# Explanation: Q4 - Singleton Design Pattern

## What is the Singleton Design Pattern?

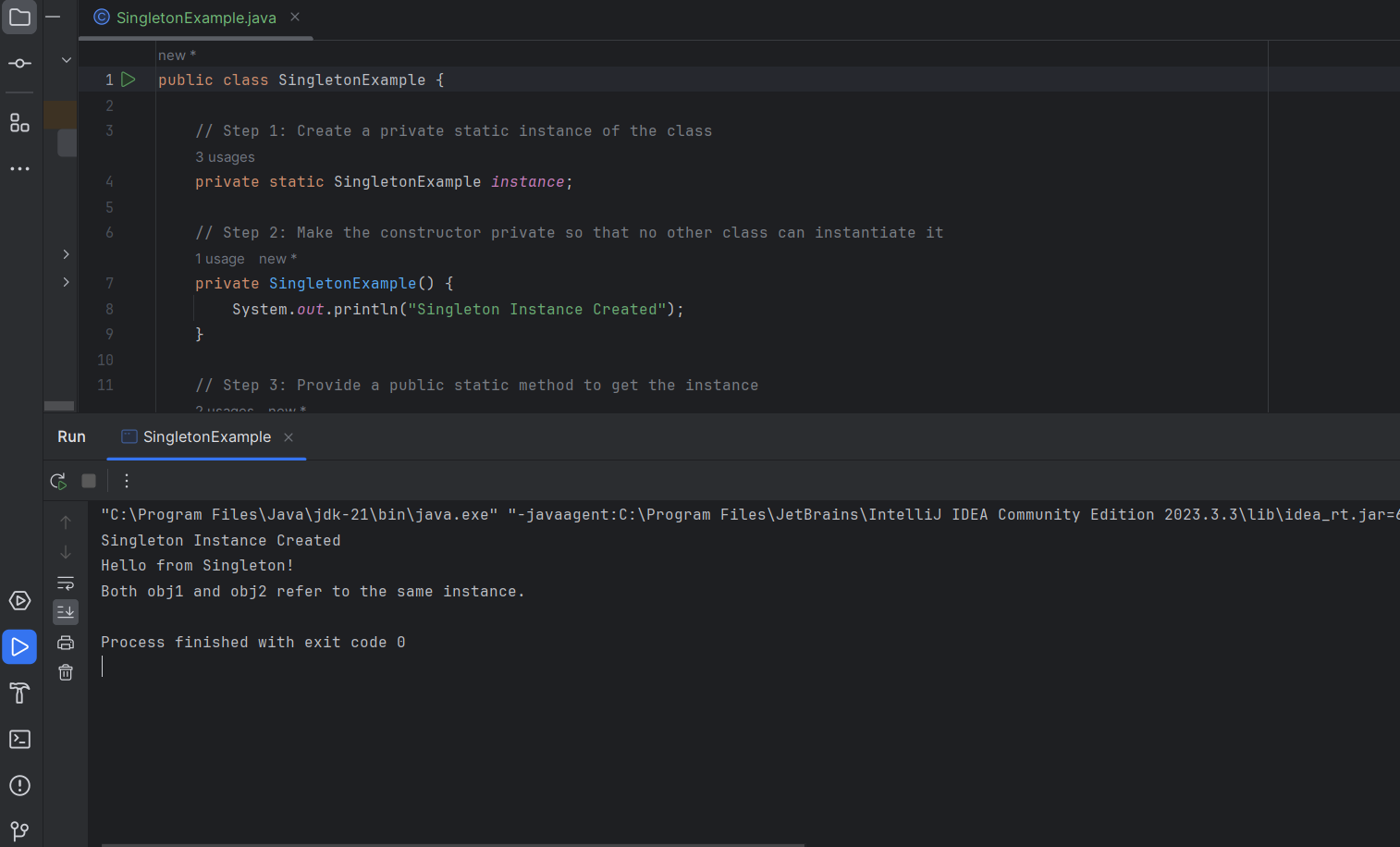
The Singleton Design Pattern ensures that a class has only one instance and provides a global point of access to it. It is commonly used for objects that are needed globally such as configuration managers, logging services, or database connections.

## Code Summary:

1. The constructor is made private to prevent external instantiation.  
2. A static method `getInstance()` is used to create and return the only instance.  
3. The instance is created lazily (only when first accessed).

## Output:

Singleton Instance Created  
Hello from Singleton!  
Both obj1 and obj2 refer to the same instance.



## Explanation:

• The first call to `getInstance()` creates the singleton object.  
• Subsequent calls return the same object.  
• The check `obj1 == obj2` confirms that both references point to the same instance.  
• This demonstrates how Singleton Pattern controls instantiation and provides a single shared object.