JavaScript 50 Q&A: Functions, Arguments, Scope, and Hoisting

Q1 [Interview] What is a function in JavaScript?

Ans: A block of code designed to perform a task.

```
function greet() { console.log('Hello'); }
```

Q2 [Interview] How to define a function in JS?

Ans: Using function keyword or arrow function.

```
function add(a, b) { return a + b; }
```

Q3 [Interview] What are arguments and parameters?

Ans: Parameters are placeholders, arguments are actual values.

Q4 [Interview] What is the return statement?

Ans: Returns a value from the function.

Q5 [Interview] What is a function expression?

Ans: A function assigned to a variable.

```
const greet = function() { return 'Hi'; };
```

Q6 [Interview] What is a callback function?

Ans: A function passed as an argument to another function.

Q7 [Interview] What is an arrow function?

Ans: Short syntax for defining functions.

```
const sum = (a, b) => a + b;
```

Q8 [Interview] What is function hoisting?

Ans: Functions declared with `function` are hoisted to the top.

Q9 [Interview] Can functions be nested?

Ans: Yes, you can define functions inside functions.

Q10 [Interview] What is the arguments object?

Ans: Array-like object containing arguments passed to function.

Q11 [Interview] What is the scope of a function?

Ans: Scope defines where variables are accessible.

Q12 [Interview] What is block scope?

Ans: Variables declared with let/const are block-scoped.

Q13 [Interview] What is global scope?

Ans: Accessible throughout the JS file.

Q14 [Interview] What is lexical scope?

Ans: Child functions can access parent variables.

Q15 [Interview] What is a pure function?

Ans: A function that returns same output for same input.

Q16 [DSA] Write a function to reverse a string.

Ans: Use loop or built-in methods.

```
function reverse(str) { return str.split('').reverse().join(''); }
```

Q17 [DSA] Check if a number is even.

Ans: Use modulus inside function.

```
function isEven(n) { return n % 2 === 0; }
```

Q18 [DSA] Sum of digits of a number.

Ans: Use loop with % and /.

```
function sumDigits(n) { let sum = 0; while(n) { sum += n % 10; n = Math.floor(n / 10); } return sum; }
```

Q19 [DSA] Check for palindrome string.

Ans: Compare string with reversed.

```
function isPalindrome(s) { return s === s.split('').reverse().join(''); }
```

Q20 [DSA] Factorial using recursion.

Ans: Base case and recursive call.

```
function fact(n) { return n <= 1 ? 1 : n * fact(n - 1); }
```

Q21 [DSA] Find largest number in array.

Ans: Loop through array.

```
function max(arr) { return Math.max(...arr); }
```

Q22 [DSA] Count vowels in string.

Ans: Loop and match vowels.

```
function countVowels(str) { return (str.match(/[aeiou]/gi) || []).length; }
```

Q23 [DSA] Return array sum.

Ans: Use reduce.

```
function sum(arr) { return arr.reduce((a,b)=>a+b, 0); }
```

Q24 [DSA] Find GCD of two numbers.

Ans: Use Euclid's algorithm.

```
function gcd(a, b) { return b ? gcd(b, a % b) : a; }
```

Q25 [DSA] Fibonacci series using function.

Ans: Use loop or recursion.

Q26 [DSA] Convert temperature F to C.

Ans: Simple conversion.

```
function toCelsius(f) { return (f - 32) * 5 / 9; }
```

Q27 [DSA] Capitalize first letter of each word.

Ans: Split, map, join.

Q28 [DSA] Check if year is leap year.

Ans: Divisible by 4 and not 100 unless 400.

Q29 [DSA] Sort array ascending.

Ans: Use sort().

```
arr.sort((a, b) => a - b);
```

Q30 [DSA] Merge two arrays.

Ans: Use concat or spread.

```
[...a, ...b];
```

Q31 [LeetCode] typeof function(){}?

Ans: 'function'

Q32 [LeetCode] function foo(){}; foo.name?

Ans: 'foo'

Q33 [LeetCode] arguments.length inside function?

Ans: Number of arguments passed.

Q34 [LeetCode] What is hoisted: var or let?

Ans: Only var and function declarations.

Q35 [LeetCode] Arrow function has 'this'?

Ans: No, uses lexical this.

Q36 [LeetCode] Can we call function before its declaration?

Ans: Yes if it's a function declaration.

Q37 [LeetCode] (() => 5)();

Ans: Returns 5

Q38 [LeetCode] arguments[0] in arrow function?

Ans: undefined

Q39 [LeetCode] Is 'function' a type?

Ans: Yes, typeof gives 'function'

Q40 [LeetCode] What is output: return; 5;

Ans: Returns undefined

Q41 [Conceptual] What is variable hoisting?

Ans: Variables declared with var are hoisted.

Q42 [Conceptual] Can you re-declare a function?

Ans: Yes, last wins.

Q43 [Conceptual] What happens if no return?

Ans: Returns undefined.

Q44 [Conceptual] Do arrow functions bind their own this?

Ans: No.

Q45 [Conceptual] Can you return object from arrow func?

Ans: Yes, use () to wrap object.

Q46 [Conceptual] Difference between declaration and expression?

Ans: Declaration is hoisted.

Q47 [Conceptual] Does scope affect closures?

Ans: Yes.

Q48 [Conceptual] Can you access function before it's defined?

Ans: Yes, if declaration.

Q49 [Conceptual] Is setTimeout callback a function?

Ans: Yes.

Q50 [Conceptual] Can a function return a function?

Ans: Yes.