* **1.**

¿Qué es Node.js?

Node.js es un ambiente de ejecución de código, es un proyecto comunitario con una de las comunidades más grandes e involucradas. y ha formado un cambio de paradigma por la forma en la que usamos javascript.

* **2**

De los siguientes calificativos, ¿cuáles aplican a Node.js?

Todos los anteriores

* **3**

¿Cómo se llama el motor de JavaScript del cual depende Node.js para la ejecución de código?

V8

* **4**

¿Cuál término se asocia mejor con Node.js?

Runtime Enviroment

* **5**

Node.js puede ser instalado en una gran diversidad de plataformas en la actualidad como Microsoft y sistemas operativos con base UNIX:

Verdadero

* **6**

Dentro del esquema de versiones de Node.js, ¿Qué significa que una versión sea LTS?

Las siglas LTS vienen del inglés Long Term Support que traduce: Soporte de largo plazo, es estable

* **7**

¿Cuál de los siguientes comandos presentaría en la terminal la actual versión en ejecución de Node.js?

node -v

* **8**

¿Mediante cúal formato se ejecutaría un fichero script.js usando el utilitario de Node.js?

node script.js

* **9**

¿Cuál de los siguientes comandos que utilizan NPM inicializa un nuevo proyecto de Node.js?

npm init

* **10**

¿Con qué comando se puede ejecutar un script definido en el archivo package.json?

npm run nombreScript

* **11**

¿Cuál es el comportamiento esperado al enviar la ruta de un directorio como parámetro a la función require()?

* **12**

Si en el mismo directorio de un fichero llamado a.js existe otro fichero llamado http.js y, desde el fichero a.js se ejecuta la instrucción require(“http”), ¿cuál es el resultado esperado?

Importa el módulo nativo

* **13**

¿Bajo qué nombre/propiedad se guarda la información de paquetes añadidos al proyecto en el fichero package.json?

Dependencies

* **14**

¿Cuál de las siguientes instrucciones instala los paquetes del proyecto de Node.js que se utilizarán durante el desarrollo del proyecto?

npm install

* **15**

¿Qué son las operaciones asíncronas?

Las funciones asíncronas son aquellas funciones que no son de acción bloqueante del hilo del proceso y permite la ejecución de otras instrucciones mientras fonaliza.

* **16**

En contexto de programación en Node.js, una promesa es un objeto de computación definido como estándar para manejar los estados esenciales de operación asíncrona

Verdadero

* **17**

En implementaciones de operaciones asíncronas con promesas, ¿qué propiedad maneja las resoluciones satisfactorias de la operación?

.then()

* **18**

El denotado “callback hell” es visto mayoritariamente en implementaciones de ámbito asíncrono que:

Resolver de forma secuencial. Los callbacks son funciones que se pasan como argumentos y envían valores de retorno.

* **19**

¿Cual de las siguientes opciones representa una correcta implementación de una función asíncrona con async/await? (Asuma que value() devolverá una promesa.):

async function name() { const value = await value()}

* **20**

¿Cuál de las siguientes proposiciones aplica para el operador “async”?

* **21**

¿Qué operación realiza el proceso de eliminación de un fichero en el módulo FileSystem?

fs.unlink

* **22**

Dentro del esquema de funciones dentro del módulo FileSystem, ¿Cuál es el tipo de implementación de asincronismo empleado durante su uso?

* **23**

Con el módulo Readline es posible crear aplicaciones interactivas en ambiente de terminal:

Verdadero

* **24**

¿Qué función del módulo Readline permite realizar una pregunta en la terminal y obtener un dato de retorno?

rl.question

* **25**

Mediante el módulo ChildProcess es posible crear subprocesos que ejecuten diferentes tareas/aplicaciones fuera del ámbito de Node.js:

Verdadero

* **26**

¿Cuál de las siguientes funciones es la que actúa como fundamento en el funcionamiento del módulo ChildProcess?

Esta función la proporciona principalmente la función **child\_process.spawn ()**:

* **27**

¿Qué expresión puede utilizarse en el código fuente para establecer un breakpoint manejable por nuestro debugger?

debugger

* **28**

¿Qué comandos de la consola interactiva DEBUG nos permiten gestionar la vista de expresiones basadas en el contexto de ejecución durante la depuración?

watch (expr) - Añadir expresión a la lista de observación unwatch (expr) - Eliminar expresión de la lista de observación

* **29**

¿Es factible utilizar el mecanismo de try…catch para capturar errores de ejecución en operaciones asíncronas? (Por ejemplo, envolviendo una promesa o un callback):

Falso

* **30**

¿En cuál categoría se clasificaría un intento de apertura de un archivo inexistente en el sistema de archivos?

System error

* **31**

¿Qué función es convocada por los EventEmitters para disparar la ocurrencia de un evento?

.emit()

* **32**

El módulo EventEmitter es uno de los pilares fundamentales de Node.js, ya que sobre este se implementan muchas funciones de sus otros módulos:

Verdadero

* **33**

¿Cuál verbo HTTP es el encargado de representar acciones que modificarán o crearán contenido en el servidor?

Put | post

* **34**

¿Cuál verbo HTTP nos permite solicitar recursos a un servidor web?

.get()

* **35**

En contexto de uso del módulo nativo HTTP de Node.js, ¿qué función se utiliza para crear un servidor web básico?

http.createServer()

* **36**

En contexto de uso del módulo nativo HTTP de Node.js, ¿qué función se utiliza para realizar una petición HTTP POST?

http.request()