1. **What does the `push()` method do?**
   - A) Removes the last element of an array
   - B) Adds one or more elements to the end of an array
   - C) Adds one or more elements to the beginning of an array
   - D) Removes the first element of an array

2. **Which method removes the last element from an array and returns that element?**
   - A) `shift()`
   - B) `unshift()`
   - C) `pop()`
   - D) `push()`

3. **What is the return value of the `push()` method?**
   - A) The length of the array after the new element is added
   - B) The element that was added
   - C) The array before the new element was added
   - D) `undefined`

4. **Which method adds one or more elements to the beginning of an array?**
   - A) `shift()`
   - B) `unshift()`
   - C) `pop()`
   - D) `slice()`

5. **What does the `shift()` method do?**

A) Removes the last element of an array

B) Adds an element to the beginning of an array

C) Removes the first element of an array

D) Adds an element to the end of an array

6. **Which method returns a shallow copy of a portion of an array into a new array?**

A) `splice()`

B) `slice()`

C) `pop()`

D) `shift()`

7. **How does the `splice()` method differ from `slice()`?**

A) `splice()` adds/removes elements; `slice()` only copies elements

B) `splice()` copies elements; `slice()` removes elements

C) `splice()` only removes elements; `slice()` adds elements

D) They perform the same function

8. **What is the return value of the `pop()` method?**

A) The last element of the array

B) The first element of the array

C) The modified array

D) The length of the array

9. **What does the `unshift()` method return?**

A) The new length of the array

B) The first element of the array

C) The new first element of the array

D) `undefined`

10. **If you want to remove 2 elements starting from index 3 in an array, which method would you use?**

A) `slice(3, 5)`

B) `splice(3, 2)`

C) `pop(3, 2)`

D) `shift(3, 2)`

11. **Which array method can be used to merge two or more arrays?**

A) `concat()`

B) `join()`

C) `slice()`

D) `splice()`

12. **What does the `reverse()` method do?**

A) Removes the last element of an array

B) Sorts the elements in ascending order

C) Reverses the order of elements in an array

D) Adds an element to the start of an array

13. **Which method can be used to convert an array to a string with a specified separator?**

A) `concat()`

B) `toString()`

C) `join()`

D) `slice()`

14. **What does the `fill()` method do?**
    - A) Fills all elements in an array with a static value
    - B) Adds elements to the start of an array
    - C) Adds elements to the end of an array
    - D) Removes elements from an array
15. **Which method removes and replaces elements in an array?**
    - A) `slice()`
    - B) `splice()`
    - C) `pop()`
    - D) `shift()`
16. **Which array method removes the first element and returns it?**
    - A) `push()`
    - B) `pop()`
    - C) `shift()`
    - D) `unshift()`
17. **What does the `toString()` method do?**
    - A) Converts an array to a string
    - B) Sorts the array alphabetically
    - C) Removes elements from an array
    - D) Fills the array with a value
18. **What happens when you call `slice(1, 4)` on an array?**
    - A) Returns the first four elements of the array

B) Removes the elements at indices 1 through 4

C) Returns a new array with elements from index 1 up to but not including index 4

D) Removes elements starting from index 4

19. **Which method would you use to add an item to a specific position in an array?**

A) `push()`

B) `splice()`

C) `unshift()`

D) `slice()`

20. **What is the return value of the `concat()` method?**

A) The length of the new array

B) The newly merged array

C) The number of arrays merged

D) The last element of the array

21. **Which method modifies the original array?**

A) `concat()`

B) `slice()`

C) `splice()`

D) `toString()`

22. **What does the `sort()` method return?**

A) A new sorted array

B) The original array sorted in place

C) The first element of the sorted array

D) A reversed array

23. **Which method can be used to create a new array that excludes a portion of the original array?**

- A) `splice()`
- B) `concat()`
- C) `slice()`
- D) `shift()`

24. **What does the `indexOf()` method return?**

- A) The last index of an element
- B) The index of the first occurrence of an element
- C) The length of the array
- D) The first element of the array

25. **Which array method can be used to remove duplicate values from an array?**

- A) `pop()`
- B) `filter()`
- C) `concat()`
- D) `sort()`

26. **Which method removes elements from an array and optionally inserts new elements in their place?**

- A) `push()`
- B) `splice()`
- C) `slice()`
- D) `shift()`

27. **What does the `lastIndexOf()` method return?**

-

A) The last element of an array

B) The index of the last occurrence of an element

C) The length of the array

D) A reversed array

28. **Which method flattens an array by a specified depth?**

A) `flat()`

B) `concat()`

C) `reverse()`

D) `pop()`

29. **What does the `isArray()` method check?**

A) If all elements in an array are of the same type

B) If a value is an array

C) If the array is empty

D) If all elements in an array are numbers

30. **What does the `copyWithin()` method do?**

A) Copies array elements within the array to another position

B) Removes elements from an array

C) Joins arrays together

D) Reverses the order of elements in an array

---

31. What does this code return?

```
let arr = [10, 20, 30, 40];
let result = arr.slice(1, 3);
console.log(result);
```

-

A) [10, 20]

B) [20, 30]

C) [10, 20, 30]

D) [30, 40]

---

32. What will be the output of this code?

```
let arr = [5, 10, 15, 20];
arr.splice(1, 2);
console.log(arr);
```

A) [5, 10, 15, 20]

B) [10, 15]

C) [5, 20]

D) [5, 15, 20]

---

33. What does the following code return?

```
let arr = [1, 2, 3, 4, 5];
let result = arr.indexOf(3);
console.log(result);
```

A) 1

B) 3

C) 2

D) 5

---

34. What will this code log?

```
let arr = ['a', 'b', 'c', 'd'];
arr.reverse();
console.log(arr);
```

A) ['d', 'c', 'b', 'a']

- B) ['a', 'b', 'c', 'd']
- C) ['a', 'c', 'b', 'd']
- D) undefined

---

35. What will be the result of this code?

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

- A) [1, 2, 3, 4]
- B) [1, 2]
- C) [3, 4]
- D) [2, 3]

---

36. What does this code return?

```
let arr = [10, 20, 30];
let result = arr.join('-');
console.log(result);
```

- A) 102030
- B) 10-20-30
- C) [10, 20, 30]
- D) [10-20-30]

---

37. What will be logged by this code?

```
let arr = [1, 2, 3];
arr.fill(0);
console.log(arr);
```

- A) [0, 2, 3]
- B) [0, 0, 0]
- C) [1, 2, 0]
- D) [1, 0, 3]

---

38. What is the output of the following code?

```
let arr = [100, 200, 300];
arr.copyWithin(1, 2);
console.log(arr);
```

- A) [100, 300, 300]
- B) [100, 200, 300]
- C) [100, 200, 200]
- D) [100, 100, 300]

---

39. What will the code return?

```
let arr = [1, 2, 3, 4];
let result = arr.includes(3);
console.log(result);
```

- A) true
- B) false
- C) 3
- D) [3]

---

40. What will this code produce?

```
let arr = [4, 3, 2, 1];
arr.sort();
console.log(arr);
```

- A) `[4, 3, 2, 1]`
- B) `[1, 2, 3, 4]`
- C) `[4, 2, 1, 3]`
- D) `[3, 1, 4, 2]`

---

41. What is the result of this code?

```
let arr = [1, 2, 3, 4];
arr.splice(2, 1, 5);
console.log(arr);
```

- A) `[1, 2, 3, 4]`
- B) `[1, 2, 5, 4]`
- C) `[5, 2, 3, 4]`
- D) `[1, 5, 3, 4]`

---

42. What will the following code output?

```
let arr = ['a', 'b', 'c'];
let result = arr.shift();
console.log(result);
```

- A) `['a', 'b']`
- B) `'a'`
- C) `'c'`
- D) `undefined`

---

43. What will the code return?

```
let arr = [10, 20, 30];
arr.unshift(5);
console.log(arr);
```

- A) `[5, 10, 20, 30]`
- B) `[10, 20, 30, 5]`
- C) `[10, 5, 20, 30]`
- D) `undefined`

---

44. What is the result of this code?

```
let arr = [1, 2, 3, 4, 5];
arr = arr.slice(2);
console.log(arr);
```

- A) `[3, 4, 5]`
- B) `[1, 2, 3, 4, 5]`
- C) `[2, 3]`
- D) `[1, 2]`

---

45. What will the following code produce?

```
let arr = [1, 2, 3];
let result = arr.find(element => element > 2);
console.log(result);
```

- A) `3`
- B) `2`
- C) `[3]`
- D) `[2, 3]`

---

46. What will be the output of the following code?

```
let arr = [1, 2, 3, 4, 5];
let result = arr.reverse().join('');
console.log(result);
```

- A) 54321
- B) [5, 4, 3, 2, 1]
- C) 5,4,3,2,1
- D) [54321]

---

47. What will be the result of this code?

```
let arr = ['apple', 'banana', 'cherry'];
let result = arr.indexOf('banana');
console.log(result);
```

- A) 2
- B) 1
- C) 'banana'
- D) ['apple', 'banana', 'cherry']

---

48. What is the output of this code?

```
let arr = [1, 2, 3, 4];
arr.pop();
console.log(arr.length);
```

- A) 3
- B) 4
- C) 2
- D) 1

---

49. What does the following code return?

```
let arr = [5, 10, 15, 20];
let result = arr.splice(1, 0, 25);
console.log(arr);
```

- A) `[5, 10, 15, 20]`
- B) `[5, 25, 10, 15, 20]`
- C) `[5, 10, 25, 15, 20]`
- D) `[5, 25, 15, 20]`

---

50. What will the following code output?

```
let arr = [100, 200, 300];
arr.push(400);
console.log(arr.length);
```

- A) 3
- B) 4
- C) 400
- D) 200

Here are 10 more complicated JavaScript array method questions:

---

51. What will be the output of this code?

```
let arr = [10, 20, 30, 40, 50];
let result = arr.splice(1, 2, 25, 35, 45);
console.log(arr, result);
```

- A) `[10, 25, 35, 45, 40, 50], [20, 30]`
- B) `[10, 25, 35, 45, 50], [20, 30]`
- C) `[10, 25, 35, 45], [20, 30, 40, 50]`
- D) `[25, 35, 45, 50], [20, 30, 40]`

---

52. What will the following code return?

```
let arr = [100, 200, 300, 400];
arr.copyWithin(1, 2, 4);
console.log(arr);
```

- A) [100, 300, 400, 400]
- B) [100, 300, 300, 400]
- C) [300, 100, 200, 400]
- D) [100, 100, 300, 400]

---

53. What is the output of the following code?

```
let arr = [10, 20, 30, 40, 50];
let result = arr.slice(2, -1).reverse();
console.log(result);
```

- A) [30, 20]
- B) [40, 30]
- C) [30, 40]
- D) [50, 40]

---

54. What will this code output?

```
let arr = [1, 2, 3, 4];
arr.fill(5, 1, 3);
console.log(arr);
```

- A) [5, 5, 5, 5]
- B) [1, 5, 5, 4]
- C) [5, 5, 3, 4]
- D) [1, 2, 5, 4]

---

55. What does this code return?

```
let arr = ['apple', 'banana', 'cherry'];
let result = arr.reverse().shift();
console.log(result, arr);
```

- A) cherry, ['apple', 'banana']
- B) cherry, ['banana', 'apple']
- C) apple, ['cherry', 'banana']
- D) banana, ['cherry', 'apple']

---

56. What is the result of the following code?

```
let arr = [5, 10, 15, 20, 25];
let result = arr.splice(-3, 2, 50, 60);
console.log(arr);
```

- A) [5, 10, 50, 60, 25]
- B) [5, 10, 15, 20, 60]
- C) [5, 50, 60, 25]
- D) [50, 60, 15, 20, 25]

---

57. What will the following code produce?

```
let arr = [10, 20, 30];
arr.reverse().push(40, 50);
console.log(arr);
```

- A) [30, 20, 10, 40, 50]
- B) [40, 50, 30, 20, 10]
- C) [50, 40, 10, 20, 30]
- D) [10, 20, 30, 50, 40]
