

Day 33

“Web Development + Security”

JavaScript Arrays:

What Is an Array?

An array is a list-like object used to store multiple values in a single variable.

Example:

```
let fruits = ["Apple", "Banana", "Mango"];  
console.log(fruits); // ["Apple", "Banana", "Mango"]
```

➔ Arrays can hold any data type: numbers, strings, objects, other arrays, even functions.

Declaring arrays:

1. // Using literal syntax (recommended)

```
let numbers = [1, 2, 3, 4, 5];
```
2. // Using Array constructor

```
let colors = new Array("Red", "Green", "Blue");
```
3. // Empty array

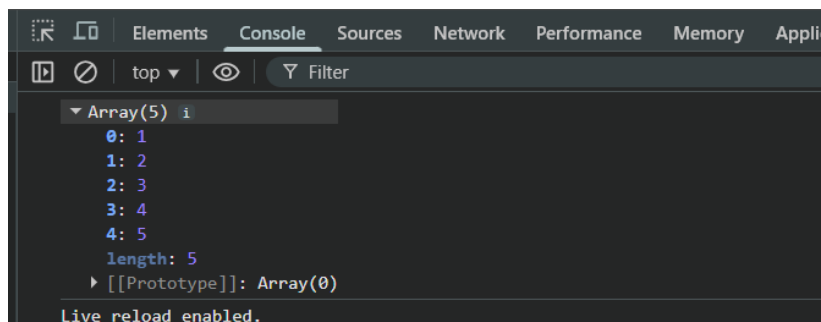
```
let emptyArr = [];
```

A very basic array:

Code:

```
JS arrays.js > ...  
1 //Arrays in javascript  
2  
3 let a = [1,2,3,4,5];  
4 console.log(a); //printing the array
```

Output:

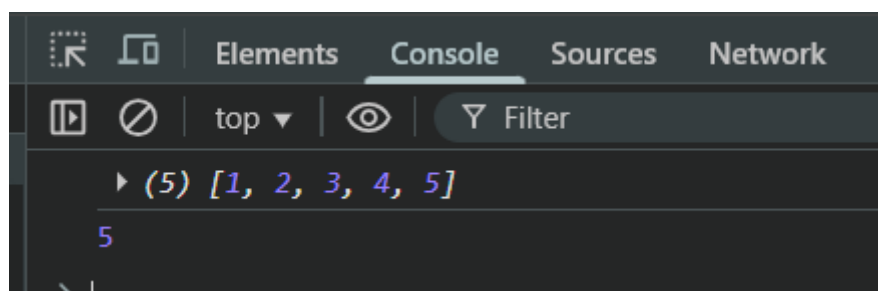


Now, to print the length of the array:

Code:

```
JS arrays.js > ...  
3   let a = [1,2,3,4,5];  
4   console.log(a); //printing the array  
5  
6   //length of the array  
7   console.log(a.length); //5
```

Output:



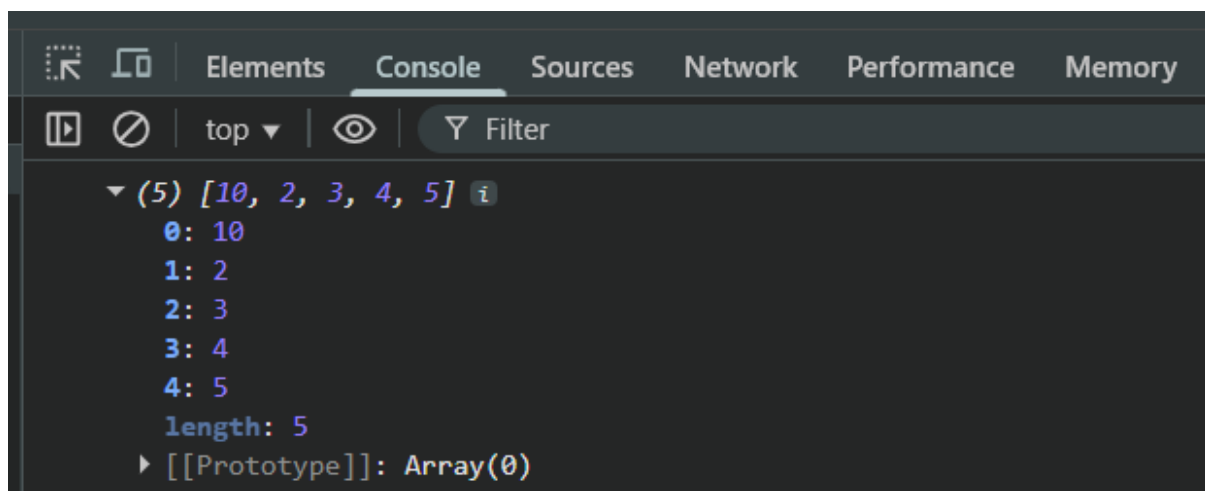
The screenshot shows a browser's developer console with the 'Console' tab selected. It displays two log entries: first, an array `[1, 2, 3, 4, 5]` with a length of 5, and second, the number `5`.

Example: arrays are mutable

Code:

```
9   //Arrays are mutable  
10  let a = [1,2,3,4,5];  
11  a[0] = 10;  
12  console.log(a); // [10,2,3,4,5]
```

Output:



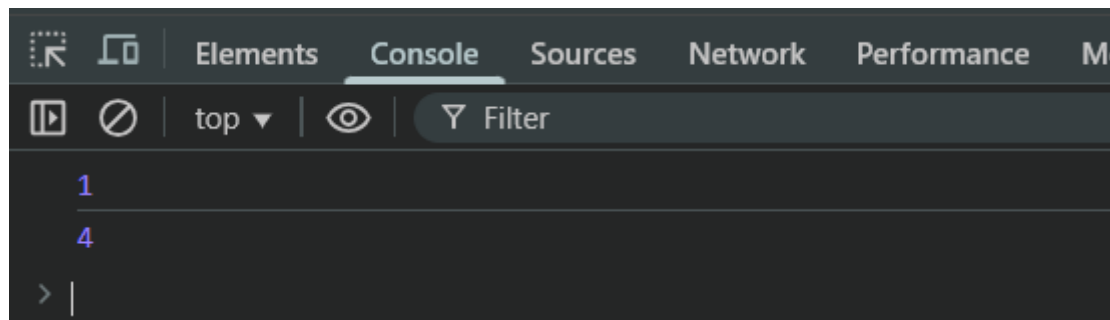
The screenshot shows a browser's developer console with the 'Console' tab selected. It displays a log entry for an array `[10, 2, 3, 4, 5]` with a length of 5. The array is expanded to show its elements: `0: 10`, `1: 2`, `2: 3`, `3: 4`, `4: 5`, and `length: 5`. The prototype is also shown as `[[Prototype]]: Array(0)`.

Example: printing the specific element of array using its index

Code:

```
14 //printing specific element of array
15 let a = [1,2,3,4,5];
16 console.log(a[0]); //1
17 console.log(a[3]); //4
```

Output:

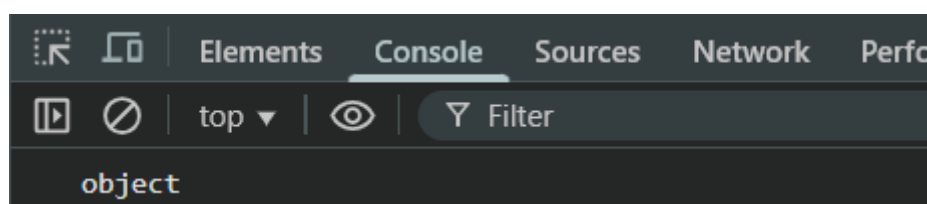


Example: type of array is object

Code:

```
18
19 //Type of array
20 let a = [1,2,3,4,5];
21 console.log(typeof a); //object
22
```

Output:

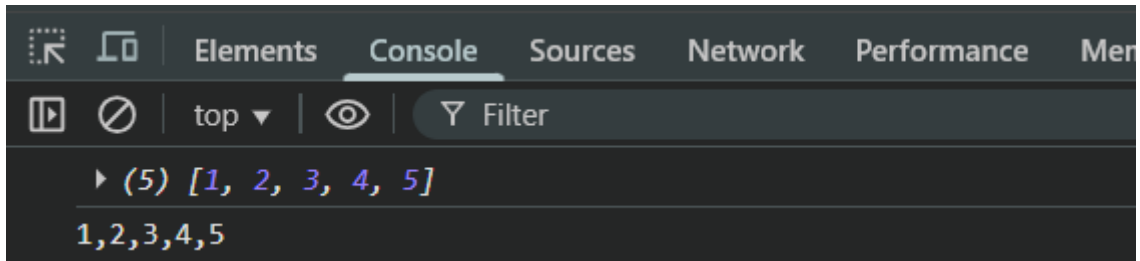


Example: changing array to string

Code:

```
23 //Changing array to string
24 let a = [1,2,3,4,5];
25 console.log(a); // [1,2,3,4,5]
26 console.log(a.toString()); //1,2,3,4,5
```

Output:

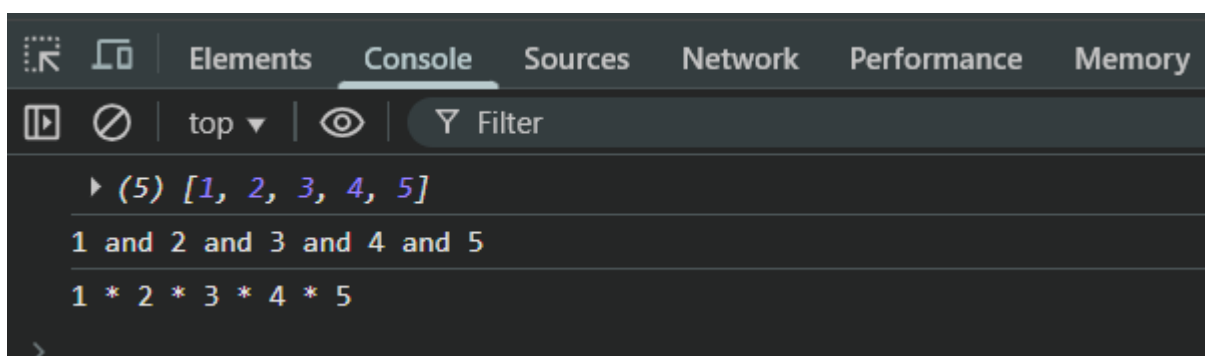


Example: use of join() to join all the arrays elements using a separator

Code:

```
28 //Use of join function
29 let a = [1,2,3,4,5];
30 console.log(a); // [1,2,3,4,5]
31 console.log(a.join(" and ")); //1 and 2 and 3 and 4 and 5
32 console.log(a.join(" * ")); //1 * 2 * 3 * 4 * 5
```

Output:

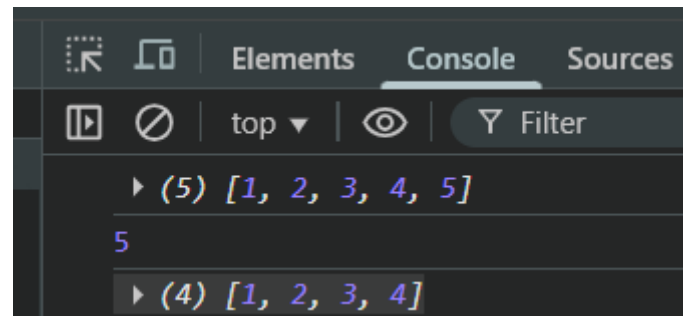


Example: of pop(), it removes the last element of the array

Code:

```
34 //use of pop function
35 let a = [1,2,3,4,5];
36 console.log(a); // [1,2,3,4,5]
37 console.log(a.pop()); //5
38 console.log(a); // [1,2,3,4]
```

Output:

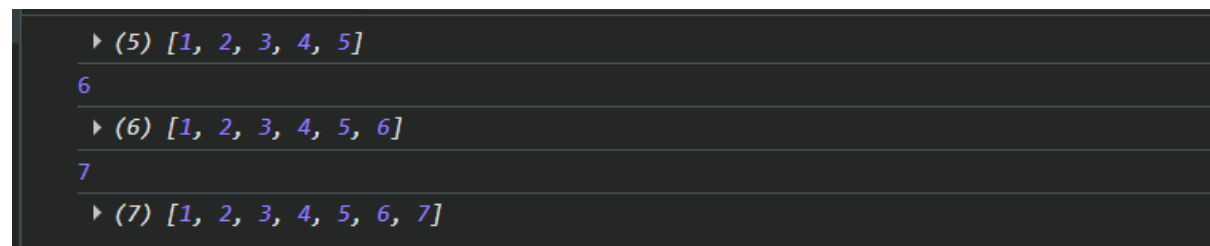


Example: use of push(), it will push the element as the last element of the array

Code:

```
40 //use of push function
41 let a = [1,2,3,4,5];
42 console.log(a); // [1,2,3,4,5]
43 console.log(a.push(6));
44 console.log(a); // [1,2,3,4,5,6]
45 console.log(a.push(7));
46 console.log(a); // [1,2,3,4,5,6,7]
```

Output:

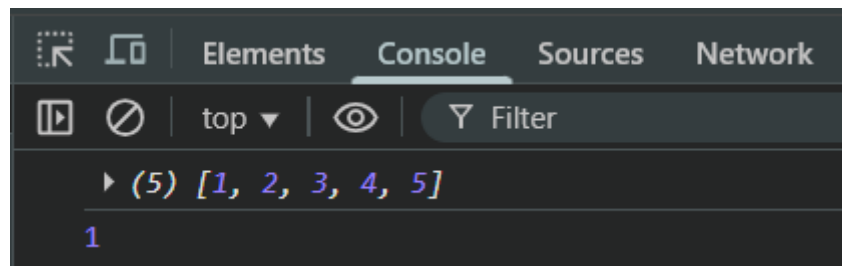


Example: shift() it removes the first element and return it

Code:

```
48 //use of shift function
49 let a = [1,2,3,4,5];
50 console.log(a); // [1,2,3,4,5]
51 console.log(a.shift()); //1
```

Output:

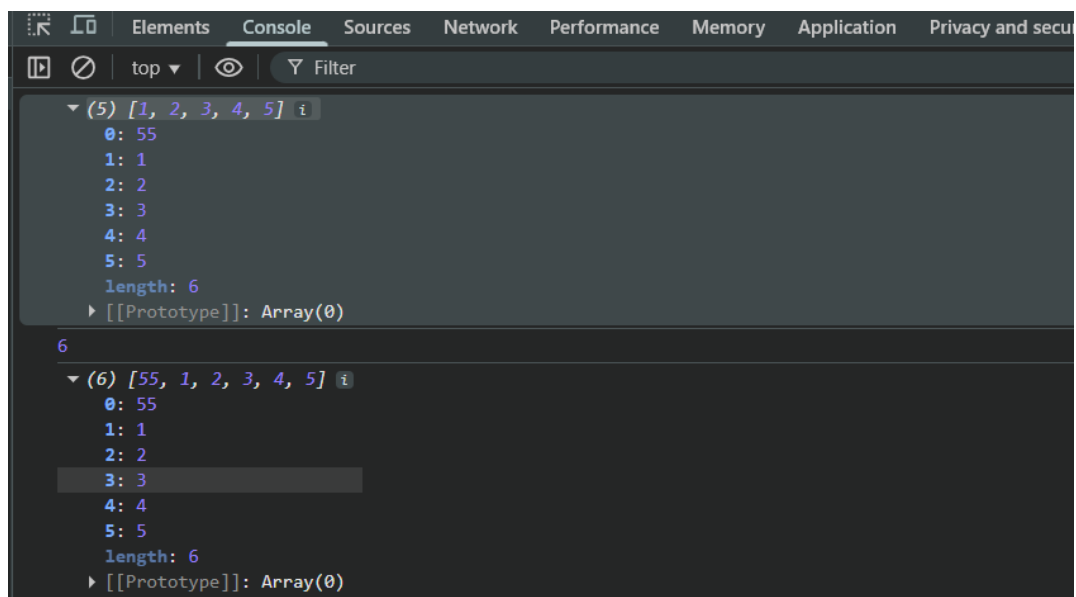


Example: use of unshift() to push the element from the front

Code:

```
53 //use of unshift function
54 let a = [1,2,3,4,5];
55 console.log(a); // [2,3,4,5]
56 console.log(a.unshift(55)); //5
57 console.log(a); // [6,1,2,3,4,5]
```

Output:

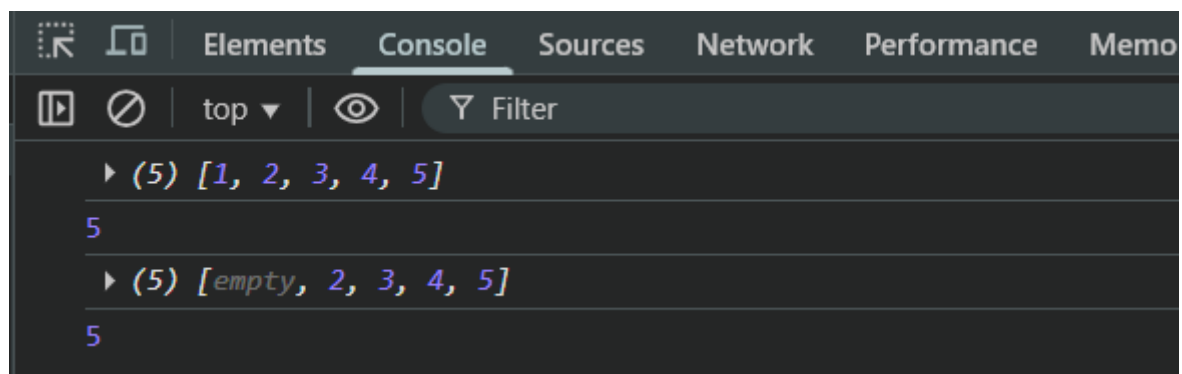


Example: use of delete, when we do so the memory of the deleted value of array be present but the value be absent.

Code:

```
59 //use of delete function
60 let a = [1,2,3,4,5];
61 console.log(a); // [1,2,3,4,5]
62 console.log(a.length); //5
63 delete a[0];
64 console.log(a); // [ <1 empty item>, 2, 3, 4, 5 ]
65 console.log(a.length); //5
```

Output:

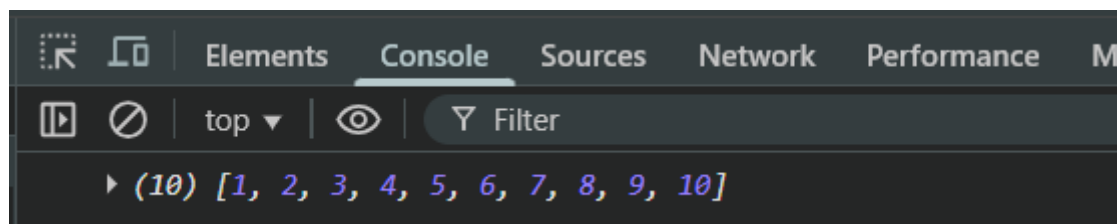


Example: use of concat operator in order to join two arrays

Code:

```
67 //use of concatenate function
68 let a = [1,2,3,4,5];
69 let b = [6,7,8,9,10];
70 let c = a.concat(b);
71 console.log(c); // [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ]
```

Output:



Also,

```
67 //use of concatenate function
68 let a = [1,2,3,4,5];
69 let b = [6,7,8,9,10];
70 // let c = a.concat(b);
71 // console.log(c); // [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ]
72 let d = a.concat(a,b)
73 console.log(d); // [ 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ]
74
75
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS E:\FullStackDevelopment\Day31-40\Day33> node .\arrays.js

```
[
  1, 2, 3, 4, 5, 1,
  2, 3, 4, 5, 6, 7,
  8, 9, 10
]
```

❖ PS E:\FullStackDevelopment\Day31-40\Day33>

Example: use of sort()

Code:

```
75 //Use of sort function
76 let a = [5,3,8,4,2,1];
77 console.log(a); // [5,3,8,4,2,1]
78 a.sort();
79 console.log(a); // [1,2,3,4,5,8]
```

Output:

Elements Console Sources Network Performance

top ▼ [Filter]

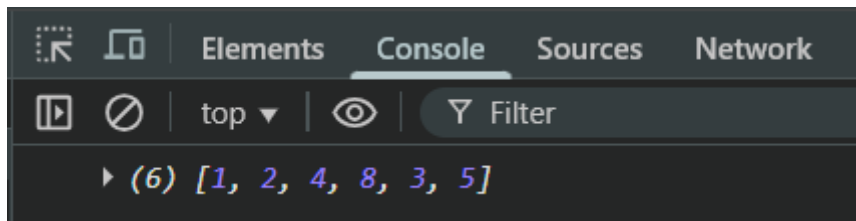
```
▶ (6) [5, 3, 8, 4, 2, 1]
▶ (6) [1, 2, 3, 4, 5, 8]
```


Example: use of reverse()

Code:

```
81 // Use of reverse function
82 let a = [5,3,8,4,2,1];
83 a.reverse();
84 console.log(a); // [8,5,4,3,2,1]
85 // Note: sort function converts number to string and then sorts it
```

Output:

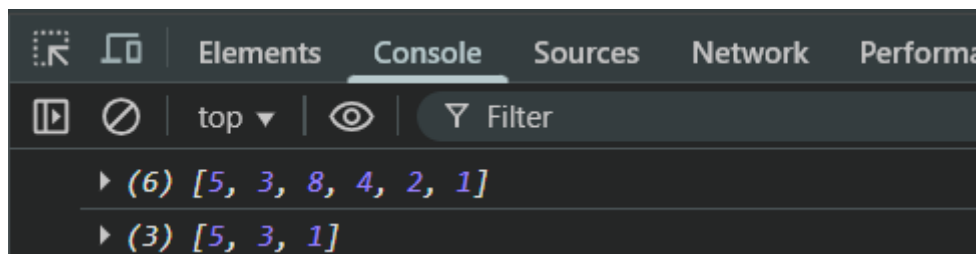


Example: use of splice()

Code:

```
87 //use of splice function
88 let a = [5,3,8,4,2,1];
89 console.log(a); // [5,3,8,4,2,1]
90 a.splice(2,3); //starting index, number of elements to be removed
91 console.log(a); // [5,3,1]
```

Output:

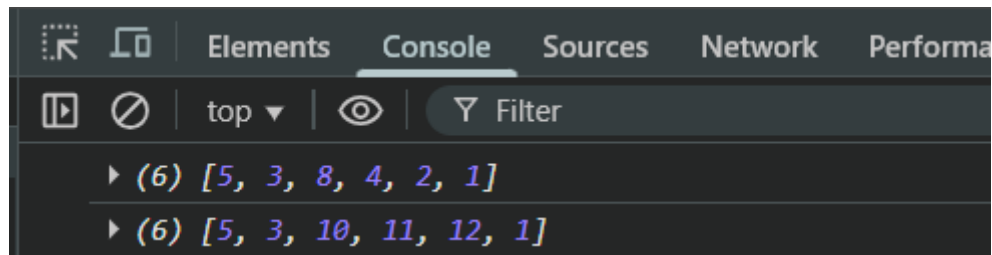


Example: inserting the elements using the splice()

Code:

```
93 //Use of splice function
94 let a = [5,3,8,4,2,1];
95 console.log(a); // [5,3,8,4,2,1]
96 a.splice(2,3,10,11,12); //starting index, number of elements to be removed, elements to be added
97 console.log(a); // [5,3,10,11,12,1]
```

Output:

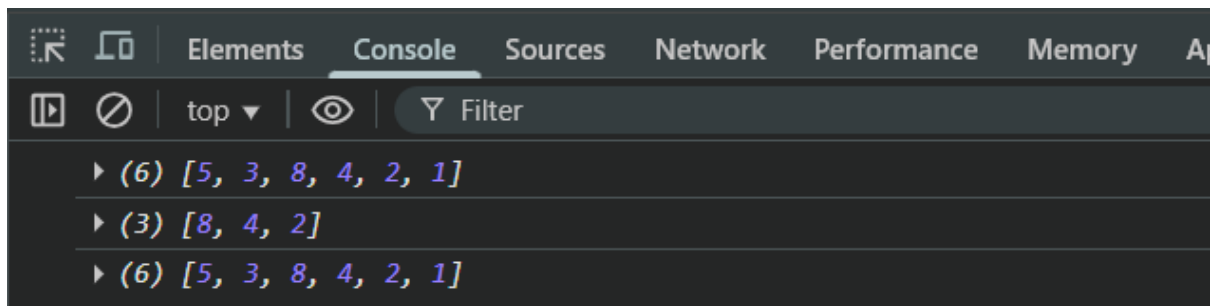


Example: use of slice()

Code:

```
99 //Use of slice function
100 let a = [5,3,8,4,2,1];
101 console.log(a); // [5,3,8,4,2,1]
102 let b = a.slice(2,5); //starting index, ending index(not included)
103 console.log(b); // [8,4,2]
104 console.log(a); // [5,3,8,4,2,1]
105 //Note: splice function modifies the original array, slice function does not modify the original array
106
```

Output:

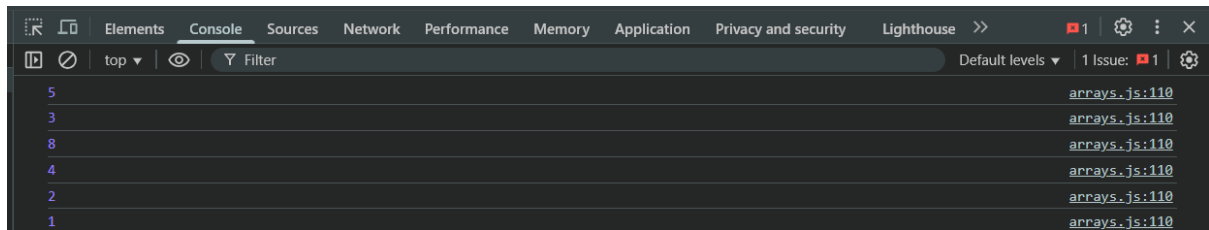


Example: printing the element of array

Code:

```
107 //Printing array using for loop
108 let a = [5,3,8,4,2,1];
109 for(let i=0; i<a.length; i++){
110     console.log(a[i]);
111 }
```

Output:

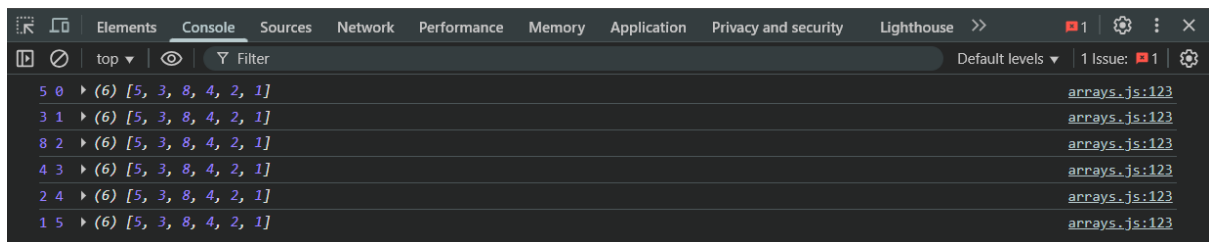


Example: printing the element of array using the for each and arrow

Code:

```
120 //for each loop using arrow function
121 let a = [5,3,8,4,2,1];
122 a.forEach((element, index, array) => {
123     console.log(element, index, array);
124 });
125
```

Output:

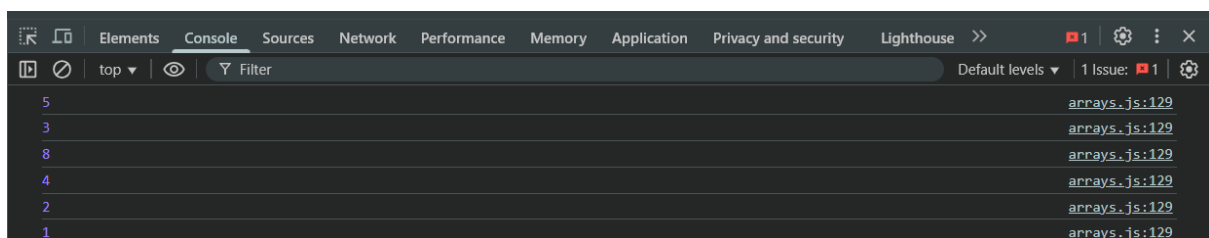


Example: printing the element of array using the for of loop

Code:

```
126 //for of loop
127 let a = [5,3,8,4,2,1];
128 for(let element of a){
129     console.log(element);
130 }
131 //Note: for of loop cannot access index and array, only element can be accessed
```

Output:

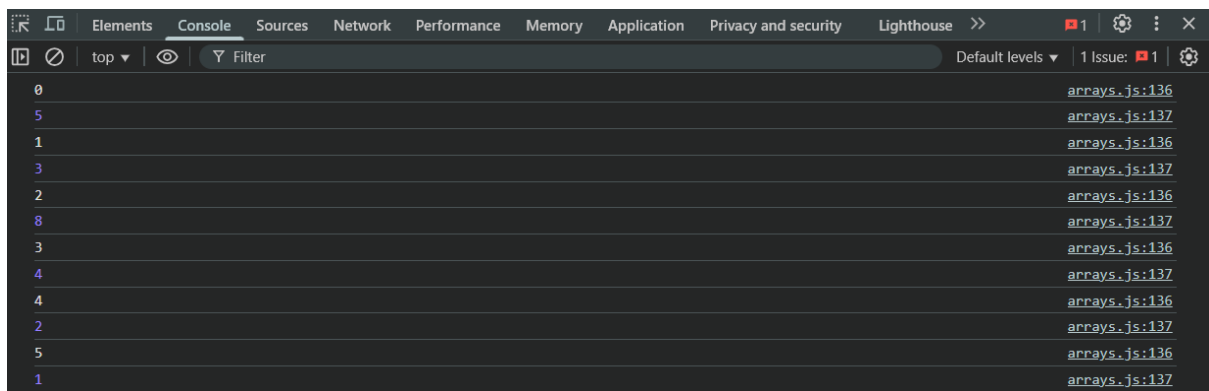


Example: use of for in loop

Code:

```
133 //for in loop
134 let a = [5,3,8,4,2,1];
135 for(let index in a){
136     console.log(index); //prints index
137     console.log(a[index]); //prints element
138 }
139 //Note: for in loop can access index but not element directly
140 //Note: for in loop is not recommended for arrays, it is used for objects
141 //Note: forEach loop is the best way to iterate an array
142
```

Output:



--The End--