



Arsalan Khattak
@khattakdev

7 METHODS TO LOOP IN JAVASCRIPT



FOR LOOP

The most simple and basic loop. Very straightforward to use.



```
1 const numbers = [22, 34, 46, 58];
2
3 for (let i = 0; i < numbers.length; i++) {
4     console.log(numbers[i] * 2);
5 }
6
7 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

WHILE LOOP

It's similar to **for loop** in terms of complexity and also very straightforward to use.



```
1 const numbers = [22, 34, 46, 58];
2 let i = 0;
3 while ( i < numbers.length) {
4     console.log(numbers[i] * 2);
5     i++;
6 }
7
8 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

DO-WHILE LOOP

The syntax looks similar to **while loop** but the only difference is the code is executed one time before checking the condition.



```
1 const numbers = [22, 34, 46, 58];
2 let i = 0;
3 do {
4     console.log(numbers[i] * 2);
5     i++;
6 } while ( i < numbers.length)
7
8 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

FOR-OF LOOP

A loop that can be used for arrays to iterate over them without specifying the last limit. This loop ends when reach the last index of the array.



```
1 const numbers = [22, 34, 46, 58];
2
3 for(const num of numbers) {
4     console.log(num*2);
5 }
6
7 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

FOR-IN LOOP

Similar to for-of loop but it's used on JavaScript objects.



```
1 const numbers = {  
2     1: 22,  
3     2: 34,  
4     3: 46,  
5     4: 58  
6 };  
7  
8 for(const num in numbers) {  
9     console.log(numbers[num] * 2);  
10 }  
11  
12 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

MAP LOOP

An ES6 approach to iterate over an array and store the result in a new variable.



```
1 const numbers = [22, 34, 46, 58];
2 const doubleofNumbers = numbers.map(num => {
3     return num * 2;
4 }
5
6 // Result: doubleofNumbers = 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe

FOR-EACH LOOP

Also an ES6 approach and similar to the “**map**” but it only iterates over an array without making any changes or returning any values.

This means you can't tweak the array but you can use the individual indexes' values.



```
1 numbers.forEach(num => {  
2     console.log( num * 2);  
3 })  
4  
5 // Result: 44 68 92 116
```



Arsalan Khattak
@khattakdev

Swipe



@Khattakdev

FOLLOW FOR MORE
SUCH CONTENT



CHECKOUT MY YOUTUBE
@KHATTAKDEV AS WELL