# Assignment 03 Application Design: Patterns and Frameworks 44642

Answer **all** the questions below. In your answer for each question explain with sample code or image whichever is preferable.

1. What are generics?

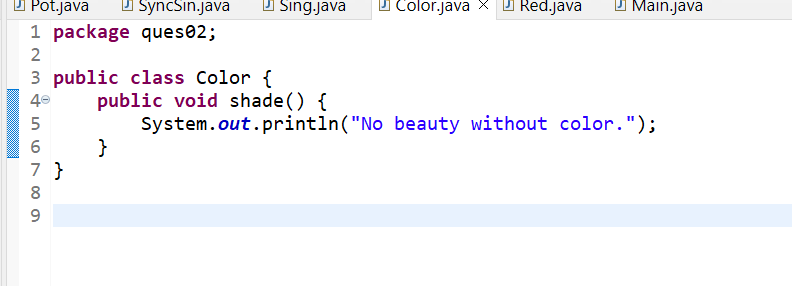
A class or method can be declared once and **used** at runtime with **many different types** of data**. This makes the** code more **configurable, reusable** and **easier** to maintain.Generics **provide** type safety, which **allows** the compiler to find and **expose errors** during the compilation **process,** rather than **while** the program is running. This **improves** software **reliability** and helps prevent errors.

Graphical user interface, application

Description automatically generated

1. Can we change the scope of the overridden method in the subclass for private, public, default and protected? Explain how can it be changed for each scope?

Yes, we can change the scope of the overridden method in the subclass. However, we must notice that we cannot decrease the accessibility of the method.



Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

1. What is the covariant return type?

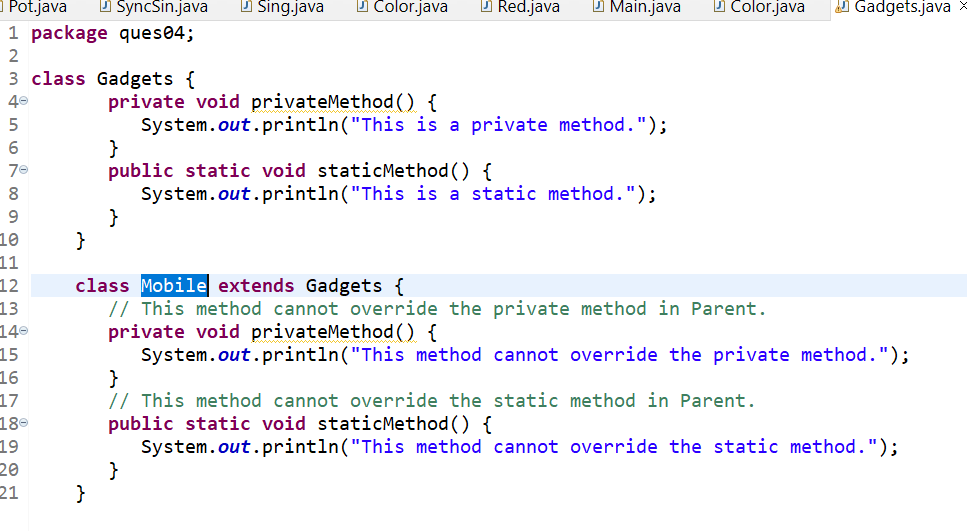
An overriding method's covariant return type is its return type. Without casting the type or checking the return type, it enables one to specify the return type of a method that has been overridden. Only non-primitive return types can be used with a covariant return type.

Text

Description automatically generated

1. Can we override the static and private methods? Why?

Java does not allow overriding of static or private methods. The super class function will be hidden if a similar method with the same return type and method arguments is created in a child class. Method concealing is the term for this. Similar to how you cannot access a private method in a subclass, you also cannot override it.



1. Difference between String Buffer and StringBuilder?

|  |  |
| --- | --- |
| String Buffer | StringBuilder |
| Less efficient | More efficient |
| String Buffer is synchronized | String Builder is not synchronized |
| Slower | Faster |

Graphical user interface, text, application

Description automatically generated

1. Difference between String class and String Buffer?

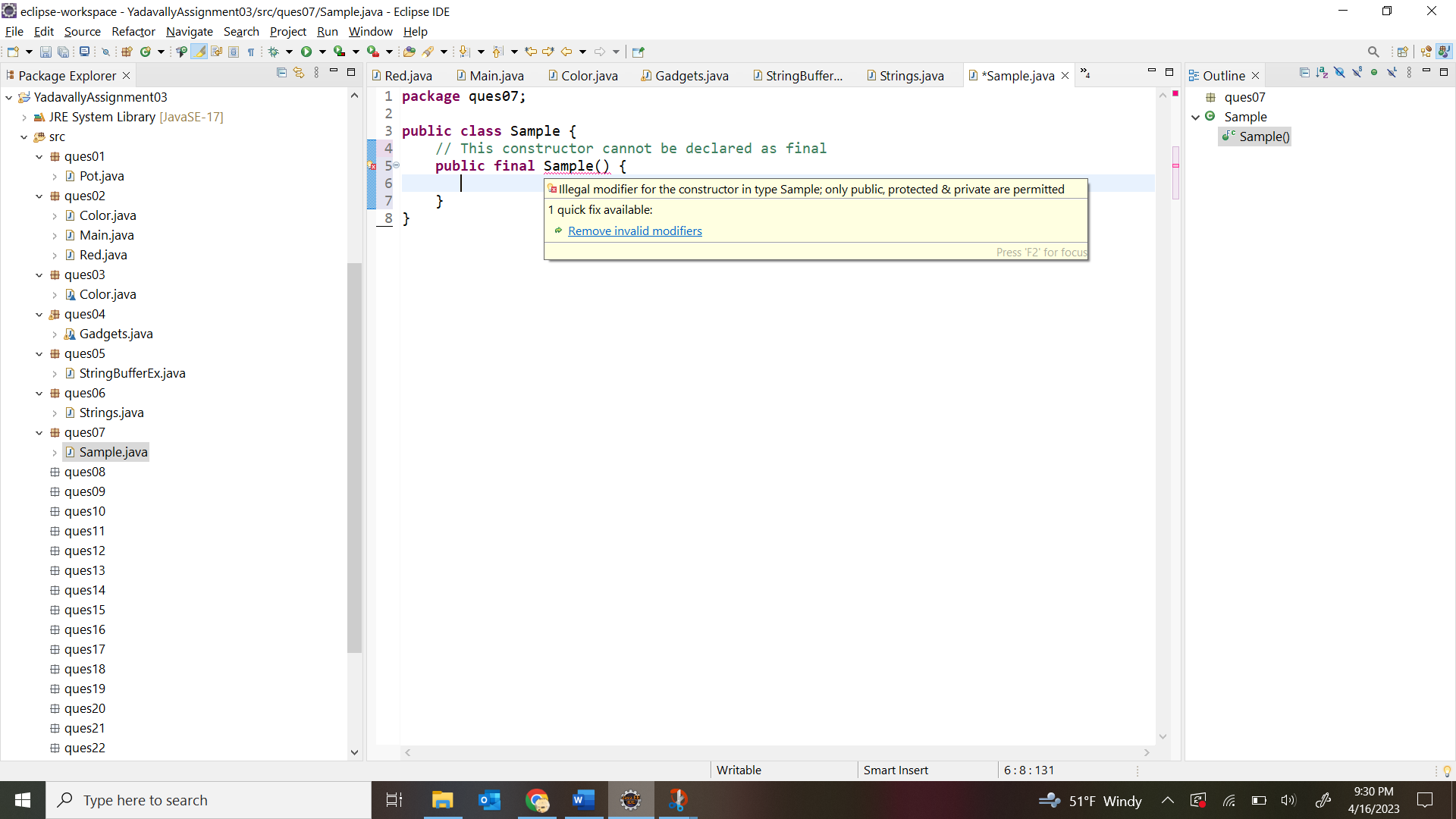
|  |  |
| --- | --- |
| **String class** | **String Buffer** |
| String class is immutable. | String Buffer is mutable. |
| While performing a concatenation operation, the String class is slower. | String buffer is faster. |
| Uses String constant pool. | Uses heap memory. |

Graphical user interface, text, application

Description automatically generated

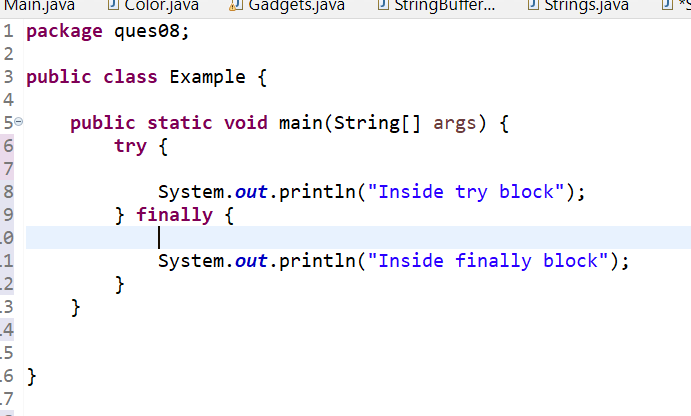
1. Can we declare constructor as final?

A constructor cannot be changed to a final one. Any subclass cannot override a final method.



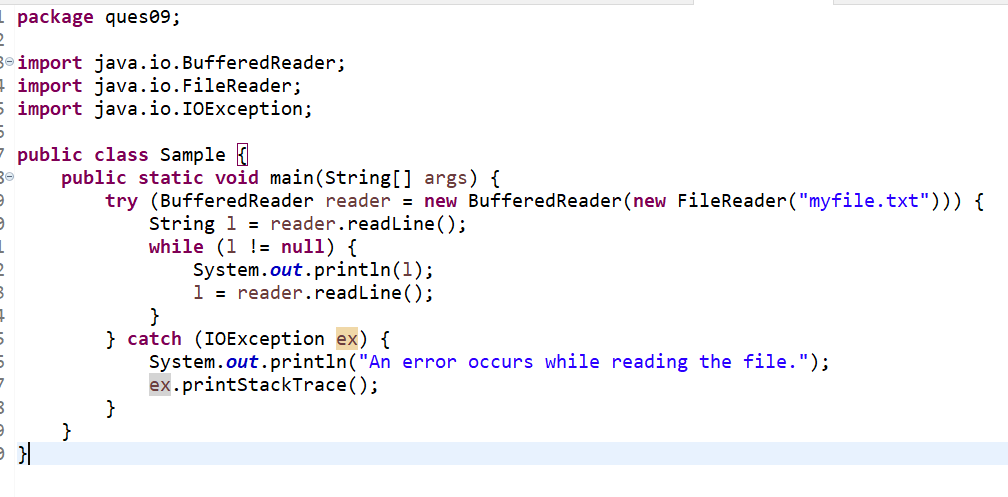
1. Can we have try without catch block in java?

The use of a final block enables the use of a try block without a catch block.



1. What is try with the resource?

The try -with-resources statement is a try statement that declares one or more resources. A resource is an object that must be closed after the program is finished with it. The try -with-resources statement ensures that each resource is closed at the end of the statement.

~~~~

1. Can we modify the throws clause of the superclass method while overriding it in the subclass?

Yes, while overriding the superclass method in the subclass, we can change the throws clause.

Graphical user interface, application, Word

Description automatically generated

1. What is an association, aggregation, and composition in UML?

Composition and Aggregation are subsets of association, which means that they are particular instances of association. An object of one class "owns" an object of another class in both aggregate and composition. However, there is a small distinction: Aggregation suggests a relationship in which the kid can exist separately from the parent.

1. Difference between final, finally and finalize()?

The basic difference between final, finally and finalize is that the [**final**](https://www.javatpoint.com/final-keyword) is an access modifier, [**finally**](https://www.javatpoint.com/finally-block-in-exception-handling) is the block in Exception Handling and [**finalize**](https://www.javatpoint.com/java-object-finalize-method) is the method of object class.

Graphical user interface, text, application

Description automatically generated

1. Difference between Vector and ArrayList?

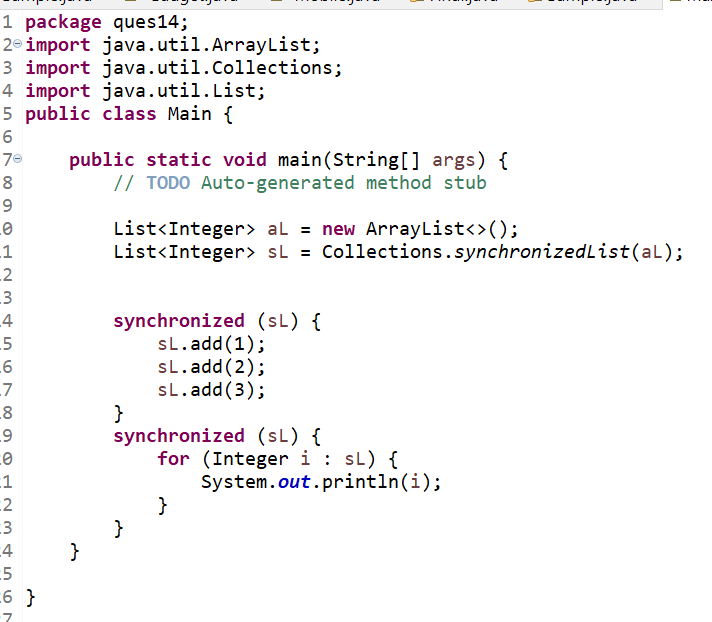
|  |  |
| --- | --- |
| Vector | ArrayList |
| Vector is slow because it is synchronized. | ArrayList is fast. |
| Performance is low. | Performance is high. |
| Only one thread is allowed. | Multiple threads are allowed. |

Text

Description automatically generated with medium confidence

14.What are the different ways to make ArrayList methods synchronized?

* Using Collections.synchronizedList() method
* Using CopyOnWriteArrayList.



Text

Description automatically generated

15.Difference between Hash table and Hash Map?

|  |  |
| --- | --- |
| HashMap | Hash table |
| Null is allowed for both key and value. | Null is not allowed for both key and value. Otherwise, we will get a null pointer exception. |
| Multiple threads can operate simultaneously and hence hashmap’s object is not thread-safe. | At a time only one thread is allowed to operate the Hashtable’s object. Hence it is thread-safe. |

Text

Description automatically generated

16.In Java 8, explain how Hasp Map internally works?

HashMap operates on the idea of a hashing data structure or technique, which inserts an object into the map using the object's hashcode.

Bucket, Hash function (hashCode() method), and Hash value are all components of hashing.Therefore, it is the best-suited data structure for storing key-value pairs that later on can be retrieved in minimum time.

17.Difference between fail fast and fail-safe iterator?

Fail-safe iterators means they will not throw any exception even if the collection is modified while iterating over it.

Fail-fast iterators throw an exception (ConcurrentModificationException) if the collection is modified while iterating over it.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

18.Can we start the thread twice?

No. A thread can only be started once. If you do, an IllegalThreadStateException is raised.

Graphical user interface, text, application, Word

Description automatically generated

19.What are the different ways to create a thread in java? Which one is preferred?

There are two ways to create a thread:

* By extending Thread class.
* By implementing Runnable interface.

Implementing the Runnable interface is a better way to construct a thread in Java because when we extend the Thread class to establish a thread, all of the Thread class methods are inherited while we can only utilize the run method to complete the operation.

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

Graphical user interface, text, application, Word

Description automatically generated

Graphical user interface, text

Description automatically generated

20.What are the different states a thread will go through?

In Java, a thread always exists in any one of the following states.

1. New
2. Active
3. Blocked / Waiting
4. Timed Waiting
5. Terminated.

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

21.What is Serialization? How do we achieve it?

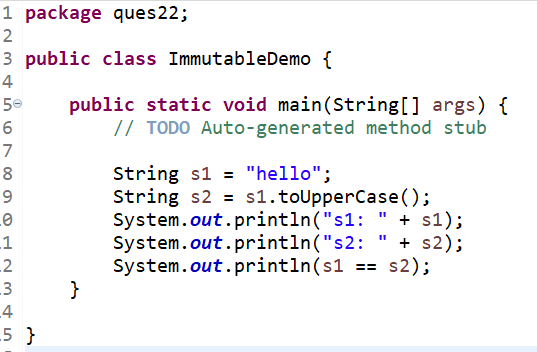
serialization is the process of converting an object into a stream of bytes to more easily save or transmit it. To achieve it, we call the writeObject() method of ObjectOutputStream class.

**Graphical user interface, text

Description automatically generated**

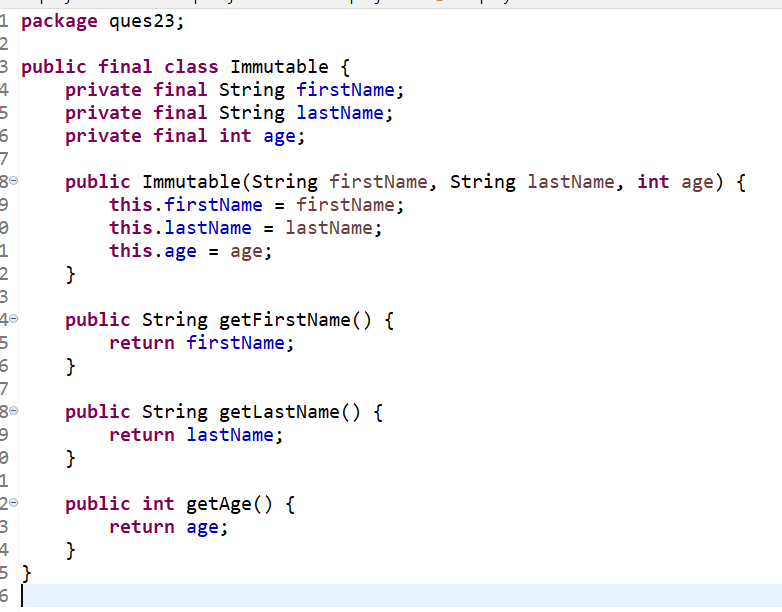
22.What is immutable class? Is String class immutable?

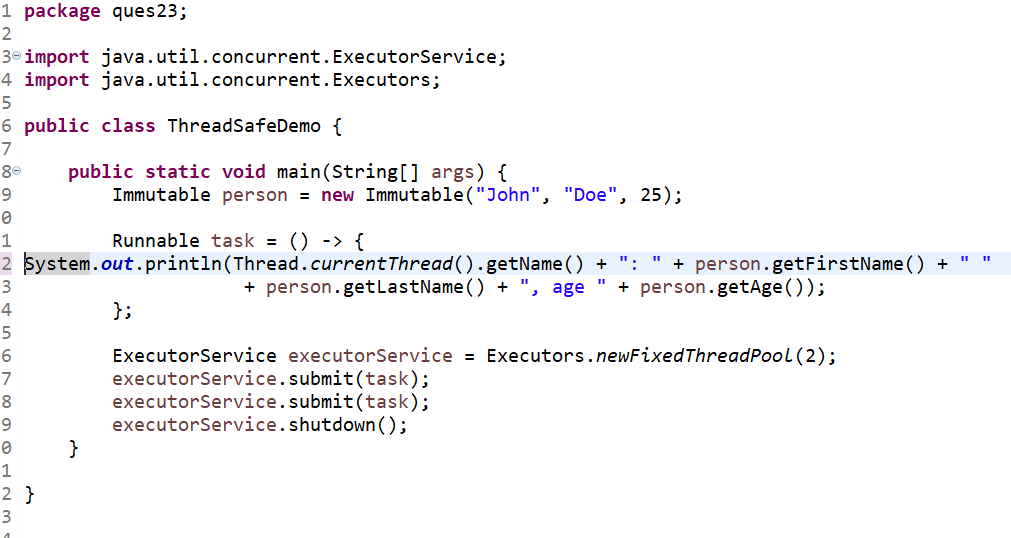
A string is an immutable object, which means once it has been created, it cannot be changed. Every time we alter a string, a new instance is produced.



23.Do immutable classes thread safe? If yes then how?

Immutable objects are thread-safe. Because threads must be able to write to an object's instance variables in order to be a read/write or write/write conflict.





24.Can we call the garbage collector explicitly? Will it trigger the garbage collector?

We can call the Garbage Collectionexplicitly in two ways

* System.gc() method
* Runtime.gc() method

 but it doesn't guarantee that the Garbage collector will start executing immediately.

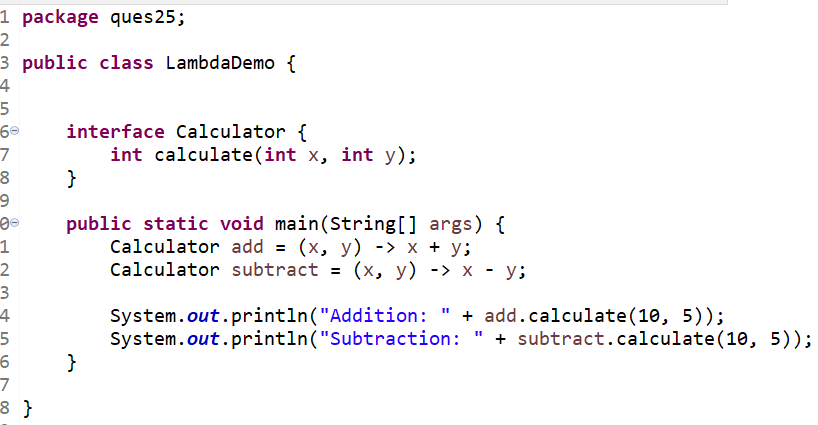
Graphical user interface, text

Description automatically generated

25.What are Java 8 features? Explain all of them with examples?

Java 8 Features:

* Lambdas and functional interfaces



* Default Methods.

Graphical user interface, text, application

Description automatically generated

* Streams

Graphical user interface, text, application

Description automatically generated

26.How to make a pure singleton?

The simplest way to construct a Singleton class is by eager initialization, which creates an instance of the class at the same time as the class is loaded.

Text

Description automatically generated

27.How to make a singleton synchronized?

The synchronized keyword can be used in the singleton class's getInstance() function to synchronize a singleton. The getInstance() method can only be accessed by one thread at a time due to the synchronized keyword, which prevents multiple instances of the singleton class from being generated.

Graphical user interface, text, application, email

Description automatically generated