Amrutvahini College of Engineering, Sangamner
Department of Electronics and Telecommunication Engineering
Subject: JavaScript (Elective-III) (BE, E&TC)

Experiment No.: 04

Date of Performance:

TITLE: Compare two strings using various methods

Aim: Write a JavaScript program to compare two strings using various methods.

Problem Statement:

Write a JavaScript program to following operations on a given string,

i. Reverse string

ii. Replace characters of a string

iii String is Palindrome

Hardware Requirement:

Any CPU with Pentium Processor or similar, 256 MB RAM or more, 1 GB Hard Disk or more

Software Requirement:

- **IDE Software:** Visual Studio Code / Sublime Text3
- **Web Browser:** Google Chrome / Firefox/ Internet Explorer.
- **Operating System:** Windows 7/10, Linux or MAC OS

Objective:

1. Learn basics of how to use document object.
2. Learn different tokens and operators used in JavaScript.
3. To learn looping in JavaScript.

Theory:

The JavaScript string is an object that represents a sequence of characters.

There are 2 ways to create string in JavaScript

1. By string literal
2. By string object (using new keyword)

1) By string literal

The string literal is created using double quotes. The syntax of creating string using string literal is given below:

```
var stringname="string value";
```

2) By string object (using new keyword)

The syntax of creating string object using new keyword is given below:

```
var stringname=new String("string literal");
```

Here, new keyword is used to create instance of string.

Comparing strings

In C, the strcmp() function is used for comparing strings. In JavaScript, you just use the less-than and greater-than operators:

```
const a = 'a';
const b = 'b';
if (a < b) {       // true
  console.log(`${a} is less than ${b}`)
} else if (a > b) {
  console.log(`${a} is greater than ${b}`)
} else {
  console.log(`${a} and ${b} are equal.`)
}
```

A similar result can be achieved using the localeCompare() method inherited by String instances.

Note that a === b compares the strings in a and b for being equal in the usual case-sensitive way. If you wish to compare without regard to upper- or lower-case characters, use a function similar to this:

```
function isEqual(str1, str2) {
  return str1.toUpperCase() === str2.toUpperCase();
}
```

JavaScript String Methods

Let's see the list of JavaScript string methods with examples.

Methods	Description
charAt()	It provides the char value present at the specified index.
charCodeAt()	It provides the Unicode value of a character present at the specified index.
concat()	It provides a combination of two or more strings.

indexOf()	It provides the position of a char value present in the given string.
lastIndexOf()	It provides the position of a char value present in the given string by searching a character from the last position.
search()	It searches a specified regular expression in a given string and returns its position if a match occurs.
match()	It searches a specified regular expression in a given string and returns that regular expression if a match occurs.
replace()	It replaces a given string with the specified replacement.
substr()	It is used to fetch the part of the given string on the basis of the specified starting position and length.
substring()	It is used to fetch the part of the given string on the basis of the specified index.
slice()	It is used to fetch the part of the given string. It allows us to assign positive as well negative index.
toLowerCase()	It converts the given string into lowercase letter.
toLocaleLowerCase()	It converts the given string into lowercase letter on the basis of host?s current locale.
toUpperCase()	It converts the given string into uppercase letter.
toLocaleUpperCase()	It converts the given string into uppercase letter on the basis of host?s current locale.
toString()	It provides a string representing the particular object.
valueOf()	It provides the primitive value of string object.
split()	It splits a string into substring array, then returns that newly created array.
trim()	It trims the white space from the left and right side of the string.

Conclusion:

____

____

____

____

____

Review Questions:

1. How to declare String array. Write an example.
2. How to get the number of occurrences of each letter in specified string in JavaScript?
3. How to split multiline string into an array of lines in JavaScript? Write code for same.
4. Explain with example use of back-tick instead of single quote or double quote in representing string variable.

REFERENCES:

1. Jon Duckett, “JavaScript & JQuery: Interactive Front-End Web Development”, Wiley, ISBN-13. 978-1118531648
2. David Flanagan, “JavaScript: The Definitive Guide”, O’Reilly, 6th Edition, ISBN: 9781491952023.
3. Mike Mackgrath, “Javascripts in easy steps” Dreamtech