

Hoja de trabajo #4

Repositorio GIT: <https://github.com/Qu3zada22/HDT4---Estructuradedatos.git>

- Revise que ventajas / desventajas hay al utilizar el patrón Singleton en general, ya que su comportamiento es muy similar a una variable global. ¿Cree que su uso es adecuado en este programa?

El patrón Singleton ofrece varias ventajas y desventajas que deben considerarse al decidir si utilizarlo en un programa.

Entre las ventajas se encuentran el control sobre la instancia de una clase, lo que garantiza que solo exista una única instancia en todo el programa. Esto ayuda a ahorrar recursos, como memoria y tiempo de procesamiento, al limitar la creación de objetos. Además, proporciona un punto de acceso global a la instancia, lo que facilita la comunicación entre diferentes partes del programa y simplifica el mantenimiento y la gestión de la clase.

Sin embargo, el Singleton también presenta desventajas. Puede introducir un acoplamiento fuerte en el código, lo que dificulta la prueba unitaria y puede tener efectos secundarios inesperados debido a su estado global. Además, si se utiliza incorrectamente, puede convertirse en un cuello de botella y afectar el rendimiento del sistema. Su complejidad puede aumentar el diseño del código y puede ser difícil de extender para admitir múltiples instancias en el futuro si es necesario.

En cuanto a su idoneidad para este programa específico, depende del contexto y los requisitos del programa. El Singleton puede ser apropiado cuando se necesita exactamente una instancia de una clase en todo el programa y se desea un acceso controlado a esa instancia. Sin embargo, es importante evaluar cuidadosamente si estas condiciones se aplican realmente al problema que se está resolviendo y si las ventajas del Singleton superan sus posibles desventajas en el contexto específico del programa.