



What is the output of following?

// Assume Car and Vehicle are two classes that have been already implemented.

```
Car.prototype.getCar = function(){
  console.log('Hello');
}
```

Car.prototype is updated
updates to Car.prototype are
reflected in all instances of Car
was updated (include constructor function)

```
Car.prototype = {
  constructor: Car,
  getCar: function() { console.log("Hello"); }
};
```

```
Car.prototype = Vehicle.prototype;
```

```
const car = new Car();
```

```
car.getCar();
```

```
Vehicle.prototype = {
  constructor: Vehicle
  // No getCar method here
};
```

Select one:

- ☒ a. An error will be thrown since there is no method as getCar().
- ☐ b. No output.
- ☐ c. Code will execute and print 'Hello'.
- ☐ d. None of the above.

Car and Vehicle are two classes (or constructor functions) that have already been implemented.

```
function Car() {}
```

```
function Vehicle() {}
```

creates a constructor function Car / Vehicle, which will be used to create Car instances.
At this point, Car.prototype / Vehicle.prototype exists

```
Car.prototype = {
  constructor: Car
};
```

```
Vehicle.prototype = {
  constructor: Vehicle
};
```

```
function Car() {}
Car.prototype.getModel = function() { return "Sedan"; };
const car = new Car();
console.log(car.getModel()); // "Sedan"
```

```
Car.prototype.getModel = function() { return "SUV"; };
console.log(car.getModel()); // "SUV"
console.log(car.__proto__.getModel()); // "SUV"
```



NetExam

Sri Lanka Institute of Information Technology

In JavaScript, what does "__proto__" in an Object is pointing to?

Select one:

- ☒ a. To its class (Skelton) Prototype
- ☐ b. There is no property as "__proto__".
- ☐ c. To the Object itself. [this](#)
- ☐ d. To its own unique Prototype object.

__proto__ is a property on every JavaScript object that points to its prototype—the object from which it inherits properties and methods.



14

answered

out of

question

What is the output of the following code?

```
function add(a, b){  
    return a+b;  
}
```

```
console.log(add(3,4,5));
```

third argument (5) is ignored because there's no parameter to receive it.

Select one:

- ☒ a. 12
- ☐ b. 7
- ☐ c. Compilation error.
- ☐ d. undefined.

In a REST service developed for a library has a route for creating books. What should be the URI for adding a new book for a given author? (Select the best suited one)

Select one:

- ☐ a. /v1/author/{author ID}/book/ singular, which is unusual
- ☐ b. /v1/books?author={author ID} query parameter for creation can make the API less intuitive and harder to extend (e.g. need to filter books by other criteria later?).
- ☒ c. /v1/authors/{author ID}/books/
- ☐ d. /v1/authors/{author ID}/create /create: An action verb.

In a REST service developed for a library has a route for creating books. What should be the URI for adding a new book for a given author? (Select the best suited one)

Select one:

- ☒ a. /v1/authors/{author ID}/books/
- ☐ b. /v1/author/{author ID}/book/
- ☐ c. /v1/books?author={author ID}
- ☐ d. /v1/authors/{author ID}/create

Next page



NetExam

Sri Lanka Institute of Information Technology

What is the output of the following code?

```
function add(a, b){  
    return a+b;  
}
```

```
console.log(add(3,4,5));
```

Select one:

- ☐ a. undefined.
- ☒ b. 7
- ☐ c. Compilation error.
- ☐ d. 12



NetExam

Sri Lanka Institute of Information Technology

Question 24

Not yet answered

Marked out of
1.00

Flag question

What is a code review? Select the best answer.

Select one:

- ☐ a. Test the code manually and using tools.
- ☐ b. Use a tool to analyze the code.
- ☒ c. Collaboratively check the code for potential errors and design compliance.
- ☐ d. Build code continuously to check if there are any build failures.



NetExam

Sri Lanka Institute of Information Technology

What is the output of the following code?

```
function add(a, b){  
    return a+b;  
}  
  
console.log(add(3,4,5));
```

Select one:

- ☐ a. undefined.
- ☐ b. 12
- ☐ c. Compilation error.
- ☐ d. 7

What is the main way of creating a static variable in JavaScript?

Select one:

- ☐ a. Define a variable inside the class with the 'var' keyword.
- ☐ b. Add variable to class (Constructor function) itself.
- ☐ c. Add the variable to 'this'.
- ☒ d. Use the 'static' keyword.

shared across all instances and can be accessed without creating an instance of the class.

```
class MyClass {  
  static myStaticVariable = "I am static";  
  
  constructor() {  
    // Instance-specific code  
  }  
}
```

```
console.log(MyClass.myStaticVariable); // Output: "I am  
static"
```

What is/are the correct statement(s) about promises in JavaScript?

1. There is no advantage of using promises over callbacks. ✗
2. Promises make the code more readable. ✓
3. A promise can chain asynchronous tasks to work in a synchronous manner. ✓
4. Promises can make asynchronous (ex: perform NodeJS read file synchronously) operations synchronous. ✗

Select one:

- ☐ a. 1
- ☐ b. 2 and 3
- ☒ c. 2, 3 and 4
- ☐ d. 3



4
answered
out of
question

What is the main difference between a JavaScript object and a JSON object?

1. There is no difference.

2. JavaScript objects can have methods and JSON cannot. ✓

only data like strings, numbers, booleans, arrays, and objects.

3. In JSON attribute names (Object keys) should always be a String. ✓

Select one:

- ☐ a. 1
- ☐ b. 1 and 3
- ☐ c. 2 and 3
- ☒ d. 2



Question 1

Not yet answered

Marked out of
1.00

Flag question

If you want to create a git managed repository from scratch what should be the approach?

starting with nothing—no existing repository, code, or structure—and building it entirely a new.

Select one:

- ☐ a. Pull from repository.
- ☐ b. Create a remote repository and clone it or init a local repository.
- ☒ c. Push to the repository.
- ☐ d. Commit to the repository.



Question 27


Not yet answered


Marked out of
1.00

Flag question

What is the best suited Verb and URI for creating a user in a REST API?

Select one:

- ☐ a. PUT /v1/users 
- ☒ b. POST /v1/users

- ☐ c. ~~CREATE~~ /v1/users
- ☐ d. POST /v1/users/create 



Question 4

Not yet answered

Marked out of
1.00

Flag question

What is the the most accurate difference in arrow functions and normal functions in JavaScript?

Select one:

- ☐ a. No difference
- ☐ b. Arrow functions cannot be used with function pointers (a variable whose value is a function).
- ☒ c. Normal functions have 'this' pointing to the execution context and arrow function 'this' pointing to the surrounding context.
- ☐ d. Number of variables that can be passed to an arrow function is limited

a normal function (defined with function keyword),

normal Functions: Good when you need this to change based on the caller (e.g., object methods).

Arrow Functions: Great when you want this to stay consistent, like in callbacks or event handlers where you need the outer scope's this.

Next page



Question 25

Not yet answered

Marked out of
1.00

Flag question

If you have a class that fetches data from the database and returns the data in HTML format. What is the principle you are violating? Refer to the following pseudo-code.

```
function getDate(){
```

```
    const connection = createConnection(connectionString);
```

```
    const data = connection.getDate();
```

 pseudo-code has two responsibilities: fetching data from a database and formatting it as HTML.

```
    return convertToHtml(data);
```

```
}
```

One class/function could handle database access (e.g., DataFetcher).
Another could handle HTML conversion (e.g., HtmlFormatter).

Select one:

- ☐ a. Liskov substitution principle.
- ☐ b. Interface segregation principle.
- ☐ c. Class content violation principle.
- ☒ d. Single responsibility principle.



Question 20

Not yet answered

Marked out of 1

Flag question

What does it mean by static code analysis?

Select one:

- ☐ a. Testing the code manually.
- ☐ b. Checking the code for errors by another developer.
- ☒ c. Scan code using a tool for any potential erroneous scenarios and bad practices.
- ☐ d. Testing the code manually and automatically.

Static code analysis involves using automated tools to examine the source code (without running it)

identify potential errors, bugs, security vulnerabilities, code smells, and bad practices. tools include ESLint (for JavaScript), SonarQube, or linters



Question 21

Not yet answered

Marked out of 1.00

Flag question

JavaScript (Node JS) is best suited for?

Select one:

- ☐ a. Computational applications ✗
- ☐ b. Image processing applications ✗
- ☒ c. I/O heavy applications
- ☐ d. None of the above

because of its asynchronous, non-blocking I/O model. It's designed to handle many concurrent operations efficiently,

Question 11

Not yet answered

Marked out of
1.00

Flag question

What is the output of the following code?

```
function Vehicle(model){      Vehicle.prototype = { constructor: Vehicle }
    this.model=model;
}

function Car(model){          Car.prototype = {
                                constructor: Car
                                };
}
```

Car.prototype.__proto__ === Vehicle.prototype
Car.prototype = Object.create(Vehicle.prototype); Car instances will inherit properties from Vehicle.prototype,
Car.prototype.constructor = Car; constructor property of Car.prototype points back to Car
var car = new Car('car');
console.log(car.model);

Select one:

- ☐ a. Compilation error.
- ☐ b. car
- ☒ c. undefined
- ☐ d. None of the above



Question 30

Not yet answered

Marked out of 10

Flag question

Which conditions will return true in JavaScript?

1. `1=="1"` ✓2. `1=== "1"` ✗3. `1===1` ✓4. `1==1` ✓

Select one:

- ☐ a. 1 and 4
- ☒ b. 1, 3 and 4
- ☐ c. 3 and 4
- ☐ d. 4



Question 19

Not yet answered

Marked out of
1.00

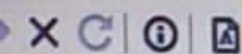
Flag question

What is/are the correct statement(s) about promises in JavaScript?

1. There is no advantage of using promises over callbacks.
2. Promises make the code more readable.
3. A promise can chain asynchronous tasks to work in a synchronous manner.
4. Promises can make asynchronous (ex. perform NodeJS read file synchronously) operations synchronous.

Select one:

- ☐ a. 1
- ☐ b. 2, 3 and 4
- ☐ c. 2 and 3
- ☐ d. 3



NetExam

Sri Lanka Institute of Information Technology

on 21

et answered

ed out of

eg question

What does higher-level modules (in dependency inversion principle)?

Select one:

- ☐ a. DB level modules.
- ☐ b. UI level modules.
- ☒ c. Business logic related modules.
- ☐ d. None of the above.



NetExam

Sri Lanka Institute of Information Technology

Question 5

Not yet answered

Marked out of
1.00

Flag question

You have developed a web application using Node JS. Recently a requirement raised to create a rules engine that needs heavy CPU power. What will be your optimum solution?

Select one:

- ☐ a. Drop the requirement and let them know that it is not feasible. ✗
- ☐ b. Implement it using Node JS without any difference.
- ☐ c. Rewrite the application in Java ✗
- ☐ d. Try it with the latest Node JS features such as workers.

Next page



An application connects to a database cluster with replication, meaning there's a primary (write) node and one or more replica (read) nodes.

14

answered

out of

question

In an application that connects to a DB cluster with replication, an issue has been detected where the data that has been written cannot be fetched immediately. But after few seconds data is available. What may be causing this (most likely)?

Select one:

- ☐ a. Error in the DB driver used in the application. ~~X~~
- ☒ b. DB replication lag (delay in writing the data to replicas/eventual consistency).
- ☐ c. Error in the application logic. ~~X~~
- ☐ d. None of the above.

Not sure

Next page



NetExam

Sri Lanka Institute of Information Technology

2

answered

out of

question

What is the reason setTimeout method always execute on or after the given millisecond value? Mark the correct statement.

Select one:

setTimeout(callback, 0) doesn't mean "execute immediately." It means "queue the callback as soon as possible."

- ☐ a. If you pass 0 as the timeout value, it will execute right away. ✗
- ☐ b. There is no function as such. ✗
- ☒ c. Irrespective of the millisecond value setTimeout function always use event loop and main thread cannot be interrupted.
- ☐ d. All above is incorrect ✗



Question 28

Not yet answered

Marked out of 1.00

Flag question

What does it mean by static code analysis?

Select one:

- ☐ a. Scan code using a tool for any potential erroneous scenarios and bad practices.
- ☐ b. Testing the code manually and automatically.
- ☐ c. Testing the code manually.
- ☐ d. Checking the code for errors by another developer.



22
answered
out of
question

What is the output of the following code?

```
window.name = 'Firefox';  
function printName() {  
    console.log(this.name);  
}  
  
const chrome = {  
    name: 'Chrome',  
    print: printName  
};  
  
printName();  
chrome.print();
```

Select one:

- ☒ a. Firefox, Chrome
- ☐ b. Chrome, Chrome
- ☐ c. Firefox, Firefox
- ☐ d. The code will not run.



Question 6

Not yet answered

Marked out of

1.00

Flag question

When you are creating a class in JavaScript if you have the requirement to create multiple instances from that class what will be the best way you can do that?

Select one:

- ☒ a. Using a constructor function and adding all the variables to 'this' and behaviors to prototype.
- ☐ b. We cannot meet the above requirement in JavaScript.
- ☐ c. Using a constructor function and adding all the variables and behaviors to 'this'
- ☐ d. Using object literals ({}).

```
function Person(name) {  
  this.name = name; // Instance-specific variable  
}  
Person.prototype.sayHello = function() { // Shared behavior  
  return `Hello, I'm ${this.name}`;  
};
```

```
const person1 = new Person("Alice");  
const person2 = new Person("Bob");  
console.log(person1.sayHello()); // "Hello, I'm Alice"  
console.log(person2.sayHello()); // "Hello, I'm Bob"  
console.log(person1.sayHello === person2.sayHello); // true (shared method)
```

Explain |

Not sure

Next page



Question 29

Not yet answered

Marked out of
1.00

Flag question

In an application that connects to a DB cluster with replication, an issue has been detected where the data that has been written cannot be fetched immediately. But after few seconds data is available. What may be causing this (most likely)?

Select one:

- ☐ a. Error in the DB driver used in the application.
- ☐ b. DB replication lag (delay in writing the data to replicas/eventual consistency).
- ☐ c. Error in the application logic.
- ☐ d. None of the above.

Next page



Question 27

Not yet answered

Marked out of

1.00

Flag question

In JavaScript, what does "__proto__" in an Object is pointing to?

Select one:

- ☐ a. To the Object itself.
- ☐ b. To its own unique Prototype object.
- ☐ c. There is no property as "__proto__".
- ☐ d. To its class (Skelton) Prototype.



Question 6

Not yet answered

Marked out of
1.00

Flag question

What is the output of following?

//Assume Car and Vehicle are two classes that have been already implemented.

```
Car.prototype.getCar = function(){  
    console.log("Hello");  
}
```

```
Car.prototype = Vehicle.prototype;
```

```
const car = new Car();
```

```
car.getCar();
```

Select one:

☐ a.

An error will be thrown since there is no method as getCar().

☐ b. No output.

☐ c. Code will execute and print 'Hello'.

☐ d. None of the above.



Question 12

Not yet answered

Marked out of
1.00

Flag question

What is/are the correct statement(s) about promises in JavaScript?

1. There is no advantage of using promises over callbacks.
2. Promises make the code more readable. ✓
3. A promise can chain asynchronous tasks to work in a synchronous manner. ✓
4. Promises can make asynchronous (ex: perform NodeJS read file synchronously) operations synchronous.

Select one:

- ☒ a. 2, 3 and 4
- ☐ b. 2 and 3
- ☐ c. 3
- ☐ d. 1



Question 30

Not yet answered

Marked out of
1.00

Flag question

What is the usage of the ".gitignore" file?

Select one:

- ☐ a. A list of node_modules that should be ignored for tracking.
- ☒ b. A list of all the files and directories that should be ignored for tracking.
- ☐ c. A list of all the branches that should be ignored for tracking.
- ☐ d. To keep some metadata about the repository.

Finish attempt



Question 17

Not yet answered

Marked out of
1.00

Flag question

If you want to create a git managed repository from scratch what should be the approach?

Select one:

- ☐ a. Create a remote repository and clone it or init a local repository.
- ☐ b. Pull from repository
- ☐ c. Commit to the repository.
- ☐ d. Push to the repository.



Question 30

Not yet answered

Marked out of
1.00

Flag question

If you have a class that fetches data from the database and returns the data in HTML format. What is the principle you are violating? Refer to the following pseudo-code.

```
function getDate(){  
    const connection = createConnection(connectionString);  
    const data = connection.getDate();  
    return convertToHtml(data);  
}
```

Select one:

- ☐ a. Class content violation principle.
- ☒ b. Single responsibility principle.
- ☐ c. Interface segregation principle.
- ☐ d. Liskov substitution principle.

≡ Qu

Finish at

Time left

1	2
9	1
17	11
25	26

Finish attempt ...



What is the best HTTP word for updating (selected set of fields not replacing the entire resource) a resource?

Select one:

- ☐ a. OPTIONS
- ☒ b. PATCH
- ☐ c. PUT
- ☐ d. POST

You send only the fields to change, leaving others intact.

```
PATCH /users/123
Content-Type: application/json
{"email": "new@example.com"}
```

PUT method replaces the entire resource with the payload provided

```
PUT /users/123
Content-Type: application/json
{"id": 123, "name": "Alice", "email": "new@example.com"}
```

Question 23

Not yet answered

Marked out of

1.00

Flag question

What is not an asynchronous programming pattern?

Select one:

☐ a. Callbacks ✓

☒ b. push/pop

☐ c. Async/await ✓

☐ d. Promises ✓



Question 8

Not yet answered

Marked out of
1.00

Flag question

What is the reason setTimeout method always execute on or after the given millisecond value? Mark the correct statement.

Select one:

- ☒ a. If you pass 0 as the timeout value, it will execute right away.
- ☐ b. There is no function as such.
- ☐ c. Irrespective of the millisecond value setTimeout function always use event loop and main thread cannot be interrupted.
- ☐ d. All above is incorrect

Next page



Question 7

Not yet answered

Marked out of 1.00

Flag question

What is not an asynchronous programming pattern?

Select one:

- ☐ a. push/pop
- ☐ b. Async/await
- ☐ c. Promises
- ☐ d. Callbacks



Question 20

Not yet answered

Marked out of
1.00

Flag question

Consider the following statements.

```
let a =
```

```
if(a) {
```

```
}
```

To evaluate the above condition to false, what is the possible value for variable 'a'?

Select one:

- ☐ a. var a = 1
- ☐ b. var a = 'false'
- ☒ c. var a="" consider as empty string "".
- ☐ d. None of the above

In JavaScript, the following values are falsy (coerce to `false`):

- `false`
- `0`
- `-0`
- `0n` (BigInt zero)
- `""` (empty string)
- `null`
- `undefined`
- `NaN`

All other values are **truthy** (coerce to `true`), including:

- Non-zero numbers (e.g., `1`, `-1`)
- Non-empty strings (e.g., `"false"`, `"0"`)
- Objects (e.g., `{}`)
- Arrays (e.g., `[]`)



Question **21**

Not yet answered

Marked out of
1.00

Flag question

What is a code review? Select the best answer.

Select one:

- ☐ a. Collaboratively check the code for potential errors and design compliance.
- ☐ b. Build code continuously to check if there are any build failures.
- ☐ c. Test the code manually and using tools.
- ☐ d. Use a tool to analyze the code.



Question 13

Not yet answered

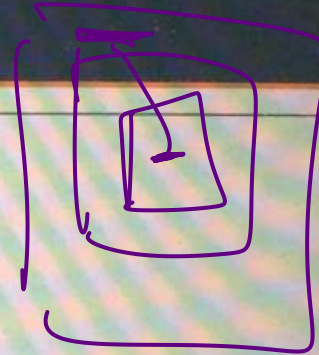
Marked out of 1.00

Flag question

What does "Closure" means in JavaScript?

Select one:

- ☐ a. Define a function in one place and pass it to anywhere and use it
- ☐ b. Completed function without any flaws
- ☒ c. The nested function ability to access its parent function context anywhere it's being used. outer scope's variables
- ☐ d. A nested function



```
function createCounter() {  
  let value = 0;  
  return function() {  
    return ++value;  
  };  
}  
  
const counter = createCounter();  
console.log(counter()); // 1  
console.log(counter()); // 2
```

Next page



Question 3

Not yet answered

Marked out of
1.00

Flag question

JavaScript (Node JS) is best suited for?

Select one:

- ☐ a. Computational applications
- ☐ b. Image processing applications
- ☐ c. I/O heavy applications
- ☐ d. None of the above



covered
of
question

You have developed a web application using Node JS. Recently a requirement raised to create a rules engine that needs heavy CPU power. What will be your optimum solution?

Select one:

- ☐ a. Try it with the latest Node JS features such as workers.
- ☐ b. Drop the requirement and let them know that it is not feasible.
- ☐ c. Rewrite the application in Java.
- ☐ d. Implement it using Node JS without any difference.

Next page