

Day 8



JavaScript String Methods:

String Methods in JavaScript

JavaScript provides built-in methods to work with strings. All these methods return new strings — because strings are immutable.

Example: `.length` -> to get the length of the string

```
JS main.js > ...
1  let name = "Aditya";
2  console.log(name);
3  console.log(name.length);
4

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya
6
```

Example: escape sequence character are treated as 1 character

```
JS main.js > ...
1  let name = "Aditya\"";
2  console.log(name);
3  console.log(name.length);
4

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya"
7
```

Example: `.toUpperCase()` to convert all characters to Upper case

```
JS main.js > [e] name
1  let name = "Aditya";
2  console.log(name);
3  console.log(name.toUpperCase());

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya
ADITYA
```

Example: .toLowerCase() to convert to lower case.

```
JS main.js > ...
1 let name = "Aditya";
2 console.log(name);
3 console.log(name.toLowerCase());
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya
aditya
```

Example: .slice() -> to get the part of string

```
JS main.js > ...
1 let name = "Aditya";
2 console.log(name);
3 console.log(name.slice());
4 console.log(name.slice(2));
5 console.log(name.slice(2,4));
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya
Aditya
itya
it
```

Example: .replace(old,new)

```
JS main.js > ...
1 let name = "Aditya Kumar";
2 console.log(name);
3 console.log(name.replace("Kumar", "Thakur"));
4 console.log(name.replace("K", "Thakur"));
5 console.log(name.replace("", "Thakur"));
6 console.log(name.replace("z", "Thakur"));
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya Kumar
Aditya Thakur
Aditya Thakurumar
ThakurAditya Kumar
Aditya Kumar
```

Example: .trim()

```
JS main.js > [?] name
1 let name = " Aditya Kumar ";
2 console.log(name.trim());
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya Kumar
```

Example: .concat()

```
JS main.js > ...
1  let name1 = "Aditya";
2  let name2 = "Utsav";
3  console.log(name1,"is a friend of", name2);
4  console.log(name1.concat("is a brother of", name2));
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS E:\JavaScript\Day1-10\Day8> node main.js
Aditya is a friend of Utsav
Adityais a brother ofUtsav
```

Example: using for...of loop to print character of strings

```
JS main.js > ...
1  let name1 = "Aditya";
2  for (const key of name1) {
3    console.log(key);
4  }
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
● PS E:\JavaScript\Day1-10\Day8> node main.js
A
d
i
t
y
a
```

Introduction to Arrays:

What is an Array?

An Array is a special variable that can store multiple values in a single variable. Each value in an array is called an element, and each element has an index (starting from 0).

Ways to Create Arrays

Method	Example	Description
Array Literal (recommended)	<code>let arr = [1, 2, 3];</code>	Easiest and most common way
Array Constructor	<code>let arr = new Array(1, 2, 3);</code>	Less common, but valid
Empty Array + Push	<code>let arr = []; arr.push(10);</code>	Create first, fill later

Array Indexing

- Index starts from 0
- Last element index = array.length - 1

Important to remember:

Concept	Explanation
Arrays are zero-indexed	First element at index 0
Arrays can store mixed data types	[1, "Hello", true]
Arrays are mutable	You can change elements
Strings are immutable, Arrays are mutable	Important difference
Use for...of for arrays	Simpler iteration
length is dynamic	Changes as you add/remove items

Example: basic arrays

```
JS arrays.js > ...
1  let a = ["Aditya", "Utsav"];
2  let b = [2,3];
3  let c = ["Adii", true, 2, "Utuu"];
```

Example: another way to create an array, as well as we are printing

```
7 let arr = new Array(1,2,3);
8 console.log(arr);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node .\arrays.js
[ 1, 2, 3 ]
```

Example: accessing values of arrays

```
10 let arr = [2,3,4];
11 console.log(arr[0]);
12 console.log(arr[2]);
13 console.log(arr[5]);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node .\arrays.js
2
4
undefined
```

Example: to get the length of the array

```
16 let arr = [1,3,5];
17 console.log(arr.length);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node .\arrays.js
3
```

Example: arrays are mutable

```
19 let arr = [2,3,4];
20 console.log(arr);
21 arr[2] = 6;
22 console.log(arr);
23 arr[4] = 8;
24 console.log(arr);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day8> node .\arrays.js
[ 2, 3, 4 ]
[ 2, 3, 6 ]
[ 2, 3, 6, <1 empty item>, 8 ]
```

--The End--