# Task 1: String Operations

Write a method that takes two strings, concatenates them, reverses the result, and then extracts the middle substring of the given length. Ensure your method handles edge cases, such as an empty string or a substring length larger than the concatenated string.

**Code-:**

```
package com.wipro.assignment;
public class StringOperations {
    public static String
extractMiddleSubstring(String str1,
String str2, int length) {
        // Concatenate the two strings
        String concatenated =
str1.concat(str2);

        // Reverse the concatenated
string
        StringBuilder reversed = new
StringBuilder(concatenated).reverse();

        // Get the length of the reversed
string
        int reversedLength =
reversed.length();
```

```
        // If the length of the reversed
string is smaller than the specified
length, return null
        if (reversedLength < length) {
            return null;
        }

        // Calculate the starting index
of the middle substring
        int startIndex = (reversedLength
- length) / 2;

        // Extract the middle substring
        return
reversed.substring(startIndex, startIndex
+ length);
    }

    public static void main(String[]
args) {
        String str1 = "Hello";
        String str2 = "World";
        int length = 5;

        String result =
extractMiddleSubstring(str1, str2,
length);
```

```
            if (result != null) {
                System.out.println("Middle
substring of length " + length + ": " +
result);
            } else {
                System.out.println("Substring
length is larger than the concatenated
string.");
            }
        }
}
```

**Output: -**