**Bootcamp Test**

**1. Which of the following is not a primitive data type in JavaScript?**

a) String

b) Number

c) Object

d) Boolean

**2. What will be the output of the following code snippet?**

**console.log(typeof null);**

a) "null"

b) "object"

c) "undefined"

d) "number"

**3. What is a closure in JavaScript?**

a) A function bundled together with its lexical environment.

b) A function that calls itself.

c) A function with no parameters.

d) A function that is called before it is defined.

**4. Which of the following is true about closures?**

a) They can access variables from their own scope only.

b) They can access variables from their own scope and the outer function scope.

c) They can only access global variables.

d) They cannot access variables from the outer function scope.

**5. How do you define an object in JavaScript?**

a) let obj = [];

b) let obj = {};

c) let obj = ();

d) let obj = <>;

**6. Which of the following is the correct way to access the property "name" in the object person?**

a) person[name]

b) person.name

c) person>name

d) person:name

**7. What is the primary purpose of the event loop in JavaScript?**

a) To handle synchronous code execution.

b) To handle asynchronous code execution.

c) To handle variable declaration.

d) To handle object creation.

**8. Which part of the JavaScript runtime handles the event loop?**

a) Web APIs

b) Call Stack

c) Node.js

d) Browser

**9. What is the purpose of prototypes in JavaScript?**

a) To allow object inheritance.

b) To handle asynchronous operations.

c) To define strict types for objects.

d) To create immutable objects.

**10. Which of the following methods is used to create a new object with a specified prototype?**

a) Object.prototype

b) Object.create()

c) Object.new()

d) Object.construct()

**11. Which CSS property is used to create a flex container?**

a) display: block;

b) display: flex;

c) display: grid;

d) display: inlineblock;

**12. Which property is used to align flex items along the main axis?**

a) justifycontent

b) alignitems

c) flexdirection

d) aligncontent

**13. Which command is used to create a new React app using Create React App (CRA)?**

a) npx createreactapp myapp

b) npm createreactapp myapp

c) npx reactapp myapp

d) npm reactapp myapp

**14. Which hook is used to add state to a functional component in React?**

a) useEffect

b) useState

c) useContext

d) useReducer

**15. Which component is used for navigation between pages in a React application?**

a) <NavLink>

b) <Route>

c) <Link>

d) <BrowserRouter>

**16. What is the correct way to define a route for the component Home in React Router?**

a) <Route path="/home" component={Home} />

b) <Route path="/home" render={Home} />

c) <Route path="/home" link={Home} />

d) <Route path="/home" navigate={Home} />

**17. Which method is used to create a new server in Node.js?**

a) http.createServer()

b) express.createServer()

c) server.createServer()

d) node.createServer()

**18. How do you define a route in Express to handle GET requests at the path '/users'?**

a) app.get('/users', (req, res) => {});

b) app.route('/users', (req, res) => {});

c) app.fetch('/users', (req, res) => {});

d) app.request('/users', (req, res) => {});

**19. Which method of the Fetch API is used to send data to a server?**

a) GET

b) POST

c) PUT

d) DELETE

**20. What does the following Fetch API code snippet do?**

**fetch('https://api.example.com/data')**

**.then(response => response.json())**

**.then(data => console.log(data));**

a) Sends data to the server and logs the response.

b) Fetches data from the server and logs the response.

c) Deletes data on the server and logs the response.

d) Updates data on the server and logs the response.

**21. What will be the output of the following code snippet?**

**const arr1 = [1, 2, 3];**

**const arr2 = [...arr1, 4, 5];**

**console.log(arr2);**

a) [1, 2, 3, 4, 5]

b) [1, 2, 3, [4, 5]]

c) [1, 2, 3, [1, 2, 3], 4, 5]

d) Error

**22. How can you combine the following two objects into one using the spread operator?**

**const obj1 = { a: 1, b: 2 };**

**const obj2 = { c: 3, d: 4 };**

a) { ...obj1, ...obj2 }

b) { obj1, obj2 }

c) Object.assign(obj1, obj2)

d) [...obj1, ...obj2]

**23. What is the output of the following code snippet?**

**const person = { name: 'John', age: 30 };**

**const { name, age } = person;**

**console.log(name, age);**

a) undefined undefined

b) John 30

c) { name: 'John', age: 30 }

d) Error

**24. How can you add a new property city with the value New York to the object person?**

**let person = { name: 'Alice', age: 25 };**

a) person.city = 'New York';

b) person['city'] = 'New York';

c) Object.assign(person, { city: 'New York' });

d) All of the above

**25. What will be the output of the following code snippet?**

**const numbers = [1, 2, 3, 4];**

**const doubled = numbers.map(num => num \* 2);**

**console.log(doubled);**

a) [2, 4, 6, 8]

b) [1, 2, 3, 4]

c) [3, 6, 9, 12]

d) Error

**26. Which code snippet correctly filters out odd numbers from the array?**

**const numbers = [1, 2, 3, 4, 5];**

a) const evens = numbers.filter(num => num % 2 === 0);

b) const evens = numbers.map(num => num % 2 === 0);

c) const evens = numbers.reduce((num => num % 2 === 0));

d) const evens = numbers.find(num => num % 2 === 0);

**27. What will be the output of the following code snippet?**

**const promise = new Promise((resolve, reject) => {**

**setTimeout(() => resolve('Done!'), 1000);**

**});**

**promise.then(result => console.log(result));**

a) undefined

b) Error

c) Done!

**28. What will be the output of the following code snippet?**

**function fetchData(callback) {**

**setTimeout(() => callback('Data fetched'), 2000);**

**}**

**fetchData(result => console.log(result));**

a) undefined

b) Error

c) Data fetched

d) Promise

**29. How do you pass a callback function to another function?**

**function processData(data, callback) {**

**// process data**

**callback(data);**

**}**

a) processData(callback, data);

b) processData(data, callback);

c) callback(processData, data);

d) processData(callback);

**30. What is the main purpose of a callback function?**

a) To handle synchronous operations

b) To handle asynchronous operations

c) To handle exceptions

d) To define types

**31. Which command is used to initialize a new Node.js project and create a package.json file?**

a) npm start

b) npm init

c) npm install

d) npm run

**32. How do you install the Express package using npm?**

a) npm init express

b) npm get express

c) npm install express

d) npm add express

**33. Which command is used to install all dependencies listed in the package.json file?**

a) npm install

b) npm update

c) npm get

d) npm add

**34. What is the purpose of the packagelock.json file?**

a) To specify the project's dependencies

b) To ensure consistent installs across different environments

c) To lock the Node.js version

d) To store environment variables

**35. Write a function that uses the spread operator to merge two arrays.**

a) function mergeArrays(arr1, arr2) { return [...arr1, ...arr2]; }

b) function mergeArrays(arr1, arr2) { return arr1.concat(arr2); }

c) function mergeArrays(arr1, arr2) { return [arr1, arr2]; }

d) function mergeArrays(arr1, arr2) { return [arr1.concat(arr2)]; }

**36. Write a function that takes an object and adds a new keyvalue pair to it.**

a) function addObjectProperty(obj, key, value) { obj[key] = value; }

b) function addObjectProperty(obj, key, value) { obj.push({ key: value }); }

c) function addObjectProperty(obj, key, value) { obj[key] = key: value; }

d) function addObjectProperty(obj, key, value) { obj.add(key, value); }

**37. What is the main advantage of using arrow functions in JavaScript?**

a) They provide better performance than regular functions.

b) They automatically bind to the this context of the surrounding code.

c) They allow for variable hoisting within their scope.

d) They can be used as generators.

**38. How do you define a default parameter in a JavaScript function?**

a) function add(x, y = 0) { return x + y; }

b) function add(x = 0, y) { return x + y; }

c) function add(x, y) { return x + y || 0; }

d) function add(x = 0, y = 0) { return x + y; }

**39. Which method is used to handle errors in Promises?**

a) then()

b) catch()

c) resolve()

d) reject()

**40. How do you chain multiple Promises together?**

a) Using then()

b) Using catch()

c) Using finally()

d) Using all()