, Completada: true, Fecha: 2025-03-05

Diario Aventurero
1. Registrar misión
2. Eliminar última misión
3. Mostrar misiones
4. Buscar misión
5. Editar misión
6. Salir
Elige una opción: 6
¡Gracias por usar el Diario Aventurero!
PS C:\Users\Compu Fire\Desktop\Estructura De Datos EDD\Diario-De-Aventurero>
```



## CONCLUSIÓN

**IMPLEMENTACION TDA** Se utiliza la pila (Pila <T>) que gestiona misiones, lo que permite aplicar el principio de entrada y primero en salir (LIFO)

**DESARROLLAR UNA CONSOLA INTERACTIVA** Se crea un menú que permite la interacción del usuario con el diario, para que el usuario experimente de manera intuitiva y amigable el manejo de datos.

**MANEJO DE EXCEPCIONES Y METODOS NO IMPLEMENTADOS** Se corrigen errores al implementar métodos necesarios para que la pila funcione correctamente. Incluyendo la verificación de la pila que estaba vacía antes de intentar quitar un elemento, esto permite asegurar que la aplicación sea robusta y se maneje de manera adecuada.

**CONCEPTOS DE ORIENTACION A OBJETOS** Se implementan clases con atributos y métodos las cuales representaran misiones y estructura de pila, lo cual ayuda a organizar el código de manera modular y reutilizable, permitiendo facilitar futuras modificaciones y extensiones de la aplicación.

## LINK

<https://github.com/EzryLeal/Diario-De-Aventurero.git>