

TypeScript JSX MCQs (140)

1. Which keyword is used to define a variable that can be hoisted?

- A. const
- B. let
- *** C. var
- D. function

2. In TypeScript, what is the scope of a variable declared with `var`?

- A. Block
- *** B. Function
- C. Global only
- D. Class

3. What will be the output?

```
```ts
console.log(a);
var a = 10;
```
```

- A. Error
- *** B. undefined
- C. 10
- D. null

4. Which of the following allows defining a method inside a class?

- A. interface
- B. map
- *** C. method
- D. enum

5. Which method is used to iterate over Map elements?

- A. forEach
- *** B. map.forEach
- C. map.iterator
- D. each

6. Which of the following can be used to strongly type objects?

- A. Class
- *** B. Interface
- C. Enum
- D. Any

7. Which symbol is used to declare a decorator?

- A. \ \$
- *** B. @
- C. #
- D. \ ~

8. What does `any` type allow in TypeScript?

- *** A. Any kind of value without type checking
- B. Strict null checking
- C. Only primitive values
- D. Only object values

9. What does an enum provide in TypeScript?

- A. Encapsulation
- *** B. Named constants
- C. Runtime polymorphism
- D. Dynamic typing

10. What is the output?

```
```ts
let arr: [number, string] = [1, "Hello"];
console.log(arr[0]);
```
```

- *** A. 1
- B. Hello
- C. undefined
- D. Error

11. TypeScript interfaces are used for:

- *** A. Defining contracts for classes or objects
- B. Controlling routing
- C. HTTP requests
- D. Variable scoping

12. Which of the following iterates over the values of an array?

- A. for..in
- *** B. for..of
- C. forEach index
- D. map

13. Which function can be used to define metadata in a decorator?

- *** A. Reflect.metadata()
- B. Meta.define()
- C. Symbol.metadata
- D. Object.meta

14. How do you define a tuple with a number and a string in TypeScript?

- *** A. \[number, string]
- B. (number, string)
- C. {number: string}
- D. \[string, number]

15. Interfaces support which features?

- *** A. Optional properties, Readonly, Inheritance
- B. Only variables
- C. No inheritance
- D. Only methods

16. Which of these is a feature of `Map` object?

- A. Indexed access
- *** B. Key-value pairs with any type of keys
- C. Only string keys
- D. Immutable

17. The correct syntax to declare an enumeration:

- *** A. `enum Colors {Red, Green, Blue}`
- B. `colors = [Red, Green, Blue]`
- C. `enum = Colors[Red]`
- D. `enum: Colors = Red, Green`

18. Type annotations in TypeScript help with:

- *** A. Compile-time type checking
- B. Code minification
- C. DOM rendering
- D. CSS parsing

19. Tuple type allows:

- A. Arrays with variable length
- *** B. Arrays with fixed length and types
- C. Key-value pairs
- D. JSON structures

20. A decorator can be applied to:

- *** A. Class, Method, Property, Parameter
- B. Only functions
- C. Only variables
- D. Only methods

21. `let` and `const` are block scoped, while `var` is:

- *** A. Function scoped
- B. Block scoped
- C. Class scoped
- D. Object scoped

22. Which TypeScript feature ensures type safety?

- A. `var`
- B. Arrays

*** C. Interfaces
D. for loop

23. Which TypeScript type can store any value and bypass type checking?

A. string
*** B. any
C. object
D. void

24. In a class, the constructor is used for:

*** A. Initializing class members
B. Creating interfaces
C. Declaring tuples
D. Assigning enums

25. TypeScript supports which types of inheritance?

A. Only single
*** B. Single and multiple via interfaces
C. Only mixins
D. No inheritance

26. What is the correct way to declare a class in TypeScript?

*** A. ``class Person { constructor() {} }``
B. ``interface Person {}``
C. ``function class() {}``
D. ``class = Person()``

27. The ``.values()`` method in a ``Map`` returns:

A. Key-value
*** B. Values only
C. Keys only
D. Length

28. TypeScript iterators are mainly used with:

A. Sets only
B. Objects
*** C. Arrays, Maps, Sets
D. Enums

29. A tuple defined as ``[string, number]`` will throw an error when:

A. Values are assigned in the same order
*** B. Values are assigned as `\[10, "test"]`
C. It's empty
D. It's longer than 2

30. The correct way to define a function returning ``void``:

*** A. `function greet(): void {}`
B. `function greet() = void {}`
C. `void greet() {}`
D. `function greet: void()`

31. Which is NOT a valid use of a decorator?

A. @Component
B. @Injectable
*** C. @Loop
D. @Input

32. A class must implement all members of:

*** A. The interface it implements
B. The tuple it returns
C. Any enum
D. A variable

33. Iterators in TypeScript must implement:

*** A. next() method
B. push() method
C. index() method
D. done() method only

34. A feature of using Maps over Objects:

*** A. Keys can be of any type
B. Keys must be strings
C. No iterable methods
D. Can't be iterated

35. Which of these declarations is incorrect?

A. let a: any
B. let b: string
C. let c: number\[]
*** D. let d = \[number, string]

36. Which concept helps in reusability of code in TypeScript?

A. Tuple
B. var
*** C. Interface
D. enum

37. Type annotations are used:

*** A. To specify variable types explicitly
B. For HTML rendering
C. Only in classes
D. For generating IDs

38. Output of the following:

```
```ts
const a = [1, 2];
a.push(3);
console.log(a.length);
```
```

- A. Error
- *** B. 3
- C. 2
- D. undefined

39. Decorators execute:

- A. After class creation
- *** B. At runtime, before instantiation
- C. After constructor
- D. After main()

40. `typeof` operator in TypeScript returns:

- *** A. A string representing the type
- B. A number
- C. An enum
- D. A function

Great! Here's the next batch:

ðŸ”§ Working with NPM & Node.js â€“ MCQs (41â€“80)

41. What does `npm` stand for?

- *** A. Node Package Manager
- B. Node Project Manager
- C. Node Process Manager
- D. Node Protocol Manager

42. What is the command to initialize a new npm project?

- A. npm start
- *** B. npm init
- C. npm create
- D. npm generate

43. What file is generated by running `npm init`?

- A. config.json
- *** B. package.json
- C. index.js
- D. npm.json

44. What does `package-lock.json` do?

- A. Stores all dependencies
- *** B. Records exact version of installed packages
- C. Locks the package.json file
- D. Prevents updates

45. Which command installs all packages from `package.json`?

- A. npm update
- B. npm build
- *** C. npm install
- D. npm create

46. How do you uninstall a package?

- A. npm remove
- B. npm delete
- *** C. npm uninstall
- D. npm reset

47. To update a package to the latest version:

- *** A. npm update [package-name]
- B. npm upgrade
- C. npm latest
- D. npm push

48. What is a dependency in `package.json`?

- A. Node version info
- *** B. Libraries required to run the app
- C. Compilation files
- D. Test cases

49. Which key in `package.json` defines the main file of the app?

- *** A. "main"
- B. "start"
- C. "root"
- D. "index"

50. What is the correct command to install a dev dependency?

- *** A. npm install package-name --save-dev
- B. npm dev package-name
- C. npm install --dev package-name
- D. npm add-dev package-name

51. Which of the following is a built-in Node.js module?

- *** A. fs
- B. lodash
- C. express
- D. chalk

52. What does the `fs` module handle?

- A. Database
- B. HTTP Requests
- *** C. File System Operations
- D. Routing

53. What does `http` module in Node.js help with?

- *** A. Creating servers and handling HTTP requests
- B. File storage
- C. Encryption
- D. Data compression

54. What module is used for creating secure servers in Node.js?

- A. secure
- *** B. https
- C. tls
- D. ssl

55. Which built-in module helps with working on file and directory paths?

- A. fs
- *** B. path
- C. url
- D. routes

56. `EventEmitter` is part of which Node module?

- A. events-stream
- *** B. events
- C. emitter-core
- D. node-core

57. Which method emits an event in Node.js?

- *** A. emitter.emit()
- B. emitter.send()
- C. event.run()
- D. signal.emit()

58. What does the `require()` function do in Node.js?

- A. Delete a module
- B. Convert JSON
- *** C. Import a module
- D. Export a module

59. Which is the correct way to import the `fs` module?

- *** A. const fs = require('fs');

- B. import 'fs';
- C. get fs from 'fs';
- D. fs.import();

60. `npm install` without any package installs:

- *** A. All dependencies in package.json
- B. Only global packages
- C. Dev dependencies only
- D. Production packages only

61. Which file stores exact dependency versions installed?

- A. package.json
- *** B. package-lock.json
- C. npm.json
- D. lock.json

62. What is required to create a minimal Node.js app?

- A. index.html
- *** B. index.js + package.json
- C. app.css
- D. node.ts

63. What does this code do?

```
```js
const fs = require("fs");
fs.writeFileSync("hello.txt", "Hello Node");
```
```

- A. Reads file
- *** B. Creates file with content
- C. Deletes file
- D. Downloads file

64. What kind of programming model does Node.js use?

- A. Blocking
- *** B. Non-blocking, event-driven
- C. Threaded
- D. Linear

65. What is a callback in Node.js?

- *** A. A function passed to another function to be executed later
- B. A return value
- C. A method chain
- D. A node

66. What is a Promise?

- A. An error handler
- *** B. An object representing the future completion/failure of an operation

- C. A stream
- D. A timer

67. What is the purpose of `async/await` in Node.js?

- A. Improve routing
- *** B. Handle asynchronous code in a synchronous way
- C. Create server
- D. Load packages

68. Which statement is correct?

- *** A. Promises are cleaner than callbacks
- B. Callbacks are more modern
- C. Promises are slower
- D. Callbacks never cause nesting

69. `await` can only be used inside:

- A. function()
- B. setTimeout
- *** C. async function
- D. require

70. How do you define an async function?

- A. function async() {}
- *** B. async function name() {}
- C. async = function() {}
- D. define async()

71. Which of the following is a benefit of Express.js?

- A. File system support
- *** B. Simple API for routing and middleware
- C. Built-in templating
- D. No need for Node

72. How to import Express in a Node.js project?

- *** A. const express = require('express');
- B. import 'express';
- C. get express from 'express';
- D. express.load();

73. What is middleware in Express.js?

- *** A. A function that handles request/response objects
- B. A UI component
- C. A database wrapper
- D. A testing library

74. How to define a GET route in Express?

```
```js
app.get("/", (req, res) => res.send("Hi"));
```
```

- A. Correct
- *** B. Correct
- C. Wrong
- D. Only for POST

75. What does this route handle?

```
```js
app.post("/submit", handler);
```
```

- *** A. POST requests to /submit
- B. GET requests to /submit
- C. All requests to /submit
- D. DELETE requests

76. How do you handle dynamic routes in Express?

- *** A. `app.get('/user/:id')`
- B. `app.route('/user@id')`
- C. `app.any('/user')`
- D. `app.fetch('/user?id=')`

77. Middleware functions must call:

- A. `done()`
- *** B. `next()`
- C. `complete()`
- D. `forward()`

78. Which port is commonly used to run Express apps?

- A. 80
- B. 21
- *** C. 3000
- D. 5000

79. Express routes must be defined on:

- A. `app.route()`
- *** B. `app.METHOD(path, handler)`
- C. `route(app)`
- D. `route.set()`

80. To parse JSON request bodies in Express:

- *** A. `app.use(express.json())`
- B. `app.use(json())`
- C. `app.set('json')`
- D. `express.JSON()`

Excellent! Here's the final batch of MCQs (81â€“120) covering:

- * Advanced Express.js Routing
- * Middleware
- * Asynchronous Patterns
- * Node.js Core Modules
- * Real-world Practices

81. What will this code do?

```
```js
app.use((req, res, next) => {
 console.log("Middleware");
 next();
});
```
```

- *** A. Logs every request and moves to the next middleware
- B. Sends a response
- C. Blocks all requests
- D. Triggers an error

82. What happens if `next()` is not called in Express middleware?

- A. Response is automatically sent
- *** B. Request hangs without response
- C. App crashes
- D. It skips other routes

83. How to create an Express router?

- *** A. `const router = express.Router();`
- B. `const route = express.create();`
- C. `express.route();`
- D. `app.router();`

84. What is the purpose of `express.Router()`?

- A. Serve static files
- *** B. Create modular route handlers
- C. Read POST data
- D. Handle events

85. Which HTTP method is used for updating a resource?

- A. GET
- B. DELETE
- *** C. PUT
- D. FETCH

86. What does this route match?

```
```js
app.get('/user/:id', handler);
```
```

- A. `/user/`

*** B. /user/123
C. /user?id=123
D. /user:123

87. Route parameters are accessed using:

*** A. req.params
B. req.query
C. req.body
D. req.header

88. How do you handle form submissions in Express?

A. req.query
*** B. req.body (with body-parser or express.json())
C. req.form
D. req.send

89. When should `express.urlencoded()` be used?

*** A. For parsing form data (application/x-www-form-urlencoded)
B. For JSON APIs
C. For static files
D. For database queries

90. A chained route definition for `/product` might look like:

*** A. app.route('/product').get().post();
B. app.get('/product').post();
C. app.product('/product')
D. app.route('/product').get().put();

91. The `error-handling middleware` in Express must have:

A. 2 arguments
B. 3 arguments
*** C. 4 arguments (err, req, res, next)
D. 1 argument

92. What status code represents "Created"?

A. 200
B. 204
*** C. 201
D. 500

93. What does the `path` module in Node.js help with?

A. Working with databases
*** B. Resolving and formatting file paths
C. Creating servers
D. Handling routes

94. Node.js is single-threaded but:

- *** A. Uses an event loop for asynchronous tasks
- B. Uses multiple CPUs
- C. Uses WebAssembly
- D. Uses async threads

95. What is the default export method in Node.js?

- *** A. module.exports
- B. export default
- C. require()
- D. module.export()

96. What keyword in package.json marks scripts for execution?

- *** A. "scripts"
- B. "run"
- C. "cmd"
- D. "execute"

97. How do you run a script named `start` from package.json?

- A. npm run start.js
- *** B. npm start
- C. node start
- D. npm execute start

98. In a package.json, `dependencies` section is used for:

- *** A. Runtime packages required to run the app
- B. Testing libraries
- C. Build tools
- D. Linting tools

99. Which lifecycle method is called before `npm install` begins?

- A. preinstall
- *** B. preinstall
- C. install
- D. before

100. Which Node.js function reads a file asynchronously?

- A. fs.readFileSync
- *** B. fs.readFile
- C. readFile
- D. fs.read

101. What will this log?

```
```js
console.log(process.cwd());
```
```

*** A. Current working directory
B. Node version
C. Root path
D. File path

102. What does `__dirname` return?

A. The filename
*** B. Directory path of current module
C. Node home path
D. Relative root

103. What built-in module handles compression?

A. zip
*** B. zlib
C. compress
D. stream

104. Which of the following is used to catch unhandled exceptions in Node.js?

*** A. process.on('uncaughtException', fn)
B. catch()
C. node.catch()
D. app.onError()

105. Which module is used for streaming operations in Node?

*** A. stream
B. flow
C. pipe
D. buffer

106. What does `res.sendStatus(404)` do in Express?

A. Send 404 page
*** B. Send status code and end response
C. Send HTML file
D. Redirect

107. What is the difference between `res.send()` and `res.json()`?

A. None
*** B. json() stringifies the object automatically
C. send() parses JSON
D. send() returns XML

108. To serve static files in Express:

*** A. app.use(express.static('public'))
B. app.get('/static')
C. app.serveFiles()
D. express.serve()

109. What does `res.redirect()` do?

- A. Sends JSON
- *** B. Redirects the client to a new URL
- C. Updates headers
- D. Ends request

110. What is the result of:

```
```js
setImmediate(() => console.log("Immediate"));
process.nextTick(() => console.log("Next Tick"));
```
```

- *** A. Next Tick logs before Immediate
- B. Immediate logs first
- C. They run in parallel
- D. Error

111. The order of execution:

- A. `setTimeout` < `nextTick` < `setImmediate`
- *** B. `process.nextTick` < `Promise` < `setTimeout` < `setImmediate`
- C. `Promise` < `nextTick`
- D. `nextTick` < `setTimeout` < `Promise`

112. What tool tracks exact changes in node modules?

- *** A. `package-lock.json`
- B. `version.json`
- C. `npm-cache.json`
- D. `deps.json`

113. A key benefit of using Express over raw Node.js:

- *** A. Simplified routing and middleware system
- B. Multi-threading
- C. File uploads
- D. SSL by default

114. What does `res.status(200).json(data)` do?

- A. Sends plain HTML
- *** B. Sends HTTP status 200 with JSON data
- C. Creates a POST request
- D. Logs data

115. How can you restrict access to certain routes in Express?

- A. Using query params
- *** B. Using middleware
- C. Using templates
- D. Using static folders

116. Node.js uses what type of I/O model?

- A. Blocking
- *** B. Non-blocking, event-driven
- C. Stream-based
- D. Time-based

117. What function handles error in Express middleware?

- *** A. next(err)
- B. throw error
- C. return error
- D. send(error)

118. What keyword is used to export a function in CommonJS?

- *** A. module.exports = fn
- B. export default
- C. define()
- D. expose()

119. What is the purpose of `.env` in a Node.js app?

- *** A. Storing environment variables securely
- B. Routing rules
- C. JavaScript versioning
- D. Package settings

120. What is a recommended way to manage environment variables?

- *** A. dotenv package
- B. fs module
- C. process.js
- D. envjs