**String Property**

1. charAt (): The string objects charAt () method returns the new string according to the index of the specified string.

Let string = “Hello World”

console.log (string. charAt (0))

console.log (string. charAt (1))

console.log (string. charAt (2))

console.log (string. charAt (3))

console.log (string. charAt (4))

Output: H e l l o

1. concat (): The concat method adds two or more strings or arguments together.

Let str1 = “Hello”

Let str2 = “world”

Let str3 = str1.concat (str2)

Output: Hello World

1. includes (): The includes method performs a case-sensitive search, whereas to find a string from another string returning true or false as output.

Let string = “Hello World”

string. includes(‘hello’)

Output: false

1. endsWith (): The endsWith () method determines whether a string ends with the character or not. It returns a Boolean output like true or false.

Let string = “Hello World”

string. endsWith(‘hello’)

Output: false

1. indexOf (): returns the index within the calling string object of the first occurrence of the specified value. It returns -1 if the value is not found.

Let string = “Hello World”

string. indexOf (‘’, 0)

Output: 0

1. lastIndexOf(): searching backward from the index. Returns -1 if the value is not found.

Let string = “Hello World”

string. lastIndexOf (‘a’)

Output: -1

1. replace(): The replace method replaces the string within the given string.

Let string = “Hello World”

string. replace (” World”,” Bangladesh”)

Output: Hello Bangladesh

1. slice: The slice method extracts a section of a string and returns a new string, without modifying the original string.

Let string = “Hello World”

string. slice (6)

Output: World

**JavaScript Array**

1. Filter: The filter method creates a **new array** according to some test implementation by the function.

const words = ['spray', 'limit', 'elite', 'exuberant', 'destruction', 'present'];

const result = words. filter (word => word. length > 6);

1. Find: The find method returns **only the single array** from an array or object or returns a single element from an array.

const array1 = [5, 12, 8, 130, 44];

const found = array1.find(element => element > 10);

1. forEach: The forEach method returns **only a single array** from the executed functions.

const array1 = ['a', 'b', 'c'];

array1.forEach(element => console.log(element));

1. shift: shift method removes the first element from an array and returns the removed element.

const array1 = [1, 2, 3];

const firstElement = array1.shift();

console.log(array1);

1. unshift: The unshift method adds one or more elements at the beginning of the array.

const array1 = [1, 2, 3];

console.log(array1.unshift(4, 5)); Output: [4,5,1,2,3]

1. map: The map() method creates a new array populated with the results of calling a provided function on every element in the calling array.

**JavaScript try-catch**

JavaScript try catch uses from test block of code and detecting of error. Like the try statement gives you to test the block of code for errors and catch statement handles the error. Both try and catch statement are used together for running code detect error and also handle the error.

E.g.

try {

Alert (“hello world”)

}

Catch {

Console.log (“alert is not working.”)

}

**ES6**

1. **Functions with Default Parameter Values:** Default function parameters allows named parameter to be initialized with default values or pass undefined or no values.

E.g. function add (a, b = 1) {

return a + b;

}

console.log (add (5, 2));

Output: 7

console.log (add (2));

Output: 3

1. **Destructuring**: Destructuring makes it possible to unpack the values from arrays or objects. By using destructuring we can get values from array and objects property.

E.g. let a, b, rest;

[a, b] = [10, 20];

console.log(a);

Output: 10

1. **Spread Operator:** Spread operator just copies all the array elements from the previous array and returns a new array.

E.g. function sum (x, y, z) {

return x + y + z;

}

const numbers = [1, 2, 3]’

console.log (sum (...numbers));

1. Rest Operator