

Introducción

El propósito de este proyecto es demostrar el uso del patrón de diseño Singleton en Java a través de un sistema de gestión de usuarios. El patrón Singleton garantiza que una clase tenga una única instancia y proporciona un punto de acceso global a esa instancia.

Descripción del Proyecto

Este proyecto implementa un sistema de gestión de usuarios donde se puede agregar, eliminar y listar usuarios. La clase UserManager es responsable de gestionar estos usuarios y sigue el patrón Singleton para asegurar que solo haya una instancia de esta clase en la aplicación.

Estructura del Proyecto

```
src
├── com
│   ├── miempresa
│   │   ├── singleton
│   │   │   ├── UserManager.java
│   │   ├── modelo
│   │   │   ├── User.java
│   │   ├── main
│   │   │   ├── Main.java
```

Clases del Proyecto

User

La clase User representa un usuario en el sistema con atributos como nombre y email. Esta clase contiene métodos para obtener y establecer estos atributos, así como un método para representar al usuario como una cadena de texto.

UserManager

La clase UserManager sigue el patrón Singleton y se encarga de gestionar los usuarios. Esta clase tiene métodos para agregar usuarios, eliminar usuarios por su email y obtener la lista de todos los usuarios. La implementación Singleton asegura que solo haya una instancia de UserManager en toda la aplicación.

Main

La clase Main es la clase principal que interactúa con el sistema de gestión de usuarios a través de la consola. Proporciona un menú para que el usuario pueda agregar, eliminar y mostrar usuarios, así como salir del programa.

Instrucciones de Compilación y Ejecución

Compilar

Navega a la carpeta src y ejecuta los siguientes comandos:

```
sh
```

```
javac com/miempresa/modelo/User.java  
com/miempresa/singleton/UserManager.java  
com/miempresa/main/Main.java
```

Ejecutar

Desde la carpeta src, ejecuta:

```
sh
```

```
java com.miempresa.main.Main
```

Ejemplos de Uso

Menú Principal

El programa presenta un menú principal con las siguientes opciones:

1. Agregar Usuario
2. Eliminar Usuario
3. Mostrar Usuarios
4. Salir

Agregar Usuario

Para agregar un usuario, selecciona la opción 1 e ingresa el nombre y el email del usuario. El sistema confirmará que el usuario ha sido agregado con éxito.

Eliminar Usuario

Para eliminar un usuario, selecciona la opción 2 e ingresa el email del usuario a eliminar. El sistema confirmará que el usuario ha sido eliminado con éxito.

Mostrar Usuarios

Para mostrar la lista de usuarios, selecciona la opción 3. El sistema mostrará todos los usuarios agregados con su nombre y email.

Salir

Para salir del programa, selecciona la opción 4.

Conclusión

El patrón Singleton es una herramienta poderosa para asegurar que solo haya una instancia de una clase en una aplicación. En este proyecto, hemos demostrado cómo se puede utilizar este patrón para gestionar usuarios en un sistema, proporcionando un punto de acceso global y asegurando una única instancia de gestión.