Singleton design pattern is a pattern that allows the programmer to restrict the class instantiation to one and provide a global point of access. Singleton pattern implements a simple strategy, “If a class instance does not exist; create a new instance else return the same instance each time an instance is requested”.

Common use of implementing Singletons is use of Logging or interacting with the database maintaining a single database connection and Following are the reasons why Singleton is considered as an anti-pattern:

Use of Singletons make it difficult to unit test the classes because for unit testing, classes must be loosely coupled allowing classes to be tested individually

Singletons are bad when used with multi-threading because with just a single object, the options of threading are limited.

Singletons promote tight coupling between classes Singletons tightly couples the code to the exact object type and removes the scope of polymorphism