```java
/*
Bhuvana Kanakam
SE21UCSE035 : Lab11

Problem Statement 1 - Concatenation and Substring ( String )
·     Write a Java program that takes two strings as input from the user.

·      Concatenate the two strings and print the result.

·      Prompt the user to enter a starting index and an ending index.

·      Extract and print the substring of the concatenated string based on the provided indices.
*/

import java.util.Scanner ;
public class ConcatenationAndSubstring {
      public static void main ( String[] args ) {
            Scanner scanner = new Scanner ( System.in ) ;

            System.out.println ("Enter the first string:") ;
            String str1 = scanner.nextLine() ;
            System.out.println ("Enter the second string:") ;
            String str2 = scanner.nextLine() ;

            String concatenatedString = str1 + str2     ;
            System.out.println("The Concatenated String generated is : " + concatenatedString) ;

            System.out.println("Enter the starting index for substring from the concatenated String:");
            int startIndex = scanner.nextInt();

                  System.out.println("Enter the ending index for substring from the concatenated String:");
            int endIndex = scanner.nextInt();

            if (startIndex >= 0 && endIndex < concatenatedString.length() && startIndex <= endIndex) {
                  String substring = concatenatedString.substring(startIndex, endIndex);
                  System.out.println("The Substring from the indices provided is : " + substring);
            } else {
                        System.out.println("Invalid indices for the substring.");
            }

            scanner.close();
      }
}
```

```
Last login: Thu Nov 23 09:23:20 on ttys000
➜  ~ cd documents
➜  documents cd college
➜  college cd semester5
➜  semester5 cd CS3105-objectOrientedProgramming
➜  CS3105-objectOrientedProgramming cd Lab
➜  Lab cd Lab 11
cd: no such file or directory: /Users/bhuvanakanakam/d
ocuments/college/semester5/CS3105-objectOrientedProgra
mming/11
➜  Lab cd Lab11
➜  Lab11 javac ConcatenationAndSubstring.java
➜  Lab11 java ConcatenationAndSubstring
Enter the first string:
Bhuvana
Enter the second string:
Kanakam
The Concatenated String generated is : BhuvanaKanakam
Enter the starting index for substring from the concat
enated String:
3
Enter the ending index for substring from the concaten
ated String:
6
The Substring from the indices provided is : van
➜  Lab11
```

```java
/*
Bhuvana Kanakam
SE21UCSE035 : Lab11

Problem Statement 2 – Reverse and Replace ( StringBuffer )
·	Create a StringBuffer object and initialize it with a string of your choice.

·	Reverse the content of the StringBuffer and print the result.

·	Replace a specific substring within the StringBuffer with another substring and print the modified content.

*/

public class ReverseAndReplace {
	public static void main(String[] args) {
		StringBuffer stringBuffer = new StringBuffer("Hello, World! Welcome to Java Programming!");

		stringBuffer.reverse() ;
		// reverse () method of StringBuffer, it reverses the characters in the StringBuffer in place.
		System.out.println("Reversed StringBuffer: " + stringBuffer) ;

		stringBuffer.replace (7, 12, "Universe") ;
		//replace ( stratIndex, endIndex, replacement )
		System.out.println("Modified StringBuffer: " + stringBuffer) ;
	}
}
```

```
[➜  Lab11 javac ReverseAndReplace.java
[➜  Lab11 java ReverseAndReplace
Reversed StringBuffer: !gnimmargorP avaJ ot emocleW !dlroW ,olleH
Modified StringBuffer: !gnimmaUniverse avaJ ot emocleW !dlroW ,olleH
➜  Lab11
```

```
/*clear
Bhuvana Kanakam
SE21UCSE035 : Lab11

Problem Statement 3 - String Comparison:
·	Write a method that compares two strings without using the built-in equals() method.

·	The method should return true if the strings are equal and false otherwise.

·	Test the method with different strings to verify its correctness.
*/

import java.util.Scanner;

public class StringComparison {
	public static void main(String[] args) {
		Scanner scanner = new Scanner(System.in);

		System.out.println("Enter the first string for comparison:");
		String str1 = scanner.next();

		System.out.println("Enter the second string for comparison:");
		String str2 = scanner.next();

		scanner.nextLine();
		boolean isEqual = compareStrings(str1, str2);
		System.out.println("Are the strings equal? " + isEqual);

		scanner.close();
	}

	public static boolean compareStrings(String str1, String str2) {
		if (str1.length() != str2.length()) {
			return false;
		}
		for (int i = 0; i < str1.length(); i++) {
			if (str1.charAt(i) != str2.charAt(i)) {
				return false;
			}
		}
		return true;
	}
}
```

```
[➜  Lab11 javac StringComparison.java
[➜  Lab11 java StringComparison
Enter the first string for comparison:
Bhuvana
Enter the second string for comparison:
Bhuvan
Are the strings equal? false
[➜  Lab11 java StringComparison
Enter the first string for comparison:
Bhuvan
Enter the second string for comparison:
Bhuvan
Are the strings equal? true
➜  Lab11
```

```java
/*
Bhuvana Kanakam
SE21UCSE035 : Lab11

Problem Statement 4 - StringBuffer Append and Insert:

·	Create two StringBuffer objects and initialize them with different strings.

·	Append the content of the second StringBuffer to the first one.

·	Insert a new substring into the modified StringBuffer at a specified index.

·	Print the final result.

*/

public class StringBufferAppendAndInsert {
     public static void main(String[] args) {
          StringBuffer buffer1 = new StringBuffer("This is ");
          StringBuffer buffer2 = new StringBuffer("a Lab.");

          System.out.println("The String without Appended and Inserting is : " +
buffer1);

          buffer1.append(buffer2);

          buffer1.insert(8, "modified ");
          System.out.println("Appended and Inserted StringBuffer: " + buffer1);
    }
}
```

```
➜  Lab11 javac StringBufferAppendAndInsert.java
➜  Lab11 java StringBufferAppendAndInsert
The String without Appended and Inserting is : This is
Appended and Inserted StringBuffer: This is modified a Lab.
➜  Lab11
```

```java
/*
Bhuvana Kanakam
SE21UCSE035 : Lab11

Problem Statement 5 - Palindrome Check:

·	Write a Java program that checks if a given string is a palindrome.

·	Utilize both String and StringBuffer to demonstrate different approaches.

·	Print whether the input string is a palindrome or not.

*/

import java.util.Scanner;

public class PalindromeCheck {
	public static void main(String[] args) {
		Scanner scanner = new Scanner(System.in);

		System.out.println("Enter a string for palindrome check:");
		String inputString = scanner.next();

		boolean isPalindromeString = inputString.equals(new
StringBuilder(inputString).reverse().toString());
		System.out.println("Is the input string a palindrome? (Using String): " +
isPalindromeString);

		boolean isPalindromeBuffer = new
StringBuffer(inputString).reverse().toString().equals(inputString);
		System.out.println("Is the input string a palindrome? (Using
StringBuffer): " + isPalindromeBuffer);

		scanner.close();
	}
}
```

```
➜  Lab11 javac PalindromeCheck.java
➜  Lab11 java PalindromeCheck
Enter a string for palindrome check:
Kanakam
Is the input string a palindrome? (Using String): false
Is the input string a palindrome? (Using StringBuffer): false
➜  Lab11 java PalindromeCheck
Enter a string for palindrome check:
Kanak
Is the input string a palindrome? (Using String): false
Is the input string a palindrome? (Using StringBuffer): false
➜  Lab11 java PalindromeCheck
Enter a string for palindrome check:
abba
Is the input string a palindrome? (Using String): true
Is the input string a palindrome? (Using StringBuffer): true
➜  Lab11
```