

Arrays

Question 1: What is an array in JavaScript? How do you declare and initialize an array?

Ans.

An array can hold multiple data types, including numbers, strings, objects, or even other arrays.

Example: `let fruits = ["apple", "banana", "cherry"];`

Here:

- fruits is an array.
- It contains 3 elements: "apple", "banana", and "cherry".
- The index positions are:

`fruits[0] → "apple"`

`fruits[1] → "banana"`

`fruits[2] → "cherry"`

- How to Declare an Array :

There are two main ways:

1. Using Array Literal (Recommended) :

`let colors = ["red", "green", "blue"];`

2. Using the Array Constructor :

`let colors = new Array("red", "green", "blue");`

example :-

`let arr = new Array(3); // Creates an empty array with length 3, not [3]`

- How to Initialize an Array:

You can initialize an array with values:

```
let numbers = [1, 2, 3, 4, 5];
```

Or start empty and add items later:

```
let animals = [];
```

```
animals.push("dog");
```

```
animals.push("cat");
```

❖ Examples

- Mixed Data Types:

```
let mixed = [42, "hello", true, {name: "Alice"}, [1, 2, 3]];
```

- Accessing Elements:

```
console.log(mixed[1]); // "hello"
```

- Modifying Elements:

```
mixed[0] = 100;
```

```
console.log(mixed); // [100, "hello", true, {name: "Alice"}, [1, 2, 3]]
```

Question 2: Explain the methods `push()`, `pop()`, `shift()`, and `unshift()` used in arrays.

Ans.

The methods `push()`, `pop()`, `shift()`, and `unshift()` are fundamental array manipulation methods in JavaScript (and similar in other languages). They allow for adding or removing elements from either the beginning or the end of an array.

1. push()

- Purpose: Adds one or more elements to the end of an array.
- Return Value: The new length of the array.
- Example:

```
let fruits = ["apple", "banana"];  
fruits.push("orange", "grape");  
console.log(fruits); // Output: ["apple", "banana", "orange", "grape"]
```

2. pop()

- Purpose: Removes the last element from an array.
- Return Value: The removed element. If the array is empty, it returns undefined.
- Example:

```
let fruits = ["apple", "banana", "orange"];  
let removedFruit = fruits.pop();  
console.log(fruits); // Output: ["apple", "banana"]  
console.log(removedFruit)
```

3. shift()

- Purpose: Removes the first element from an array. This shifts all subsequent elements to a lower index.
- Return Value: The removed element. If the array is empty, it returns undefined.
- Example:

```
let fruits = ["apple", "banana", "orange"];  
let removedFruit = fruits.shift();  
console.log(fruits); // Output: ["banana", "orange"]  
console.log(removedFruit); // Output: "apple"
```

4. unshift()

- Purpose: Adds one or more elements to the beginning of an array.

This shifts existing elements to higher indices.

- Return Value: The new length of the array.

- Example:

```
let fruits = ["banana", "orange"];
```

```
fruits.unshift("apple", "grape");
```

```
console.log(fruits); // Output: ["apple", "grape", "banana", "orange"]
```
