Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 70%. We keep your highest score.

1. What best describes the network operations that Node.js makes? 1 / 1 point

○ Applications block every network operation to complete at the same time on the server

○ Blocked operations return immediately without added processing time on the server

● Non-blocked operations return immediately without added processing time on the server

○ Non-blocked operations return in a synchronized manner with added processing time on the server

✓ Correct
Node.js makes all network operations in a non-blocking manner, and every network operation returns immediately.

2. Before the Node.js framework receives the HTTP response message from the remote web server, it immediately returns a result for the http.request function call. What does this result state? 1 / 1 point

○ The return message will be sent successfully.

● A request is in progress.

○ The response message will be sent successfully.

○ The callback function was called successfully.

✓ Correct
This result indicates that the request message was sent successfully.

3. What parameter is optional in an HTTP request? 1 / 1 point

○ Event variable parameter

○ Resource variable parameter

● Callback function parameter

○ Location function parameter

✓ Correct
The callback function parameter is optional; you can send an HTTP request and disregard the response message.

4. What object do Node.js modules in the SDK pass as the first parameter in a callback function? 1 / 1 point

● Error

○ Location

○ Destination

○ Identity

✓ Correct
Node.js modules pass an error object as the first parameter in a callback function.}

5. What do you pass back in the callback handler to indicate a successful return? 1 / 1 point

● Null object

○ Empty string

○ 404 status code

○ Error

✓ Correct
Null indicates everything worked.

6. Which function calls the resultCallback callback function to return results to the main application? 1 / 1 point

○ result()

○ step()

● http.request()

○ aggregate_context()

✓ Correct
The http.request( )callback function calls resultCallback to return the result to the main application.

7. Why can inversion of control be an issue when using callbacks? 1 / 1 point

○ Inversion of control is used when something needs to be done sequentially which makes code readability difficult

○ Inversion of control creates the need for nested callbacks

● Callbacks sometimes hand over control to third-party code which can make errors hard to identify

○ Inversion of control uses promises which are difficult to implement

✓ Correct
Inversion of control refers to a callback handing over control of the application to third-party code, making errors hard to identify.

8. JSON is the standard format for API data exchange. What relationship does JSON have with Node.js? 1 / 1 point

○ JSON_SET

○ Metadata file

○ Common data model folder

● Standard representation of native JavaScript objects

✓ Correct
JSON is a standard representation of native JavaScript objects, and Node.js handles it easily.

9. When an error occurs, which state is the promise at? 1 / 1 point

● Rejected

○ Pending

○ Resolved

○ Aborted

✓ Correct
When there is an error, the promise is in the rejected state.

10. The axios package handles HTTP requests in Node.js and returns a promise object. The promise object has a method that is called after the promise completes. What method does the promise object use? 1 / 1 point

○ or

● then

○ else

○ if

✓ Correct
The promise has a "then" method, which is called after the promise is resolved.