JavaScript MCQs (Based on Chapter 1.3 & 1.4)

1. Which keyword is used to declare a variable in modern JavaScript?
a) var
b) let
c) define
d) int
Answer: b) let

2. What is the value of `x` after executing:

```
let x;
x = 0;
```

a) null
b) undefined
c) 0
d) false
Answer: c) 0

3. Which of the following is NOT a valid JavaScript value type?
a) Number
b) String
c) Boolean
d) Character
Answer: d) Character

4. Which of the following represents a Boolean value in JavaScript?
a) "true"
b) true
c) 'false'
d) "Boolean"
Answer: b) true

5. What does `null` represent in JavaScript?
a) Zero
b) Undefined
c) No value
d) Empty string
Answer: c) No value

6. Which symbol is used for single-line comments in JavaScript?
a) <!--
b) #
c) //
d) /* */
Answer: c) //

7. How do you access an object property using dot notation?

```
book.topic
```

a) book("topic")
b) book.topic
c) book->topic
d) book["topic"]
Answer: b) book.topic

8. Which operator was introduced in ES2020 for safe property access?
a) ??
b) ?.
c) ::
d) -->
Answer: b) ?.

9. What does the following return?

```
let primes = [2, 3, 5, 7];
primes.length
```

a) 3
b) 4
c) 7
d) undefined
Answer: b) 4

10. What is the last element of `primes`?

```
primes[primes.length - 1]
```

a) 5
b) 6
c) 7
d) undefined
Answer: c) 7

11. Which of the following creates an empty array?
a) let arr = new Array();
b) let arr = [];
c) Both a and b
d) None
Answer: c) Both a and b

12. Arrays in JavaScript can contain:
a) Only numbers

b) Only strings
c) Arrays and objects
d) Only Booleans
Answer: c) Arrays and objects

13. Which of the following is an object initializer expression?
a) []
b) {}
c) ""
d) ()
Answer: b) {}

14. What does `"3" + "2"` evaluate to?
a) 5
b) 32
c) "5"
d) Error
Answer: b) "32"

15. Which shorthand operator increments a variable?
a) ++
b) +=
c) --
d) **
Answer: a) ++

16. Which operator tests for strict equality in JavaScript?
a) =
b) ==
c) ===
d) :=
Answer: c) ===

17. What is the result of:

```
"two" > "three"
```

a) true
b) false
c) undefined
d) error
Answer: a) true

18. Which logical operator represents OR in JavaScript?
a) &
b) &&

c) ||
d) or
Answer: c) ||

19. What does the ! operator do?
a) Bitwise NOT
b) Logical inversion
c) Arithmetic negation
d) Modulus
Answer: b) Logical inversion

20. What is the difference between expressions and statements?
a) Expressions change state; statements compute values
b) Expressions compute values; statements alter state
c) Both are same
d) Expressions only exist in functions
Answer: b) Expressions compute values; statements alter state

21. Which statement declares a function in JavaScript?
a) func myFunc()
b) define myFunc()
c) function myFunc()
d) fn myFunc()
Answer: c) function myFunc()

22. What does the following function return?

```
function plus1(x) { return x + 1; }
plus1(3);
```

a) 3
b) 4
c) undefined
d) Error
Answer: b) 4

23. Functions in JavaScript can be:
a) Assigned to variables
b) Passed as arguments
c) Returned from functions
d) All of the above
Answer: d) All of the above

24. Which syntax is used for arrow functions?
a) ->
b) =>
c) :=

d) ::
Answer: b) =>

25. What is the correct arrow function for squaring `x`?
a) const square = (x) => x * x;
b) const square = x -> x * x;
c) const square = => x * x;
d) const square(x) = x * x;
Answer: a) const square = (x) => x * x;

26. When a function is assigned to an object property, it is called a:
a) Variable
b) Method
c) Function-object
d) Constructor
Answer: b) Method

27. Which keyword refers to the current object inside a method?
a) self
b) current
c) this
d) obj
Answer: c) this

28. What does `a.push(1,2,3)` do?
a) Removes elements
b) Adds elements to end of array
c) Adds elements to start of array
d) Reverses array
Answer: b) Adds elements to end of array

29. Which method reverses array order?
a) flip()
b) reverse()
c) invert()
d) swap()
Answer: b) reverse()

30. What does the `if...else` statement do?
a) Always executes both branches
b) Chooses between two code paths
c) Loops until condition is false
d) Declares variables
Answer: b) Chooses between two code paths

31. What will `abs(-10)` return?

```
function abs(x) { if (x>=0) return x; else return -x; }
```

a) -10
b) 10
c) 0
d) undefined
Answer: b) 10

32. Which loop is used in the code `for(let x of array)`?
a) for/in
b) for/of
c) while
d) do/while
Answer: b) for/of

33. What does `sum([2,3,5])` return?
a) 10
b) 9
c) 0
d) Error
Answer: a) 10

34. Which loop executes while a condition is true?
a) for/of
b) while
c) do/for
d) repeat
Answer: b) while

35. What does `factorial(4)` return?
a) 6
b) 12
c) 24
d) 120
Answer: c) 24

36. What does `factorial2(5)` return?
a) 24
b) 60
c) 120
d) 720
Answer: c) 120

37. In object-oriented JavaScript, how are classes defined?
a) function ClassName() {}
b) class ClassName {}

c) object ClassName {}
d) prototype ClassName {}
Answer: b) class ClassName {}

38. Which method is used to initialize objects in a class?
a) init()
b) start()
c) constructor()
d) create()
Answer: c) constructor()

39. What does `new Point(1,1)` create?
a) Function
b) Array
c) Instance of Point
d) Object literal
Answer: c) Instance of Point

40. What does the `distance()` method of class Point return?
a) Distance from origin
b) Distance between two points
c) Distance from (0,1)
d) Undefined
Answer: a) Distance from origin

41. Which JavaScript class was extended in the histogram example?
a) Object
b) Set
c) Map
d) Array
Answer: c) Map

42. What does the `DefaultMap` class override?
a) set()
b) get()
c) delete()
d) has()
Answer: b) get()

43. What default value does `DefaultMap` return if key not found?
a) undefined
b) null
c) 0
d) false
Answer: c) 0

44. In the histogram example, what is stored in `letterCounts`?
a) Strings only
b) Map of letters to counts
c) Array of letters
d) Objects with text
Answer: b) Map of letters to counts

45. Which function converts text to uppercase and removes whitespace?
a) replace(/\s/g,"").toUpperCase()
b) trim().toUpperCase()
c) clean().upper()
d) filter().upper()
Answer: a) replace(/\s/g,"").toUpperCase()

46. Which method was used to sort the histogram entries?
a) order()
b) sort()
c) arrange()
d) sequence()
Answer: b) sort()

47. Which built-in method repeats a string multiple times?
a) repeat()
b) clone()
c) times()
d) copy()
Answer: a) repeat()

48. In the histogram, entries with less than what percentage are dropped?
a) 5%
b) 2%
c) 1%
d) 0.5%
Answer: c) 1%

49. What is the role of `process.stdin.setEncoding("utf-8")`?
a) Reads bytes
b) Reads Unicode strings
c) Reads JSON
d) Reads binary only
Answer: b) Reads Unicode strings

50. Which Node.js version is required for the histogram program?
a) Node 8
b) Node 10
c) Node 12

d) Node 14
Answer: c) Node 12

MCQs on JavaScript Lexical Structure

1. JavaScript is a _____ language.
a) Case-sensitive
b) Case-insensitive
c) Partially case-sensitive
d) None of these
Answer: a) Case-sensitive

2. Which of the following is **not** a valid JavaScript identifier?
a) myVar
b) _dummy
c) 1variable
d) $str
Answer: c) 1variable

3. Which keyword must be typed exactly in lowercase?
a) While
b) WHILE
c) while
d) Any of the above
Answer: c) while

4. Which of the following are considered whitespace in JavaScript?
a) Space character (\u0020)
b) Tabs
c) Unicode spaces
d) All of the above
Answer: d) All of the above

5. JavaScript ignores line breaks except in some cases related to _____.
a) Comments
b) Optional semicolons
c) Reserved words
d) Functions
Answer: b) Optional semicolons

6. A single-line comment in JavaScript starts with:
a) #
b) //

c) /*
d) <!--
Answer: b) //

7. A multi-line comment in JavaScript starts and ends with:
a) # … #
b) // … //
c) /* … /
d) <! ... !>
Answer: *c) /* … */

8. Which of the following is **not** a JavaScript literal?
a) "hello"
b) 1.2
c) null
d) var
Answer: d) var

9. Which of the following is a Boolean literal in JavaScript?
a) yes
b) no
c) true
d) on
Answer: c) true

10. Identifiers in JavaScript can begin with:
a) A digit
b) A letter, underscore, or dollar sign
c) Only a letter
d) Only underscore
Answer: b) A letter, underscore, or dollar sign

11. Which of the following is a reserved keyword in JavaScript?
a) myVar
b) const
c) _dummy
d) $str
Answer: b) const

12. Which of the following keywords is used in ES6 for variable declaration?
a) let
b) var
c) const
d) All of the above
Answer: d) All of the above

13. JavaScript programs are written using the _____ character set.
a) ASCII
b) Unicode
c) UTF-16 only
d) ANSI
Answer: b) Unicode

14. Which Unicode escape correctly represents "é"?
a) \u00E9
b) \uE900
c) \u1234
d) \xE9
Answer: a) \u00E9

15. Which version of JavaScript introduced curly braces `{ }` in Unicode escapes?
a) ES3
b) ES5
c) ES6
d) ES7
Answer: c) ES6

16. What will `console.log("\u{1F600}");` print?
a) A number
b) Error
c) A smiley face emoji 😀
d) A blank space
Answer: c) A smiley face emoji 😀

17. Which of the following characters **cannot** start a JavaScript identifier?
a) $
b) _
c) A digit (0–9)
d) A letter
Answer: c) A digit (0–9)

18. Which word is not fully reserved in JavaScript to ensure backward compatibility?
a) let
b) while
c) for
d) null
Answer: a) let

19. The identifiers `online`, `Online`, and `ONLINE` in JavaScript are:
a) Same variable
b) Treated as distinct variables
c) Invalid

d) Case-insensitive
Answer: b) Treated as distinct variables

20. Which of the following is **not** a valid reserved keyword in JavaScript?
a) enum
b) package
c) super
d) target
Answer: d) target

21. Which of the following words should you **avoid using as identifiers** because they have special restrictions?
a) eval and arguments
b) class and var
c) let and const
d) null and true
Answer: a) eval and arguments

22. Which of the following is **not allowed** in JavaScript identifiers?
a) Unicode letters
b) Digits after the first character
c) Emojis
d) Dollar sign ($)
Answer: c) Emojis

23. JavaScript assumes the source code it interprets has already been _____.
a) Escaped
b) Normalized
c) Parsed
d) Encoded in ASCII
Answer: b) Normalized

24. Which statement separator is commonly used in JavaScript?
a) Colon (:)
b) Semicolon (;)
c) Comma (,)
d) Space
Answer: b) Semicolon (;)

25. JavaScript allows omitting semicolons in some cases due to:
a) Automatic Semicolon Insertion (ASI)
b) Compiler rules
c) Comments
d) Reserved keywords
Answer: a) Automatic Semicolon Insertion (ASI)

26. Which of the following may cause ambiguity if written without a semicolon?
a) return
b) (…)
c) […]
d) All of the above
Answer: d) All of the above

27. Which operator requires being on the **same line** as its operand if used as a postfix?
a) ++ and --
b) + and -
c) * and /
d) =
Answer: a) ++ and --

28. Which statement is always followed by a semicolon if a newline comes after it?
a) var
b) break
c) let
d) const
Answer: b) break

29. What is a "defensive semicolon" in JavaScript?
a) Extra semicolon used to avoid syntax errors when a line begins with (or [
b) A semicolon at the end of every statement
c) A semicolon inside comments
d) Deprecated semicolon usage
Answer: a) Extra semicolon used to avoid syntax errors when a line begins with (or [

30. Which is the safest style for writing semicolons in JavaScript?
a) Omit them always
b) Use them only in rare cases
c) Explicitly mark ends of statements with semicolons
d) Never use semicolons at all
Answer: c) Explicitly mark ends of statements with semicolons

---

MCQs: Variable Declaration and Assignment (50 Questions)

1. In JavaScript, assigning a value to a name is called:
a) Binding
b) Referencing
c) Declaring
d) Hoisting
Answer: a) Binding

2. A variable's value in JavaScript can:
a) Never change
b) Change during execution
c) Only be numeric
d) Only be constant
Answer: b) Change during execution

3. A permanently assigned name in JavaScript is called:
a) Identifier
b) Constant
c) Variable
d) Literal
Answer: b) Constant

4. In ES6, variables are declared using:
a) var
b) let and const
c) global
d) declare
Answer: b) let and const

5. Before ES6, variables were declared using:
a) val
b) var
c) const
d) declare
Answer: b) var

6. Which keyword allows multiple variables to be declared in one statement?
a) const
b) var
c) let
d) Both var and let
Answer: d) Both var and let

7. If no initial value is assigned with `let`, the variable's value is:
a) null
b) undefined
c) 0
d) false
Answer: b) undefined

8. Which keyword requires initialization at declaration?
a) var
b) let
c) const

d) assign
Answer: c) const

9. Attempting to reassign a `const` variable results in:
a) SyntaxError
b) ReferenceError
c) TypeError
d) No error
Answer: c) TypeError

10. Constants are often written in:
a) CamelCase
b) snake_case
c) ALL_CAPS
d) kebab-case
Answer: c) ALL_CAPS

11. One programming style recommends declaring everything with:
a) var
b) let
c) const
d) auto
Answer: c) const

12. Loop variables in `for` loops are commonly declared with:
a) var
b) let
c) const
d) both let and const
Answer: d) both let and const

13. Declaring loop variables with `const` works if:
a) They are reassigned each iteration
b) They remain constant within each iteration
c) They are undefined
d) They are global
Answer: b) They remain constant within each iteration

14. Variables declared with `let` and `const` are:
a) Globally scoped
b) Function scoped
c) Block scoped
d) Class scoped only
Answer: c) Block scoped

15. Block scope means a variable is available:
a) Everywhere in the program
b) Only inside the curly braces `{ }` it is declared in
c) Only in functions
d) Only in loops
Answer: b) Only inside the curly braces `{ }` it is declared in

16. A top-level variable declared outside blocks has:
a) Local scope
b) Block scope
c) Global scope
d) Module scope only
Answer: c) Global scope

17. In Node.js, global scope means:
a) Across all files
b) Within the same file
c) Inside all functions
d) Inside all loops
Answer: b) Within the same file

18. In client-side JavaScript, global scope means:
a) Across the document
b) Within the same script only
c) Limited to functions
d) Limited to classes
Answer: a) Across the document

19. Declaring the same variable twice with `let` or `const` in the same scope:
a) Is legal
b) Is ignored
c) Causes a syntax error
d) Overwrites the old value
Answer: c) Causes a syntax error

20. Declaring the same variable in a nested block:
a) Is legal and creates a new variable
b) Is illegal
c) Overwrites the global variable
d) Throws a ReferenceError
Answer: a) Is legal and creates a new variable

21. Redeclaring `const x = 1; let x = 3;` in the same scope results in:
a) Value updated to 3
b) ReferenceError
c) SyntaxError

d) No error
Answer: c) SyntaxError

22. JavaScript variables:
a) Must have a fixed type
b) Can hold any type
c) Must be numbers only
d) Must be strings only
Answer: b) Can hold any type

23. Assigning a number and later a string to the same variable is:
a) Illegal
b) Allowed but bad style
c) Always recommended
d) Causes TypeError
Answer: b) Allowed but bad style

24. Variables declared with `var`:
a) Are block scoped
b) Are function scoped
c) Are module scoped
d) Cannot be redeclared
Answer: b) Are function scoped

25. Declaring `var` outside a function creates:
a) Local variable
b) Block variable
c) Global variable
d) Constant
Answer: c) Global variable

26. Global variables declared with `var` are:
a) Properties of the global object
b) Private variables
c) Immutable
d) Deleted automatically
Answer: a) Properties of the global object

27. Global variables declared with `let` or `const`:
a) Are properties of the global object
b) Are not properties of the global object
c) Can be deleted
d) Are function scoped
Answer: b) Are not properties of the global object

28. Redeclaring a variable with `var` in the same function is:
a) Illegal
b) Legal and common
c) Causes ReferenceError
d) Reassigns as const
Answer: b) Legal and common

29. `var` declarations allow:
a) Hoisting
b) Block scope
c) Const initialization
d) Type enforcement
Answer: a) Hoisting

30. Hoisting means:
a) Variables are initialized at top
b) Variable declarations are moved to the top
c) Values are automatically assigned
d) Constants become variables
Answer: b) Variable declarations are moved to the top

31. In `var`, the declaration is hoisted but:
a) Initialization is not
b) Value is constant
c) Scope is ignored
d) Block scope applies
Answer: a) Initialization is not

32. Using a `var` variable before initialization gives:
a) Error
b) null
c) undefined
d) ReferenceError
Answer: c) undefined

33. Using a `let` variable before declaration gives:
a) undefined
b) TypeError
c) ReferenceError
d) null
Answer: c) ReferenceError

34. Using undeclared variables in strict mode gives:
a) undefined
b) ReferenceError
c) SyntaxError

d) TypeError
Answer: b) ReferenceError

35. Using undeclared variables outside strict mode creates:
a) Local variable
b) Constant
c) Global variable
d) Block scoped variable
Answer: c) Global variable

36. Accidental globals created by assignment without `let/const/var` are:
a) Properties of the global object
b) Immutable
c) Function scoped
d) Deleted automatically
Answer: a) Properties of the global object

37. Accidental globals can be deleted with:
a) remove()
b) delete operator
c) clear()
d) reset()
Answer: b) delete operator

38. Destructuring assignment is introduced in:
a) ES3
b) ES5
c) ES6
d) ES7
Answer: c) ES6

39. Destructuring assignment works on:
a) Arrays only
b) Objects only
c) Arrays and objects
d) Functions only
Answer: c) Arrays and objects

40. `let [x, y] = [1, 2];` assigns:
a) x=2, y=1
b) x=1, y=2
c) x=undefined, y=2
d) x=1, y=undefined
Answer: b) x=1, y=2

41. `[, x, , y] = [1,2,3,4];` assigns:
a) x=1, y=3
b) x=2, y=4
c) x=undefined, y=2
d) x=3, y=4
Answer: b) x=2, y=4

42. Using `...` in array destructuring collects:
a) First element
b) All elements
c) Remaining elements
d) Ignored values
Answer: c) Remaining elements

43. `let [first, ...rest] = "Hello";` gives:
a) first="H", rest=["e","l","l","o"]
b) first="Hello", rest=[]
c) first="He", rest=["llo"]
d) Error
Answer: a) first="H", rest=["e","l","l","o"]

44. Object destructuring `let {r,g,b} = color;` assigns:
a) Values of r,g,b properties
b) Keys of r,g,b properties
c) Null values
d) String values only
Answer: a) Values of r,g,b properties

45. In object destructuring, variables must:
a) Always match property names
b) Can be renamed using `:`
c) Must be constants
d) Cannot be nested
Answer: b) Can be renamed using `:`

46. `const { cos: cosine } = Math;` means:
a) Copies cos to cosine
b) Assigns cosine=undefined
c) Error
d) cos=cosine
Answer: a) Copies cos to cosine

47. Extra variables in destructuring become:
a) Deleted
b) Ignored
c) undefined

d) Errors
Answer: c) undefined

48. Destructuring supports:
a) Nested arrays
b) Nested objects
c) Objects of arrays and arrays of objects
d) All of the above
Answer: d) All of the above

49. Destructuring works on:
a) Only arrays
b) Any iterable
c) Only objects
d) Only functions
Answer: b) Any iterable

50. Complex destructuring can be validated by:
a) Reversing assignment (using LHS on RHS)
b) Declaring as var
c) Printing variable types
d) Using strict mode
Answer: a) Reversing assignment (using LHS on RHS)

📓 50 Important MCQs from the Text

1. `eval()` in JavaScript is:
a) An operator
b) A reserved keyword
c) A function
d) A method of Object

Answer: c) A function

---

2. Why is `eval()` often treated like an operator?
a) Because it can only be used inside loops
b) Because it shares variable environment of the caller
c) Because it has strict operator precedence
d) Because it always returns boolean

Answer: b) Because it shares variable environment of the caller

---

3. Which of the following prevents aggressive optimization in JavaScript?
a) typeof
b) eval()
c) delete
d) void

Answer: b) eval()

---

4. What happens if `eval()` is given a non-string argument?
a) Throws SyntaxError
b) Returns `undefined`
c) Returns the same value
d) Returns null

Answer: c) Returns the same value

---

5. `eval("x=1")` inside a function will:
a) Create a global variable `x`
b) Modify local variable `x` if it exists
c) Always throw error
d) Do nothing

Answer: b) Modify local variable `x` if it exists

---

6. Which declaration inside `eval()` does **not** affect the caller's scope?
a) var
b) function
c) let
d) assignment

Answer: c) let

---

7. Code `eval("return;")` results in:
a) Returns undefined
b) Works as normal return

c) SyntaxError
d) Creates a function return

Answer: c) SyntaxError

---

8. An indirect call to `eval()` uses:
a) Local scope
b) Global scope
c) Block scope
d) Function parameters only

Answer: b) Global scope

---

9. What is a “direct eval”?
a) Using eval with alias name
b) Using eval with qualified object name
c) Using the unqualified identifier `eval`
d) Using eval inside strict mode

Answer: c) Using the unqualified identifier `eval`

---

10. In strict mode, `eval()` creates variables:
a) In global scope only
b) In its own private scope
c) In caller’s scope always
d) As global constants

Answer: b) In its own private scope

---

11. Which of the following is **not allowed** in strict mode?
a) Overwriting eval
b) Using eval() for global scope
c) Assigning variables in eval
d) Declaring with let inside eval

Answer: a) Overwriting eval

---

12. Conditional operator `?:` is also called:
a) Unary operator
b) Nullish operator
c) Ternary operator
d) Assignment operator

Answer: c) Ternary operator

---

13. Syntax of conditional operator is:
a) condition ? expr1 : expr2
b) condition : expr1 ? expr2
c) expr1 ? expr2 : condition
d) ? expr1 : expr2 ? condition

Answer: a) condition ? expr1 : expr2

---

14. Which operator provides a compact replacement for `if-else`?
a) ??
b) ?:
c) &&
d) ||

Answer: b) ?:

---

15. First-defined operator is represented by:
a) ??
b) ||
c) &&
d) ?:

Answer: a) ??

---

16. The `??` operator checks for:
a) Falsy values
b) Nullish values (null or undefined)
c) Boolean values only
d) Zero and empty strings

Answer: b) Nullish values

---

17. `0 ?? 100` evaluates to:
a) 100
b) undefined
c) 0
d) null

Answer: c) 0

---

18. Which operator is short-circuiting?
a) typeof
b) delete
c) ??
d) void

Answer: c) ??

---

19. Which operator requires parentheses when mixed with `||` or `&&`?
a) typeof
b) ??
c) delete
d) void

Answer: b) ??

---

20. The `typeof` operator returns:
a) Type of object as a function call
b) Always “object”
c) A string describing operand type
d) Boolean true or false

Answer: c) A string describing operand type

---

21. `typeof null` returns:
a) null

b) undefined
c) object
d) nullish

Answer: c) object

---

22. `typeof function() {}` returns:
a) object
b) function
c) method
d) undefined

Answer: b) function

---

23. Which operator removes a property from an object?
a) typeof
b) delete
c) void
d) ??

Answer: b) delete

---

24. After `delete o.x`, what happens?
a) o.x becomes undefined but exists
b) o.x is permanently removed
c) Throws error
d) Object is destroyed

Answer: b) o.x is permanently removed

---

25. Deleting an array element:
a) Decreases array length
b) Leaves a hole (sparse array)
c) Resets all values to null
d) Throws SyntaxError

Answer: b) Leaves a hole (sparse array)

---

26. In strict mode, `delete variableName;` causes:
a) Returns false
b) SyntaxError
c) Deletes variable
d) Does nothing

Answer: b) SyntaxError

---

27. Which property type cannot be deleted?
a) Configurable
b) Non-configurable
c) Writable
d) Enumerable

Answer: b) Non-configurable

---

28. The `await` operator works only inside:
a) Normal functions
b) Loops
c) async functions
d) eval()

Answer: c) async functions

---

29. The return value of `await` is:
a) A Promise object itself
b) The rejection reason
c) Fulfillment value of Promise
d) Always undefined

Answer: c) Fulfillment value of Promise

---

30. Which operator always returns `undefined`?
a) void
b) delete

c) typeof
d) ??

Answer: a) void

---

31. Example `void counter++` will:
a) Increment counter but return undefined
b) Return counter's new value
c) Return counter's old value
d) Throw error

Answer: a) Increment counter but return undefined

---

32. The comma operator returns:
a) Left operand
b) Right operand
c) Both values
d) Null

Answer: b) Right operand

---

33. Typical use of comma operator is in:
a) eval()
b) async functions
c) for loops with multiple variables
d) typeof expressions

Answer: c) for loops with multiple variables

---

34. Which operator is least commonly used?
a) delete
b) typeof
c) void
d) ??

Answer: c) void

---

35. `typeof NaN` returns:
a) number
b) NaN
c) undefined
d) float

Answer: a) number

---

36. Which operator is known as “nullish coalescing”?
a) &&
b) ??
c) ?:
d) void

Answer: b) ??

---

37. Which operator treats `0`, `""`, and `false` as valid values?
a) ||
b) &&
c) ??
d) ?:

Answer: c) ??

---

38. Which operator can stop optimizations in JS engines?
a) delete
b) eval()
c) void
d) typeof

Answer: b) eval()

---

39. `typeof []` returns:
a) array
b) object
c) list
d) function

Answer: b) object

---

40. Which operator allows executing dynamic code strings?
a) delete
b) typeof
c) eval()
d) void

Answer: c) eval()

---

41. Which operator can create “holes” in arrays?
a) typeof
b) delete
c) void
d) eval()

Answer: b) delete

---

42. Which of the following is a ternary operator?
a) ?:
b) ??
c) &&
d) ||

Answer: a) ?:

---

43. Which operator cannot be overwritten in strict mode?
a) delete
b) eval
c) typeof
d) void

Answer: b) eval

---

44. What does `typeof BigInt(10)` return?
a) number

b) bigint
c) string
d) object

Answer: b) bigint

---

45. `delete 1;` will:
a) Delete the number 1 from memory
b) Return true but do nothing
c) Throw SyntaxError
d) Return false

Answer: b) Return true but do nothing

---

46. Which operator discards the operand value but keeps side effects?
a) delete
b) void
c) typeof
d) ??

Answer: b) void

---

47. Which operator checks if a property exists?
a) in
b) delete
c) void
d) typeof

Answer: a) in

---

48. The conditional operator can replace:
a) for loops
b) if-else
c) switch-case
d) while loops

Answer: b) if-else

---

49. Which operator cannot return both operands?
a) , (comma)
b) ??
c) &&
d) void

Answer: d) void

---

50. Which operator was introduced in ES2020?
a) ??
b) void
c) delete
d) eval

Answer: a) ??

📘 30 Important MCQs on JavaScript Statements

1. **Which statement allows properties of an object to be treated as variables within a block?**
   a) for/in
   b) with ✅
   c) use strict
   d) import
2. **Why is the `with` statement discouraged in JavaScript?**
   a) It increases memory usage
   b) It makes debugging harder
   c) It slows optimization and execution ✅
   d) It requires strict mode
3. **What happens if you declare a variable inside a `with` block using `let`?**
   a) It adds a property to the object
   b) It becomes a normal variable ✅
   c) It throws an error
   d) It extends the object scope
4. **Which statement is forbidden in strict mode?**
   a) with ✅
   b) debugger
   c) class
   d) export

5. **The `debugger` statement acts as:**
   a) A loop terminator
   b) A breakpoint ✓
   c) A variable declaration
   d) A function wrapper
6. **If no debugger is running, the `debugger` statement:**
   a) Throws an error
   b) Does nothing ✓
   c) Stops execution permanently
   d) Deletes variables
7. **Which directive enables strict mode in JavaScript?**
   a) enable strict
   b) "strict mode"
   c) "use strict" ✓
   d) import strict
8. **Where must `"use strict"` be placed in code?**
   a) Anywhere in the script
   b) At the start of a script or function ✓
   c) After variable declarations
   d) Inside a loop
9. **Which of the following is automatically strict code?**
   a) ES6 classes ✓
   b) var declarations
   c) with statements
   d) Non-module scripts
10. **What happens in strict mode if you assign to an undeclared variable?**
    a) It creates a global variable
    b) It throws a ReferenceError ✓
    c) It fails silently
    d) It assigns null
11. **In non-strict mode, functions invoked as functions (not methods) get `this` value as:**
    a) undefined
    b) null
    c) the global object ✓
    d) an empty object
12. **In strict mode, assigning to a non-writable property results in:**
    a) Silent failure
    b) TypeError ✓
    c) Warning
    d) Undefined behavior
13. **Which keyword is used to declare a block-scoped variable?**
    a) var
    b) let ✓
    c) const
    d) scope

14. **Which keyword declares constants?**
    a) var
    b) const ✓
    c) let
    d) static
15. **Which declaration is hoisted in JavaScript?**
    a) class
    b) function ✓
    c) let
    d) const
16. **Which declaration is NOT hoisted?**
    a) function
    b) var
    c) class ✓
    d) All of them
17. **Which directive prevents variables/functions from being declared inside `eval()` scope?**
    a) debugger
    b) import
    c) "use strict" ✓
    d) with
18. **In strict mode, duplicate function parameters cause:**
    a) Warning
    b) SyntaxError ✓
    c) Silent overwrite
    d) Ignored behavior
19. **Which of the following are reserved in strict mode?**
    a) eval and arguments ✓
    b) with and class
    c) import and var
    d) function and return
20. **Octal literals (e.g., 012) in strict mode:**
    a) Work normally
    b) Throw SyntaxError ✓
    c) Become decimal automatically
    d) Convert to hex
21. **Which of these throws an error in strict mode but not in non-strict mode?**
    a) Duplicate property names in object literals ✓
    b) Using var multiple times
    c) Declaring constants
    d) Using debugger
22. **In strict mode, attempting to delete a non-configurable property:**
    a) Fails silently
    b) Throws TypeError ✓

c) Converts to null
d) Is ignored

23. **What is the scope of variables declared with `var`?**
    a) Block
    b) Function ✓
    c) Module
    d) Global only
24. **Which keywords are considered declarations in JavaScript?**
    a) with, debugger, strict
    b) const, let, var, function, class, import, export ✓
    c) case, default, break
    d) yield, throw, return
25. **Which of the following is true for class declarations?**
    a) They are hoisted
    b) They must be declared before use ✓
    c) They replace functions
    d) They allow duplicate names
26. **What is the purpose of the `export` keyword?**
    a) To make values available in another module ✓
    b) To declare constants
    c) To import modules
    d) To enable strict mode
27. **What does `import { PI, TAU } from './constants.js';` do?**
    a) Exports values
    b) Imports values ✓
    c) Declares variables
    d) Creates objects
28. **Which statement handles exceptions?**
    a) throw
    b) try/catch/finally ✓
    c) debugger
    d) return
29. **Which statement is used to provide values in generator functions?**
    a) return
    b) yield ✓
    c) export
    d) import
30. **Which of the following is NOT a valid JavaScript statement?**
    a) break
    b) continue
    c) execute ✓
    d) switch

📃 30 MCQs on JavaScript Object Methods & Extended Syntax

1. What does the default `toLocaleString()` method in `Object` do?
a) Formats according to locale
b) Calls `toString()` and returns that value ✅
c) Adds commas automatically
d) Converts to JSON

2. Which classes override `toLocaleString()` for localization?
a) String and Boolean
b) Number and Date ✅
c) Array only
d) Math and RegExp

3. How does `Array.toLocaleString()` work?
a) Same as `toString()`
b) Calls each element's `toLocaleString()` ✅
c) Returns length
d) Always returns a JSON string

4. In the given `point` object, what will `point.toLocaleString()` output for `{x:1000,y:2000}`?
a) (1000, 2000)
b) (1,000, 2,000) ✅
c) (1000.00, 2000.00)
d) undefined

5. What is the purpose of `valueOf()` method?
a) To return a string
b) To convert object to primitive (usually number) ✅
c) To serialize object
d) To compute object length

6. Which class defines `valueOf()` to allow < and > comparisons?
a) Array
b) Date ✅
c) RegExp
d) Symbol

7. In the `point` example `{x:3,y:4}`, `Number(point)` returns?
a) 3
b) 4
c) 5 ✅
d) undefined

8. When is `valueOf()` automatically invoked?
a) During JSON.stringify()
b) When object is used in numeric context ✓
c) When calling console.log
d) Only if explicitly called

9. Does `Object.prototype` define `toJSON()`?
a) Yes
b) No ✓
c) Only in ES6
d) Only in strict mode

10. What does `JSON.stringify()` do if object has `toJSON()` method?
a) Ignores it
b) Calls it and serializes its return value ✓
c) Throws error
d) Converts to empty string

11. What does `Date.toJSON()` return?
a) Timestamp number
b) ISO string ✓
c) Locale date string
d) Null

12. In shorthand property syntax `let x=1,y=2; let o={x,y};`, what is `o.x+o.y`?
a) undefined
b) 1
c) 2
d) 3 ✓

13. Which ECMAScript version introduced shorthand properties?
a) ES3
b) ES5
c) ES6 ✓
d) ES2018

14. What is the purpose of computed property names `[ ]` in object literals?
a) To define functions
b) To dynamically compute property keys ✓
c) To enforce strict mode
d) To avoid prototype chain

15. Which of the following correctly creates `{p1:1, p2:2}`?
a) `{PROPERTY_NAME:1, computePropertyName():2}`
b) `{[PROPERTY_NAME]:1,[computePropertyName()]:2}` ✓

c) `{p1=1,p2=2}`
d) `new Object(PROPERTY_NAME,computePropertyName)`

16. Which of these can be used as property names with computed syntax?
a) Strings only
b) Numbers only
c) Symbols and strings ✓
d) Booleans

17. What is a Symbol in JavaScript?
a) A constructor
b) A primitive unique value ✓
c) A keyword
d) A class

18. Can two Symbols created with the same description be equal?
a) Yes
b) No ✓
c) Only in ES5
d) Depends on strict mode

19. Which method returns all Symbol keys of an object?
a) Object.keys()
b) Object.getOwnPropertySymbols() ✓
c) Reflect.getSymbols()
d) JSON.stringify()

20. The spread operator `...` in object literals was introduced in:
a) ES6
b) ES2015
c) ES2018 ✓
d) ES2020

21. What does `{...position,...dimensions}` do?
a) Creates an array
b) Copies properties from objects into a new one ✓
c) Creates inheritance
d) Freezes properties

22. If two objects with same property are spread, which value is kept?
a) First object's value
b) Last object's value ✓
c) Both
d) None

23. Does spread operator copy inherited properties?
a) Yes
b) No ✓
c) Sometimes
d) Only in strict mode

24. What is time complexity of spreading an object with n properties?
a) O(1)
b) O(log n)
c) O(n) ✓
d) O(n^2)

25. In shorthand method syntax `{area(){return this.side*this.side;}}`, `area` is:
a) Data property
b) Method ✓
c) Getter
d) Symbol

26. Which ECMAScript version introduced shorthand methods?
a) ES5
b) ES6 ✓
c) ES2018
d) ES2021

27. Which special Symbol must be defined to make object iterable?
a) Symbol.for()
b) Symbol.iterator ✓
c) Symbol.async
d) Symbol.loop

28. What do accessor properties use?
a) getter and setter functions ✓
b) constructor functions
c) spread operator
d) Symbol keys only

29. If accessor property defines only a getter, the property is:
a) Write-only
b) Read-only ✓
c) Read/write
d) Configurable

30. In example `const random={get octet(){return Math.floor(Math.random()*256)}};`, `random.octet` returns:
a) Always same value

b) Random number 0–255 ✓
c) Error
d) Undefined

Perfect 📃 Thanks for giving me this full text!
I'll now create **30 carefully selected MCQs** from the topics you shared:
👉 `slice()`, `splice()`, `fill()`, `copyWithin()`, `indexOf()`, `lastIndexOf()`, `includes()`, `sort()`, `reverse()`, `join()`, `toString()`, `toLocaleString()`, static array functions, array-like objects, strings as arrays.

✓ 30 MCQs on JavaScript Array Methods & Concepts

1. Which array method returns a subarray without modifying the original array?
a) splice()
b) slice() ✓
c) fill()
d) copyWithin()

2. In `a.slice(1, -1)` on `[1,2,3,4,5]`, the result is:
a) [1,2,3,4]
b) [2,3,4] ✓
c) [2,3,4,5]
d) [5]

3. What does `splice()` return?
a) A new array copy
b) The deleted elements ✓
c) The inserted elements
d) Nothing

4. In `a.splice(2, 0, "x")`, what happens?
a) Removes 2 items
b) Inserts "x" at index 2 ✓
c) Returns ["x"]
d) Throws error

5. Key difference between `slice()` and `splice()` second argument is:
a) slice → length, splice → end index
b) slice → end index, splice → length ✓
c) Both specify end index
d) Both specify length

6. What is the result of:

```
let a = [1,2,3,4,5];
a.splice(2,2,[10]);
```

a) [1,2,10,5] ✓
b) [1,2,[10],5]
c) [10,3,4,5]
d) [1,2,3,4,5]

7. Which method fills array with a constant value?
a) splice()
b) fill() ✓
c) map()
d) push()

8. Result of:

```
new Array(5).fill(0);
```

a) []
b) [0]
c) [0,0,0,0,0] ✓
d) undefined

9. What does `a.fill(8,2,-1)` on `[0,9,9,9,9]` give?
a) [0,9,8,8,9] ✓
b) [0,8,8,9,9]
c) [8,8,8,9,9]
d) [0,9,9,8,8]

10. Which method copies elements within the same array?
a) slice()
b) copyWithin() ✓
c) splice()
d) assign()

11. What is the output of:

```
[1,2,3,4,5].copyWithin(1);
```

a) [1,2,3,4,5]
b) [1,1,2,3,4] ✓
c) [1,2,2,3,4]
d) [1,2,3,3,4]

12. Which method is modeled after C's `memmove()`?
a) slice()
b) copyWithin() ✅
c) fill()
d) splice()

13. `indexOf()` compares values using:
a) ==
b) === ✅
c) Object.is
d) >

14. What does `[0,1,2,1,0].lastIndexOf(1)` return?
a) 1
b) 2
c) 3 ✅
d) -1

15. Which method can find `NaN` in an array?
a) indexOf()
b) lastIndexOf()
c) includes() ✅
d) findIndex()

16. Result of:

```
[1,true,3,NaN].indexOf(NaN)
```

a) 0
b) 2
c) -1 ✅
d) 3

17. `includes()` differs from `indexOf()` because:
a) Faster
b) Treats NaN as equal to NaN ✅
c) Returns index instead of boolean
d) Works only with strings

18. Default `sort()` order is:
a) Numerical
b) Alphabetical (string-based) ✅
c) Random
d) Ascending numeric

19. Result of:

```
[33,4,1111,222].sort();
```

a) [4,33,222,1111]
b) [1111,222,33,4] ✓
c) [33,4,1111,222]
d) Error

20. To sort numbers ascending properly:
a) arr.sort()
b) arr.sort((a,b)=>a-b) ✓
c) arr.sort((a,b)=>b-a)
d) arr.reverse()

21. Which method reverses elements in place?
a) slice()
b) reverse() ✓
c) join()
d) map()

22. `[1,2,3].reverse()` results in:
a) [1,2,3]
b) [3,2,1] ✓
c) [1,3,2]
d) Error

23. Which method joins all array elements into a string?
a) concat()
b) join() ✓
c) split()
d) stringify()

24. Result of:

```
[1,2,3].join("")
```

a) "1,2,3"
b) "1 2 3"
c) "123" ✓
d) 6

25. `[1,[2,"c"]].toString()` gives:
a) "1,[2,c]"
b) "1,2,c" ✓

c) "[1,2,c]"
d) Error

26. Which method provides locale-specific array-to-string conversion?
a) toString()
b) toLocaleString() ✓
c) join()
d) stringify()

27. Which static function checks if a value is an array?
a) Array.of()
b) Array.isArray() ✓
c) Array.from()
d) Object.isArray()

28. Array-like objects are defined as:
a) Any object with numeric indexes and length ✓
b) Must inherit from Array
c) Must be iterable
d) Must use Symbol.iterator

29. To convert array-like object to real array:
a) Array.toArray(obj)
b) Array.from(obj) ✓
c) obj.slice()
d) Object.assign(obj)

30. Strings in JavaScript behave like:
a) Mutable arrays
b) Read-only arrays ✓
c) Objects only
d) Symbols

Chapter 8: Functions

1. Which keyword is used to define a function?

A) func

B) define

C) function

D) fn

Answer: C

2. Arrow functions were introduced in which ECMAScript version?

A) ES5

B) ES6

C) ES7

D) ES8

Answer: B

3. Functions in JavaScript are _____ objects.

A) primitive

B) special

C) first-class

D) temporary

Answer: C

4. The 'this' keyword inside a function refers to?

A) global object

B) function itself

C) the object that owns it

D) window always

Answer: C

5. Arrow functions automatically bind 'this'.

A) True

B) False

C) Depends

D) Not applicable

Answer: A

6. Which method permanently binds a function to an object?

A) call()

B) bind()

C) apply()

D) attach()

Answer: B

7. Default parameter values are evaluated when?

A) At function creation

B) At function call

C) Once per script

D) Never

Answer: B

8. Closures allow a function to access variables from?

A) Global scope

B) Outer function scope

C) Another module

D) Local constants

Answer: B

9. Which operator spreads array elements into arguments?

A) ++

B) ...

C) =>

D) @

Answer: B

10. Function declarations are hoisted to the top of?

A) Global scope

B) Their block

C) Their enclosing scope

D) File

Answer: C

11. Which keyword is used to define a function?

A) func

B) define

C) function

D) fn

Answer: C

12. Arrow functions were introduced in which ECMAScript version?

A) ES5

B) ES6

C) ES7

D) ES8

Answer: B

13. Functions in JavaScript are _____ objects.

A) primitive

B) special

C) first-class

D) temporary

Answer: C

14. The 'this' keyword inside a function refers to?

A) global object

B) function itself

C) the object that owns it

D) window always

Answer: C

15. Arrow functions automatically bind 'this'.

A) True

B) False

C) Depends

D) Not applicable

Answer: A

16. Which method permanently binds a function to an object?

A) call()

B) bind()

C) apply()

D) attach()

Answer: B

17. Default parameter values are evaluated when?

A) At function creation

B) At function call

C) Once per script

D) Never

Answer: B

18. Closures allow a function to access variables from?

A) Global scope

B) Outer function scope

C) Another module

D) Local constants

Answer: B

19. Which operator spreads array elements into arguments?

A) ++

B) ...

C) =>

D) @

Answer: B

20. Function declarations are hoisted to the top of?

A) Global scope

B) Their block

C) Their enclosing scope

D) File

Answer: C

21. Which keyword is used to define a function?

A) func

B) define

C) function

D) fn

Answer: C

22. Arrow functions were introduced in which ECMAScript version?

A) ES5

B) ES6

C) ES7

D) ES8

Answer: B

23. Functions in JavaScript are _____ objects.

A) primitive

B) special

C) first-class

D) temporary

Answer: C

24. The 'this' keyword inside a function refers to?

A) global object

B) function itself

C) the object that owns it

D) window always

Answer: C

25. Arrow functions automatically bind 'this'.

A) True

B) False

C) Depends

D) Not applicable

Answer: A

Chapter 9: Classes

1. JavaScript classes are based on what concept?

A) Inheritance tree

B) Prototypes

C) Data tables

D) Functions only

Answer: B

2. What keyword is used to inherit another class?

A) implements

B) inherits

C) extends

D) super

Answer: C

3. The 'super' keyword is used to?

A) Call subclass methods

B) Access superclass constructors/methods

C) Create static methods

D) Override prototype

Answer: B

4. Private fields in classes start with?

A) $

B) _

C) #

D) @

Answer: C

5. Static methods belong to?

A) Instances

B) Subclasses

C) Class itself

D) Global object

Answer: C

6. JavaScript classes are based on what concept?

A) Inheritance tree

B) Prototypes

C) Data tables

D) Functions only

Answer: B

7. What keyword is used to inherit another class?

A) implements

B) inherits

C) extends

D) super

Answer: C

8. The 'super' keyword is used to?

A) Call subclass methods

B) Access superclass constructors/methods

C) Create static methods

D) Override prototype

Answer: B

9. Private fields in classes start with?

A) $

B) _

C) #

D) @

Answer: C

10. Static methods belong to?

A) Instances

B) Subclasses

C) Class itself

D) Global object

Answer: C

11. JavaScript classes are based on what concept?

A) Inheritance tree

B) Prototypes

C) Data tables

D) Functions only

Answer: B

12. What keyword is used to inherit another class?

A) implements

B) inherits

C) extends

D) super

Answer: C

13. The 'super' keyword is used to?

A) Call subclass methods

B) Access superclass constructors/methods

C) Create static methods

D) Override prototype

Answer: B

14. Private fields in classes start with?

A) $

B) _

C) #

D) @

Answer: C

15. Static methods belong to?

A) Instances

B) Subclasses

C) Class itself

D) Global object

Answer: C

16. JavaScript classes are based on what concept?

A) Inheritance tree

B) Prototypes

C) Data tables

D) Functions only

Answer: B

17. What keyword is used to inherit another class?

A) implements

B) inherits

C) extends

D) super

Answer: C

18. The 'super' keyword is used to?

A) Call subclass methods

B) Access superclass constructors/methods

C) Create static methods

D) Override prototype

Answer: B

19. Private fields in classes start with?

A) $

B) _

C) #

D) @

Answer: C

20. Static methods belong to?

A) Instances

B) Subclasses

C) Class itself

D) Global object

Answer: C

21. JavaScript classes are based on what concept?

A) Inheritance tree

B) Prototypes

C) Data tables

D) Functions only

Answer: B

22. What keyword is used to inherit another class?

A) implements

B) inherits

C) extends

D) super

Answer: C

23. The 'super' keyword is used to?

A) Call subclass methods

B) Access superclass constructors/methods

C) Create static methods

D) Override prototype

Answer: B

24. Private fields in classes start with?

A) $

B) _

C) #

D) @

Answer: C

25. Static methods belong to?

A) Instances

B) Subclasses

C) Class itself

D) Global object

Answer: C

Chapter 11: JavaScript Standard Library

1. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

2. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

3. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

4. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

5. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

6. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

7. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

8. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

9. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

10. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

11. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

12. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

13. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

14. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

15. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

16. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

17. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

18. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

19. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

20. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

21. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

22. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

23. The Set object stores only?

A) Primitive values

B) Unique values

C) String keys

D) Numbers only

Answer: B

24. What does console.table() do?

A) Draws charts

B) Prints tables of objects

C) Clears console

D) Prints JSON

Answer: B

25. Which object stores key-value pairs with unique keys?

A) Array

B) Map

C) Set

D) WeakMap

Answer: B

Chapter 13: Asynchronous JavaScript

1. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

2. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

3. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

4. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

5. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

6. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

7. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

8. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

9. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

10. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

11. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

12. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

13. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

14. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

15. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

16. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

17. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

18. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

19. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

20. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

21. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

22. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

23. Which keyword makes a function return a Promise?

A) async

B) await

C) defer

D) then

Answer: A

24. Which method handles promise rejection?

A) catch()

B) fail()

C) reject()

D) finally()

Answer: A

25. Promises have how many states?

A) 2

B) 3

C) 4

D) 5

Answer: B

Chapter 15: JavaScript in Web Browsers

1. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

2. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

3. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

4. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

5. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

6. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

7. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

8. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

9. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

10. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

11. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

12. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

13. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

14. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

15. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

16. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

17. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

18. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

19. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

20. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

21. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

22. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

23. DOM stands for?

A) Document Object Model

B) Data Object Model

C) Dynamic Object Manager

D) Direct Object Mapping

Answer: A

24. Which method selects an element by ID?

A) getElement()

B) select()

C) getElementById()

D) query()

Answer: C

25. The global browser object is?

A) document

B) window

C) console

D) screen

Answer: B

Chapter 17: JavaScript Tools and Extensions

1. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

2. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

3. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

4. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

5. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

6. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

7. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

8. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

9. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

10. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

11. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

12. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

13. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

14. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

15. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

16. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

17. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

18. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

19. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

20. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

21. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

22. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B

23. TypeScript adds what feature to JavaScript?

A) Dynamic typing

B) Static typing

C) Extra operators

D) Async syntax

Answer: B

24. npm stands for?

A) Node Programming Module

B) Node Package Manager

C) Network Package Manager

D) New Project Manager

Answer: B

25. Babel is used for?

A) Minifying JS

B) Transpiling modern JS

C) Linting

D) Debugging

Answer: B