Assignment No:-32

Name:-Suryawanshi Sangramsingh Sambhaji

Batch: - Delta - DCA (Java) 2024 Date:-20/6/2024

1) What is the String class in Java?

Ans: The String class in Java represents a sequence of characters. Strings are immutable objects, meaning their values cannot be changed once created. This class provides various methods to perform operations on strings such as length, concatenation, substring, etc.

2) Differentiate between String and StringBuilder/StringBuffer.

Ans: String is immutable, whereas StringBuilder and StringBuffer are mutable.

StringBuilder is not synchronized and is faster than StringBuffer for most operations.

StringBuffer is synchronized, making it thread-safe but slower compared to StringBuilder.

3) Explain the immutability of strings in Java.

Ans: Immutability means that once a String object is created, its value cannot be changed. Any operation that modifies a string actually creates a new String object with the modified value.

4) How are string literals stored in the string pool?

Ans: String literals in Java are stored in a common pool called the string pool. When a string literal is created, the JVM checks the pool to see if an identical string already exists. If it does, the reference to the existing string is returned; otherwise, the new string is added to the pool.

5) What is the purpose of the concat method in the String class?

Ans: The concat method is used to concatenate two strings. It returns a new string that is the result of concatenating the specified string to the end of the current string.

6) Discuss the significance of the intern method in strings.

Ans: The intern method ensures that all equal strings share the same memory. When intern is called on a string, it checks if the string exists in the string pool. If it does, it returns the reference from the pool; otherwise, it adds the string to the pool and returns its reference.

7) How can you compare two strings in Java?

Ans: You can compare two strings using the equals method (for content comparison) or the == operator (for reference comparison). There are also methods like compareTo for lexicographical comparison.

8) Explain the difference between == and .equals() for string comparison.

Ans: The `==` operator compares the memory addresses (references) of the strings, while the `equals` method compares the actual content of the strings.

9) Discuss the impact of immutability on string manipulation.

Ans: Immutability means every modification to a string creates a new string, which can lead to increased memory usage and reduced performance in scenarios involving frequent string manipulations. Using StringBuilder or StringBuffer can mitigate these issues.

10) How does the length() method work in the String class?

Ans: The length method returns the number of characters in the string. It is a constant-time operation, as the length is stored as an instance variable in the String object.

11) What is the significance of the charAt method in strings?

Ans: The charAt method returns the character at a specified index in the string. It is useful for accessing individual characters in a string.

12) Explain the role of the substring method in string manipulation.

Ans: The substring method extracts a portion of the string based on specified indices and returns it as a new string. It is useful for splitting strings or extracting specific parts of a string.

13) How can you convert a string to uppercase or lowercase in Java?

Ans: You can convert a string to uppercase using the toUpperCase method and to lowercase using the toLowerCase method.

14) Discuss the purpose of the trim method in the String class.

Ans: The trim method removes leading and trailing whitespace from the string and returns the result as a new string.

15) What is the difference between startsWith and endsWith methods?

Ans: The startsWith method checks if the string begins with a specified prefix, while the endsWith method checks if the string ends with a specified suffix.

16) Explain the use of the indexOf and lastIndexOf methods in strings.

Ans: The indexOf method returns the index of the first occurrence of a specified character or substring within the string. The lastIndexOf method returns the index of the last occurrence.

17) How does the replace method work in the String class?

Ans: The replace method returns a new string resulting from replacing all occurrences of a specified character or substring with another character or substring.

18) Discuss the concept of string interpolation in Java.

Ans: Java does not support string interpolation directly. Instead, you can use string concatenation with the + operator or the String.format method to achieve a similar result.

19) Explain the impact of string immutability on memory efficiency.

Ans: While immutability can lead to increased memory usage due to the creation of many temporary strings, it also allows for sharing strings in the string pool, which can save memory in certain cases.

20) What is the purpose of the valueOf method in string conversion?

Ans: The valueOf method converts different types of data (e.g., int, float, char) to their string representation. It is a convenient way to convert various data types to strings.