# JS - String Manipulation

## 1. Basic String Operations

Strings in JavaScript are sequences of characters enclosed in either single quotes, double quotes, or backticks (`). Here are some basic operations:

```
// Creating Strings
let str1 = 'Hello, World!';
let str2 = "JavaScript is fun!";
let str3 = `This is a string using backticks`;

// Accessing Characters (indexing starts at 0)
let firstChar = str1[0]; // 'H'
let lastChar = str1[str1.length - 1]; // '!'

// String Concatenation
let concatenated = str1 + ' ' + str2; // "Hello, World! JavaScript is fun!"
let templateString = `${str1} ${str2}`; // Template Literals: "Hello, World! JavaScript is fun!"

// String Length
let length = str1.length; // 13

/*
| Operation             | Example               | Description                                                  |
|-----------------------|-----------------------|--------------------------------------------------------------|
| Creating String       | let str = 'Hello!';   | A string can be created with single, double, or backticks.  |
| Accessing Characters | str[0] -> 'H'         | Access characters by their index.                           |
| Concatenation         | str1 + ' ' + str2     | Combine strings using + operator or template literals.      |
| Length                | str.length            | Get the length (number of characters) of the string.        |
*/
```

## 2. Common String Methods

JavaScript provides a variety of built-in methods to manipulate strings. Below are some of the most commonly used string methods:

### 2.1. `toUpperCase()` & `toLowerCase()`

These methods convert a string to uppercase or lowercase respectively.

```
// Convert string to uppercase
let upperCaseStr = str1.toUpperCase(); // "HELLO, WORLD!"

// Convert string to lowercase
let lowerCaseStr = str2.toLowerCase(); // "javascript is fun!"
```

### 2.2. `trim()`

The `trim()` method removes whitespace from both ends of a string.

```
let paddedString = '   Hello, World!   ';
let trimmedString = paddedString.trim(); // "Hello, World!"
```

### 2.3. `charAt()` & `charCodeAt()`

The `charAt()` method returns the character at a specified index, and `charCodeAt()` returns the Unicode value of the character at that index.

```
let char = str1.charAt(0); // "H"
let unicode = str1.charCodeAt(0); // 72 (Unicode for 'H')
```

### 2.4. `indexOf()` & `lastIndexOf()`

These methods return the position of the first or last occurrence of a substring within a string.

```
let indexOfWord = str1.indexOf('World'); // 7 (Position of 'World')
let lastIndexOfWord = str1.lastIndexOf('o'); // 8 (Last occurrence of 'o')
```

### 2.5. `includes()`

The `includes()` method checks if a string contains a specified substring and returns true or false.

```
let containsWord = str2.includes('JavaScript'); // true
let doesNotContain = str1.includes('Python'); // false
```

### 2.6. `replace()`

The `replace()` method replaces a substring with a new string. You can use regular expressions with it for advanced replacements.

```
let replacedString = str2.replace('fun', 'awesome'); // "JavaScript is awesome!"
```

### 2.7. `split()`

The `split()` method splits a string into an array of substrings based on a specified delimiter.

```
let sentence = 'Hello, how are you?';
let words = sentence.split(' '); // ["Hello,", "how", "are", "you?"]
```

### 2.8. `slice()` & `substring()`

The `slice()` and `substring()` methods are used to extract a part of a string.

```
let slicedStr = str1.slice(0, 5); // "Hello"
let substringStr = str1.substring(0, 5); // "Hello"
```

### 2.9. `concat()`

The `concat()` method is used to join two or more strings together.

```
let fullStr = str1.concat(' ', str2); // "Hello, World! JavaScript is fun!"
```

### 2.10. `repeat()`

The `repeat()` method repeats a string a specified number of times.

```
let repeatedStr = 'Hello! '.repeat(3); // "Hello! Hello! Hello! "
```

## 3. Summary of String Methods

```
| Method          | Example                            | Description
|---------------|-----------------------------------|----------------------------------------------------------
| toUpperCase()   | str.toUpperCase()                  | Converts string to uppercase.
| toLowerCase()   | str.toLowerCase()                  | Converts string to lowercase.
| trim()          | str.trim()                         | Removes whitespace from both ends of the string.
| charAt()        | str.charAt(0)                      | Returns the character at the specified index.
| charCodeAt()    | str.charCodeAt(0)                  | Returns the Unicode value of the character at the specif
| indexOf()       | str.indexOf('World')               | Returns the index of the first occurrence of a substring
| lastIndexOf()   | str.lastIndexOf('o')               | Returns the index of the last occurrence of a substring.
| includes()      | str.includes('JavaScript')         | Checks if the string contains the substring.
| replace()       | str.replace('World', 'JavaScript')| Replaces a substring with a new substring.
| split()         | str.split(' ')                     | Splits the string into an array of substrings.
| slice()         | str.slice(0, 5)                    | Extracts a portion of a string and returns it.
| substring()     | str.substring(0, 5)                | Extracts a portion of a string between two indexes.
| concat()        | str.concat(' ', 'World!')          | Combines two or more strings into one.
| repeat()        | str.repeat(3)                      | Repeats the string a given number of times.
```