**Assignment: Practical Implementation of String Classes in Java**

This assignment is designed to help you apply what you have learned about various string classes in Java, including String, StringBuffer, StringBuilder, StringJoiner, and StringTokenizer. Please complete the following exercises and submit your code for review. You can ask any doubts and queries through the WhatsApp number mentioned in the video description or comments.

**Part 1: String Class**

1. **Basic String Operations**

- Write a program to demonstrate the following methods of the String class:
  - `length()`
  - `charAt(int index)`
  - `substring(int beginIndex, int endIndex)`
  - `indexOf(String str)`
  - `replace(char oldChar, char newChar)`

2. **String Comparison**

- Write a program to compare two strings lexicographically. Use both `compareTo()` and `compareToIgnoreCase()` methods. Print the results.

3. **String Immutability**

- Demonstrate the immutable nature of the String class by creating a string and attempting to change it using various methods. Explain the results in comments.

4. **String Permutations**

- Implement a recursive method to find all permutations of a given string. Print all the permutations.

5. **String Constant Pool**

- Write a program to demonstrate the concept of the String Constant Pool. Create strings using both literal and `new` keyword. Use `==` and `equals()` to compare them and explain the results.

**You Tube Playlist Link:**

https://www.youtube.com/playlist?list=PLzrb6iZd6X9IdDOkd_bYc61w04z9_WqiJ

## Part 2: StringBuffer and StringBuilder Classes

1. **Basic Operations with StringBuffer and StringBuilder**

   - Write a program to demonstrate the following methods of the `StringBuffer` and `StringBuilder` classes:

     - `append(String str)`

     - `insert(int offset, String str)`

     - `replace(int start, int end, String str)`

     - `delete(int start, int end)`

     - `reverse()`

2. **Performance Comparison**

   - Write a program to compare the performance of `String`, `StringBuffer`, and `StringBuilder` for concatenation operations. Measure the time taken for a large number of concatenations and print the results.

## Part 3: StringJoiner Class

1. **Using StringJoiner**

   - Write a program to demonstrate the use of the `StringJoiner` class. Create a `StringJoiner` with a delimiter, prefix, and suffix. Add some strings to it and print the final string.

2. **Custom Delimiters**

   - Modify the previous program to use different delimiters and demonstrate how `StringJoiner` can be used to format a list of items (e.g., CSV format).

## Part 4: StringTokenizer Class

**You Tube Playlist Link:**

https://www.youtube.com/playlist?list=PLzrb6iZd6X9IdDOkd_bYc61w04z9_WqiJ

1. **Tokenizing a String**

- Write a program to tokenize a given string using the `StringTokenizer` class. Print each token separately.

2. **Custom Delimiters**

- Modify the previous program to use custom delimiters (e.g., comma, semicolon) and tokenize the string accordingly.

3. **Count Tokens**

- Write a program to count the number of tokens in a given string using the `StringTokenizer` class.

**Part 5: Real-World Applications**

1. **CSV Parser**

- Write a program to parse a CSV (Comma Separated Values) string using `StringTokenizer` and convert it into a 2D array. Print the array.

2. **String Manipulation Utility**

- Create a utility class that provides the following methods:

  - `reverseString(String str)` - Reverses the given string.

  - `isPalindrome(String str)` - Checks if the given string is a palindrome.

  - `capitalizeWords(String str)` - Capitalizes the first letter of each word in the given string.

3. **Password Validator**

- Write a program to validate passwords based on the following criteria:

  - Minimum length of 8 characters.

**You Tube Playlist Link:**

https://www.youtube.com/playlist?list=PLzrb6iZd6X9IdDOkd_bYc61w04z9_WqiJ

- Contains at least one uppercase letter, one lowercase letter, one digit, and one special character.

- Does not contain any whitespace.

- Use the `String` and `StringBuilder` classes to implement this program.

**Good luck, and happy coding!**

**You Tube Playlist Link:**

https://www.youtube.com/playlist?list=PLzrb6iZd6X9IdDOkd_bYc61w04z9_WqiJ