

Singleton Class

A Singleton class is a class that allows only one instance of itself to be created and provides a global point of access to that instance.

In object-oriented programming, a class is a blueprint or template for creating objects.

Implementing a Singleton Class in Java

1. Declare a private static variable to hold the single instance of the class.
2. Make the constructor of the class private, so that no other instances can be created.
3. Provide a public static method to return the single instance of the class, creating it if necessary.

Example-1

```
1. public class Singleton
2. {
3.     private static final Singleton instance = new Singleton();
4.     private Singleton()
5.     {
6.     }
7.     public static Singleton getInstance()
8.     {
9.         return instance;
10.    }
11. }
```

Output:

Error: Main method not found in class Singleton, please define the main method as:

```
public static void main(String[] args)
```

or a JavaFX application class must extend javafx.application.

Example-2

```
class singleton
```

```
{
```

```
// datamember must be static
```

```
    private static singleton singleton_instance =null;
```

```
    public String s;
```

```
    private singleton()
```

```
{
```

07/05/2024

```
        s = "Hello How are You";
    }
    public static singleton singleton()
    {
        if (singleton_instance == null)
        {
            singleton_instance = new singleton();
        }
        return singleton_instance;
    }
}
class ABC
{
    public static void main(String args[])
    {
        // instantiating singleton class with variable x
        singleton x = singleton.singleton();
        // instantiating singleton class with variable you
        singleton y = singleton.singleton();
        System.out.println(x.s);
        System.out.println(y.s);
    }
}
/*
D:\java\this>javac Test.java

D:\java\this>java ABC
Hello How are You
Hello How are You
*/
```