

All JavaScript Methods

@coding_comics

String Methods

- **charAt(index)**

```
1 let str = "Hello";  
2 console.log(str.charAt(1)); // "e"
```

- **includes(searchString)**

```
1 let str = "Hello World";  
2 console.log(str.includes("World")); // true
```

- **replace(searchValue, newValue)**

```
1 let str = "Hello World";  
2 console.log(str.replace("World", "JavaScript")); // "Hello JavaScript"
```

- **slice(start, end)**

```
1 let str = "Hello World";  
2 console.log(str.slice(0, 5)); // "Hello"
```

- **toUpperCase()**

```
1 let str = "hello";  
2 console.log(str.toUpperCase()); // "HELLO"
```

- **split(separator)**

```
1 let str = "a,b,c,d";  
2 console.log(str.split(",")); // ["a", "b", "c", "d"]
```

Array Methods

- **concat(array1, array2)**

```
1 let arr1 = [1, 2];  
2 let arr2 = [3, 4];  
3 console.log(arr1.concat(arr2)); // [1, 2, 3, 4]
```

- **filter(callback)**

```
1 let arr = [1, 2, 3, 4];  
2 console.log(arr.filter(num => num > 2)); // [3, 4]
```

- **find(callback)**

```
1 let arr = [1, 2, 3, 4];  
2 console.log(arr.find(num => num > 2)); // 3
```

- **map(callback)**

```
1 let arr = [1, 2, 3, 4];  
2 console.log(arr.map(num => num * 2)); // [2, 4, 6, 8]
```

- **push(element)**



```
1 let arr = [1, 2];  
2 arr.push(3);  
3 console.log(arr); // [1, 2, 3]
```

- **pop()**



```
1 let arr = [1, 2, 3];  
2 arr.pop();  
3 console.log(arr); // [1, 2]
```

Object Methods

- **assign(target, ...sources)**



```
1 let target = { a: 1 };  
2 let source = { b: 2 };  
3 Object.assign(target, source);  
4 console.log(target); // { a: 1, b: 2 }
```

- **keys(obj)**



```
1 let obj = { a: 1, b: 2 };  
2 console.log(Object.keys(obj)); // ["a", "b"]
```

- **values(obj)**

```
1 let obj = { a: 1, b: 2 };  
2 console.log(Object.values(obj)); // [1, 2]
```

- **entries(obj)**

```
1 let obj = { a: 1, b: 2 };  
2 console.log(Object.entries(obj)); // [["a", 1], ["b", 2]]
```

Math Methods

- **abs(x)**

```
1 console.log(Math.abs(-5)); // 5
```

- **ceil(x)**

```
1 console.log(Math.ceil(4.2)); // 5
```

- **floor(x)**

```
1 console.log(Math.floor(4.8)); // 4
```


- `max(x1, x2, ...)`

```
1 console.log(Math.max(1, 2, 3)); // 3
```

- `random()`

```
1 console.log(Math.random()); // Random number between 0 and 1
```

Date Methods

- `getDate()`

```
1 let date = new Date();  
2 console.log(date.getDate()); // Current day of the month (1-31)
```

- `getFullYear()`

```
1 let date = new Date();  
2 console.log(date.getFullYear()); // Current year (e.g., 2024)
```

- `getTime()`

```
1 let date = new Date();  
2 console.log(date.getTime()); // Milliseconds since January 1, 1970
```

- **setDate(day)**



```
1 let date = new Date();
2 date.setDate(15);
3 console.log(date); // Date set to the 15th of the current month
```

JSON Methods

- **parse(text)**



```
1 let jsonString = '{"name": "John", "age": 30}';
2 let obj = JSON.parse(jsonString);
3 console.log(obj); // { name: "John", age: 30 }
```

- **stringify(value)**



```
1 let obj = { name: "John", age: 30 };
2 let jsonString = JSON.stringify(obj);
3 console.log(jsonString); // '{"name": "John", "age": 30}'
```

Promise Methods

- **all(promises)**



```
1 let p1 = Promise.resolve(1);
2 let p2 = Promise.resolve(2);
3 Promise.all([p1, p2]).then(result => console.log(result)); // [1, 2]
```

- `race(promises)`



```
1 let p1 = new Promise((resolve) => setTimeout(resolve, 500, "One"));
2 let p2 = new Promise((resolve) => setTimeout(resolve, 100, "Two"));
3 Promise.race([p1, p2]).then(result => console.log(result)); // "Two"
```



“ Code is like humor. When you have to explain it, it’s bad ”

Coding Comics