Assignment Title: Understanding object comparison and String Immutability in Java.

STD ID: 2438520109

Date : 28-Jun-2025

Question 1: What is the difference between a.equals(b); and a==b in Java?

In Java, object comparison can be done using either == operator or the .equals() method. Although they may appear to serve the same purpose, they behave quite differently.

a. The == operator:

This operator checks whether two references point to the same memory location. It is used for reference comparison, not content comparison.

i) For primitive data types, == checks actual values.

ii) For objects, it checks whether both references refer to the same object in memory.

Example:

```
String a = new String ("Hello");
String b = new String ("Hello");
System.out.println (a==b); // false, because
                           they are two different objects
```

b. The .equals() Method:

The .equals() method is used to compare the content or values of two objects. It is defined in the object class and can be overridden by the custom classes.

- This checks whether the sequences of characters in the string are the same.
- In most wrapper and custom classes, .equals() is overridden to check value equality.

Example:

```
String a = new String ("Hello");
String b = new String ("Hello");
System.out.println (a.equals (b)); //true, because
                                   the content is the same.
```

Question 02: Why are Java String Immutable?

Java Strings are immutable, which means once a string object is created, it's value can not be changed. Any operation that appears to modify a String actually creates a new string objects.

Reasons for Immutable:

- Strings are used in many sensitive Place like usernames, passwords, and URLs.
- Java optimizes memory usages by storing String literals in a string pool,
- Multiple threads can safely access strings without synchronization.

Conclusion: String immutability is a key desig decision in Java. It ensures safety, efficien and perfomance. It also simplifies te use of Strings in multithreaded and security-sens -ve applications.