

# Salesforce

## Exam Questions JavaScript-Developer-I

Salesforce Certified JavaScript Developer I



### NEW QUESTION 1

A developer receives a comment from the Tech Lead that the code given below has error:

```
const monthName = 'July'; const year = 2019;
if(year === 2019) { monthName = 'June';
}
```

Which line edit should be made to make this code run?

- A. 01 let monthName ='July';
- B. 02 let year =2019;
- C. 02 const year = 2020;
- D. 03 if (year == 2019) {

Answer: A

### NEW QUESTION 2

A developer wants to use a try...catch statement to catch any error that countSheep () may throw and pass it to a handleError () function. What is the correct implementation of the try...catch?

A)

```
try {
  setTimeout(function() {
    countSheep();
  }, 1000);
} catch (e) {
  handleError(e);
}
```

B)

```
try {
  countSheep();
} finally {
  handleError(e);
}
```

C)

```
setTimeout(function() {
  try {
    countSheep();
  } catch (e) {
    handleError(e);
  }
}, 1000);
```

D)

```
try {
  countSheep();
} handleError (e){
  catch(e):
```

- A. Option
- B. Option
- C. Option
- D. Option

Answer: A

### NEW QUESTION 3

A developer is creating a simple webpage with a button. When a user clicks this button for the first time, a message is displayed.

The developer wrote the JavaScript code below, but something is missing. The message gets displayed every time a user clicks the button, instead of just the first time.

```
01 function listen(event) {
02
03   alert('Hey! I am John Doe');
04
05 }
06 button.addEventListener('click', listen);
```

Which two code lines make this code work as required? Choose 2 answers

- A. On line 02, use event.first to test if it is the first execution.
- B. On line 04, use event.stopPropagation (),
- C. On line 04, use button.removeEventListener(' click" , listen);

D. On line 06, add an option called once to button.addEventListener().

Answer: CD

**NEW QUESTION 4**

Refer to the code:

```
01 function Vehicle(name, price) {
02   this.name = name;
03   this.price = price;
04 }
05 Vehicle.prototype.priceInfo = function () {
06   return 'Cost of the ${this.name} is ${this.price}$';
07 }
08 var ford = new Vehicle('Ford Fiesta', '20,000');
```

Given the requirement to refactor the code above to JavaScript class format, which class definition is correct?

A)

```
01 class Vehicle {
02   constructor(name, price) {
03     this.name = name;
04     this.price = price;
05   }
06   priceInfo() {
07     return 'Cost of the ${this.name} is ${this.price}$';
08   }
09 }
```

B)

```
01 class Vehicle {
02   vehicle(name, price) {
03     this.name = name;
04     this.price = price;
05   }
06   priceInfo() {
07     return 'Cost of the ${this.name} is ${this.price}$';
08   }
09 }
```

C)

```
01 class Vehicle {
02   constructor(name, price) {
03     name = name;
04     price = price;
05   }
06   priceInfo() {
07     return 'Cost of the ${this.name} is ${this.price}$';
08   }
09 }
```

D)

```
01 class Vehicle {
02   constructor() {
03     this.name = name;
04     this.price = price;
05   }
06   priceInfo() {
07     return 'Cost of the ${this.name} is ${this.price}$';
08   }
09 }
```

A.

Answer: B

**NEW QUESTION 5**

Given the code below: Function myFunction(){ A =5;  
 Var b =1;

```
}
myFunction(); console.log(a); console.log(b);
```

What is the expected output?

- A. Both lines 08 and 09 are executed, and the variables are outputted.
- B. Line 08 outputs the variable, but line 09 throws an error.
- C. Line 08 thrones an error, therefore line 09 is never executed.
- D. Both lines 08 and 09 are executed, but values outputted are undefined.

**Answer: B**

#### NEW QUESTION 6

A developer wrote the following code to test a sum3 function that takes in an array of numbers and returns the sum of the first three numbers in the array, and the test passes.

A different developer made changes to the behavior of sum3 to instead sum only the first two numbers present in the array.

```
01 let res = sum3([1, 4, 1]);
02 console.assert(res === 6);
03
04 res = sum3([1, 5, 0, 5]);
05 console.assert(res === 6);
```

Which two results occur when running this test on the updated sum3 function? Choose 2 answers

- A. The line 05 assertion passes.
- B. The line 02 assertion passes.
- C. The line 02 assertion fails.
- D. The line 05 assertion fails.

**Answer: BD**

#### NEW QUESTION 7

Given the following code: Let x = null; console.log(typeof x);  
 What is the output of the line 02?

- A. "Null"
- B. "X"
- C. "Object"
- D. "undefined"

**Answer: C**

#### NEW QUESTION 8

A developer wants to use a module named universalContainersLib and then call functions from it.  
 How should a developer import every function from the module and then call the functions foo and bar?

- A. import \* from '/path/universalContainersLib.js'; universalContainersLi
- B. foo ()? universalContainersLib.bar ();
- C. import {foo,bar} from '/path/universalCcontainersLib.js'; foo():bar()?
- D. import all from '/path/universalContainersLib.js'; universalContainersLib.foo(); universalContainersLib.bar ();
- E. import \* as lib from '/path/universalContainersLib.js'; lib.foo();li
- F. bar ();

**Answer: D**

#### NEW QUESTION 9

Refer to following code block:

```
Let array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,];
Let output =0;
For (let num of array){ if (output >0){
Break;
}
if(num % 2 == 0){
Continue;
}
Output +=num;
```

What is the value of output after the code executes?

- A. 16
- B. 36
- C. 11
- D. 25

**Answer: A**

#### NEW QUESTION 10

Refer to the following object:

```
01 const cat = {
02   firstName: 'Fancy',
03   lastName: 'Whiskers',
04   get fullName() {
05     return this.firstName + ' ' + this.lastName;
06   }
07 };
```

How can a developer access the fullName property for cat?

- A. cat.fullName
- B. cat.fullName()
- C. cat.get.fullName
- D. cat.function.fullName()

**Answer: A**

#### NEW QUESTION 11

developer creates a new web server that uses Node.js. It imports a server library that uses events and callbacks for handling server functionality. The server library is imported with require and is made available to the code by a variable named server. The developer wants to log any issues that the server has while booting up. Given the code and the information the developer has, which code logs an error at boost with an event?

- A. Server.catch ((server) => { console.log('ERROR', error);});
- B. Server.error ((server) => { console.log('ERROR', error);});
- C. Server.on ('error', (error) => { console.log('ERROR', error);});
- D. Try{server.start();} catch(error) { console.log('ERROR', error);}

**Answer: C**

#### NEW QUESTION 12

developer removes the HTML class attribute from the checkout button, so now it is simply:  
 <button>Checkout</button>.

There is a test to verify the existence of the checkout button, however it looks for a button with class= "blue". The test fails because no such button is found. Which type of test category describes this test?

- A. True positive
- B. True negative
- C. False positive
- D. False negative

**Answer: D**

#### NEW QUESTION 13

Refer to the code below:

```
01 function foo() {
02   const a = 2;
03   function bar() {
04     console.log(a);
05   }
06   return bar;
07 }
```

Why does the function bar have access to variable a ?

- A. Inner function's scope
- B. Hoisting
- C. Outer function's scope
- D. Prototype chain

**Answer: C**

#### NEW QUESTION 14

A developer creates a generic function to log custom messages in the console. To do this, the function below is implemented.

```
01 function logStatus(status) {
02   console./*Answer goes here*/('Item status is: %s', status);
03 }
```

Which three console logging methods allow the use of string substitution in line 02?

- A. Assert
- B. Log
- C. Message

- D. Info
- E. Error

**Answer:** AD

#### NEW QUESTION 15

A developer is working on an ecommerce website where the delivery date is dynamically calculated based on the current day. The code line below is responsible for this calculation. `const deliveryDate = new Date ();`

Due to changes in the business requirements, the delivery date must now be today's date + 9 days. Which code meets this new requirement?

- A. `deliveryDate.setDate(( new Date ( )).getDate () +9);`
- B. `deliveryDate.setDate( Date.current () + 9);`
- C. `deliveryDate.date = new Date(+9) ;`
- D. `deliveryDate.date = Date.current () + 9;`

**Answer:** A

#### NEW QUESTION 16

What are two unique features of functions defined with a fat arrow as compared to normal function definition?

Choose 2 answers

- A. The function generated its own `this` making it useful for separating the function's scope from its enclosing scope.
- B. The function receives an argument that is always in scope, called `parentThis`, which is the enclosing lexical scope.
- C. If the function has a single expression in the function body, the expression will be evaluated and implicitly returned.
- D. The function uses the `this` from the enclosing scope.

**Answer:** AC

#### NEW QUESTION 17

A developer is required to write a function that calculates the sum of elements in an array but is getting `undefined` every time the code is executed. The developer needs to find what is missing in the code below.

```
const sumFunction = arr => {
  return arr.reduce((result, current) => {
    //
    result += current;
    //
  }, 10);
};
```

Which option makes the code work as expected?

- A. Replace line 02 with `return arr.map(( result, current) => (`
- B. Replace line 04 with `result = result +current;`
- C. Replace line 03 with `if(arr.length == 0 ) ( return 0; )`
- D. Replace line 05 with `return result;`

**Answer:** D

#### NEW QUESTION 18

Refer to the following code that imports a module named `utils`:

```
01 import {foo,bar} from '/path/Utils.js';
02 foo();
03 bar();
```

Which two implementations of `Utils.js` export `foo` and `bar` such that the code above runs without error?

Choose 2 answers

- A. 

```
//FooUtils.js and BarUtils.js exist
import {foo} from '/path/FooUtils.js';
import {bar} from '/path/BarUtils.js';
export {foo, bar}
```
- B. 

```
const foo = () => { return 'foo'; }
const bar = () => { return 'bar'; }
export {foo, bar}
```
- C. 

```
export default class {
  foo() { return 'foo'; }
  bar() { return 'bar'; }
}
```
- D.

```
const foo = () => { return 'foo'; }  
const bar = () => { return 'bar'; }  
export default foo, bar;
```

Answer: BC

#### NEW QUESTION 19

developer wants to use a module named universalContainersLib and then call functions from it.  
How should a developer import every function from the module and then call the functions foo and bar ?

- A. import \* as lib from '/path/universalContainersLib.js'; lib.foo();lib.bar();
- B. import {foo, bar} from '/path/universalContainersLib.js'; foo();bar();
- C. import all from '/path/universalContainerLib.js'; universalContainersLib.foo(); universalContainersLib.bar();
- D. import \* from '/path/universalContainerLib.js'; universalContainersLib.foo(); universalContainersLib.bar();

Answer: A

#### NEW QUESTION 20

A developer has a web server running with Node.js. The command to start the web server is node server.js.  
The web server started having  
latency issues. Instead of a one second turnaround for web requests, the developer now sees a five second turnaround.  
Which command can the web developer run to see what the module is doing during the latency period?

- A. NODE\_DEBUG=true node server.js
- B. DEBUG=http, https node server.js
- C. NODE\_DEBUG=http,https node server.js
- D. DEBUG=true node server.js

Answer: D

#### NEW QUESTION 21

.....

## **Thank You for Trying Our Product**

### **We offer two products:**

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

### **JavaScript-Developer-I Practice Exam Features:**

- \* JavaScript-Developer-I Questions and Answers Updated Frequently
- \* JavaScript-Developer-I Practice Questions Verified by Expert Senior Certified Staff
- \* JavaScript-Developer-I Most Realistic Questions that Guarantee you a Pass on Your First Try
- \* JavaScript-Developer-I Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

**100% Actual & Verified — Instant Download, Please Click**  
**[Order The JavaScript-Developer-I Practice Test Here](#)**