

# Salesforce

## Exam Questions JavaScript-Developer-I

Salesforce Certified JavaScript Developer I



#### NEW QUESTION 1

Refer to the code snippet:

```
01 function getAvailabilityMessage(item) {
02     if (getAvailability(item)) {
03         var msg = "Username available";
04     }
05     return msg;
06 }
```

A developer writes this code to return a message to user attempting to register a new username. If the username is available, variable. What is the return value of msg hen getAvailabilityMessage ("newUserName" ) is executed and getAvailability("newUserName") returns false?

- A. "Username available"
- B. "newUserName"
- C. "Msg is not defined"
- D. undefined

**Answer: D**

#### NEW QUESTION 2

Refer to the following code:

Let sampleText = 'The quick brown foxjumps';

A developer needs to determine if a certain substring is part of a string. Which three expressions return true for the given substring ?

Choose 3 answers

- A. sampleText.includes('fox');
- B. sampleText.includes(' quick ', 4);
- C. sampleText.includes('Fox ', 3)
- D. sampleText.includes(' fox ');
- E. sampleText.includes(' quick ') !== -1;

**Answer: BDE**

#### NEW QUESTION 3

A developer is wondering whether to use, Promise.then or Promise.catch, especially when a Promise throws an error?

Which two promises are rejected? Which 2 are correct?

- A. Promise.reject('cool error here').then(error => console.error(error));
- B. Promise.reject('cool error here').catch(error => console.error(error));
- C. New Promise((resolve, reject) => (throw 'cool error here')).catch(error => console.error(error)) ;
- D. New Promise(() => (throw 'cool error here')).then(null, error => console.error(error));

**Answer: BC**

#### NEW QUESTION 4

Refer to the code below:

```
01 let o = {
02     get js() {
03         let city1 = String('St.Louis');
04         let city2 = String('New York');
05
06         return {
07             firstCity: city1.toLowerCase(),
08             secondCity: city2.toLowerCase(),
09         }
10     }
11 }
```

What value can a developer expect when referencing o.js.secondCity?

- A. Undefined
- B. ' new york '
- C. ' New York '
- D. An error

**Answer: B**

#### NEW QUESTION 5

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 6

A developer has the following array of student test grades: Let arr = [ 7, 8, 5, 8, 9 ];  
 The Teacher wants to double each score and then see an array of the students who scored more than 15 points.  
 How should the developer implement the request?

- A. Let arr1 = arr.filter(( val ) => ( return val > 15 )) .map (( num) => ( return num \*2 ))
- B. Let arr1 = arr.mapBy (( num) => ( return num \*2 )) .filterBy (( val ) => return val > 15 );
- C. Let arr1 = arr.map((num) => num\*2). Filter (( val ) => val > 15);
- D. Let arr1 = arr.map((num) => ( num \*2)).filterBy((val) => ( val >15 ));

Answer: C

#### NEW QUESTION 7

Refer to code below:

```
Function muFunction(reassign){
  Let x = 1;
  var y = 1;
  if( reassign ) {
  Let x= 2;
  Var y = 2;
  console.log(x);
  console.log(y);}
  console.log(x);
  console.log(y);}
```

What is displayed when myFunction(true) is called?

- A. 2 2 1 1
- B. 2 2 undefined undefined
- C. 2 2 1 2
- D. 2 2 2 2

Answer: C

#### NEW QUESTION 8

A developer is setting up a new Node.js server with a client library that is built using events and callbacks. The library:  
 \* Will establish a web socket connection and handle receipt of messages to the server  
 \* Will be imported with require, and made available with a variable called we.

The developer also wants to add error logging if a connection fails.

Given this info, which code segment shows the correct way to set up a client with two events that listen at execution time?

- A. 

```
04 ws.connect(() => {
05   console.log('Connected to client');
06 }).catch((error) => {
07   console.log('ERROR', error);
08 });
```
- B. 

```
04 ws.on('connect', () => {
05   console.log('Connected to client');
06
07   ws.on('error', (error) => {
08     console.log('ERROR', error);
09   });
10 });
```
- C. 

```
04 ws.on('connect', () => {
05   console.log('Connected to client');
06 });
07
08 ws.on('error', (error) => {
09   console.log('ERROR', error);
10 });
```
- D. 

```
04 try {
05   ws.connect(() => {
06     console.log('Connected to client');
07   });
08 } catch(error) {
09   console.log('ERROR', error);
10 }
```

**Answer: C**

#### NEW QUESTION 9

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 10

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 number in the array, The test passes:

```

01 let res = sum3([1, 2, 3]);
02 console.assert(res === 6);
03
04 res=sum3([1, 2, 3, 4]);
05 console.assert(res === 6);

```

A different developer made changes to the behavior of sum3 to instead sum all of the numbers present in the array. The test passes:  
 Which two results occur when running the test on the updated sum3 function ? Choose 2 answers

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

**Answer: AD**

#### NEW QUESTION 10

Refer to the code below:

```

Let car1 = new Promise((_, reject) => setTimeout(reject, 2000, "car 1 crashed in" =>
Let car2 =new Promise(resolve => setTimeout(resolve, 1500, "car 2 completed") Let car3 =new Promise(resolve => setTimeout(resolve, 3000, "car 3 completed")
Promise.race(( car1, car2, car3))
.then (value => (
Let result = `${value} the race.`;)}
.catch(arr => {
console.log("Race is cancelled.", err);
});

```

What is the value of result when Promise.race executes?

- A. Car 3 completes the race
- B. Car 2 completed the race.
- C. Car 1 crashed in the race.
- D. Race is cancelled.

Answer: B

#### NEW QUESTION 11

Given the following code: `document.body.addEventListener('click', (event) => { if (/* CODE REPLACEMENT HERE */) { console.log('button clicked!'); } });`

Which replacement for the conditional statement on line 02 allows a developer to correctly determine that a button on page is clicked?

- A. `Event.clicked`
- B. `e.nodeTarget == this`
- C. `event.target.nodeName == 'BUTTON'`
- D. `button.addEventListener('click')`

Answer: C

#### NEW QUESTION 16

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 throws 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 20

Refer to the code below: `Const pi = 3.1415326`, What is the data type of pi?

- A. Double
- B. Number
- C. Decimal
- D. Float

Answer: B

#### NEW QUESTION 25

Refer to HTML below:

`<p> The current status of an Order: <span id="status"> In Progress </span> </p>`

Which JavaScript statement changes the text 'In Progress' to 'Completed' ?

- A. `document.getElementById("status").Value = 'Completed' ;`
- B. `document.getElementById("#status").innerHTML = 'Completed' ;`
- C. `document.getElementById("status").innerHTML = 'Completed' ;`
- D. `document.getElementById(".status").innerHTML = 'Completed' ;`

Answer: C

#### NEW QUESTION 27

The developer wants to test this code:

`Const toNumber =(strOrNum) => strOrNum;` Which two tests are most accurate for this code? Choose 2 answers

- A. `console.assert(toNumber('2') ===2);`
- B. `console.assert(Number.isNaN(toNumber()));`
- C. `console.assert(toNumber('-3') < 0);`
- D. `console.assert(toNumber () === NaN);`

Answer: AC

#### NEW QUESTION 28

Which three actions can be using the JavaScript browser console? Choose 3 answers:

- A. View and change DOM the page.
- B. Display a report showing the performance of a page.
- C. Run code that is not related to page.
- D. view , change, and debug the JavaScript code of the page.
- E. View and change security cookies.

Answer: ACD

#### NEW QUESTION 32

Which code statement correctly retrieves and returns an object from localStorage?

- A. `const retrieveFromLocalStorage = () =>{return JSON.stringify(window.localStorage.getItem(storageKey));}` `const retrieveFromLocalStorage = () => { return JSON.stringify(window.localStorage.getItem(storageKey)); }`
- B. `const retrieveFromLocalStorage = (storageKey) =>{ return window.localStorage.getItem(storageKey);}` `const retrieveFromLocalStorage = (storageKey) => { return window.localStorage.getItem(storageKey); }`
- C. `const retrieveFromLocalStorage = (storageKey) =>{ returnJSON.parse(window.localStorage.getItem(storageKey));}` `const retrieveFromLocalStorage = (storageKey) => { return JSON.parse(window.localStorage.getItemstorageKey); }`
- D. `const retrieveFromLocalStorage = (storageKey) =>{ return window.localStorage[storageKey];}` `const retrieveFromLocalStorage = (storageKey) => { return window.localStorage[storageKey]; }`

**Answer: C**

**NEW QUESTION 35**

Refer to HTML below:

```
<div id ="main">
<div id = " card-00">This card is smaller.</div>
<div id = "card-01">The width and height of this card is determined by its contents.</div>
</div>
```

Which expression outputs the screen width of the element with the ID card-01?

- A. `document.getElementById(' card-01 ').getBoundingClientRect().width`
- B. `document.getElementById(' card-01 ').style.width`
- C. `document.getElementById(' card-01 ').width`
- D. `document.getElementById(' card-01 ').innerHTML.lenght*e`

**Answer: A**

**NEW QUESTION 37**

Given the code below:

```
const copy = JSON.stringify([ new String(' false '), new Boolean( false ), undefined ]);
```

What is the value of copy?

- A. `-- [ \"false\" , { } ]-`
- B. `-- [ false, { } ]-`
- C. `-- [ \"false\" , false, undefined ]-`
- D. `-- [ \"false\" ,false, null ]-`

**Answer: D**

**NEW QUESTION 40**

A test has a dependency on database. query. During the test, the dependency is replaced with an object called database with the method, Calculator query, that returns an array. The developer does not need to verify how many times the method has been called. Which two test approaches describe the requirement? Choose 2 answers

- A. White box
- B. Stubbing
- C. Black box
- D. Substitution

**Answer: AD**

**NEW QUESTION 45**

developer publishes a new version of a package with new features that do not break backward compatibility. The previous version number was 1.1.3. Following semantic versioning format, what should the new package version number be?

- A. 2.0.0
- B. 1.2.3
- C. 1.1.4
- D. 1.2.0

**Answer: D**

**NEW QUESTION 49**

A developer wants to leverage a module to print a price in pretty format, and has imported a method as shown below:

```
Import printPrice from '/path/PricePrettyPrint.js';
```

Based on the code, what must be true about the printPrice function of the PricePrettyPrint module for this import to work ?

- A. printPrice must be a named export
- B. printPrice must be an all export
- C. printPrice must be the default export
- D. printPrice must be a multi export

**Answer: C**

#### NEW QUESTION 54

Refer to the code below:

```
01 const exec = (item, delay) =>
02   new Promise(resolve => setTimeout(() => resolve(item), delay));
03
04 async function runParallel() {
05   const [result1, result2, result3] = await Promise.all(
06     [exec('x', '100'), exec('y', '500'), exec('z', '100')]
07   );
08   return 'parallel is done: ${result1}${result2}${result3}';
09 }
```

Which two statements correctly execute the runParallel () function? Choose 2 answers

- A. Async runParallel () .then(data);
- B. runParallel ( ) . done(function(data){ return data;});
- C. runParallel () .then(data);
- D. runParallel ( ) .then(function(data) return data

**Answer:** BD

#### NEW QUESTION 55

Universal Containers (UC) notices that its application that allows users to search for accounts makes a network request each time a key is pressed. This results in too many requests for the server to handle.

Address this problem, UC decides to implement a debounce function on string change handler.

What are three key steps to implement this debounce function? Choose 3 answers:

- A. If there is an existing setTimeout and the search string change, allow the existing setTimeout to finish, and do not enqueue a new setTimeout.
- B. When the search string changes, enqueue the request within a setTimeout.
- C. Ensure that the network request has the property debounce set to true.
- D. If there is an existing setTimeout and the search string changes, cancel the existing setTimeout using the persisted timerId and replace it with a new setTimeout.
- E. Store the timerId of the setTimeout last enqueued by the search string change handle.

**Answer:** ABC

#### NEW QUESTION 59

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 62

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 63

Universal Container(UC) just launched a new landing page, but users complain that the website is slow. A developer found some functions that cause this problem. To verify this, the developer decides to do everything and log the time each of these three suspicious functions consumes.  
console.time('Performance'); maybeAHeavyFunction(); thisCouldTakeTooLong(); orMaybeThisOne(); console.endTime('Performance');  
Which function can the developer use to obtain the time spent by every one of the three functions?

- A. console.timeLog()
- B. console.getTime()
- C. console.trace()
- D. console.timeStamp()

Answer: A

### NEW QUESTION 67

Refer to the following code:

```
01 let obj = {  
02   foo: 1,  
03   bar: 2  
04 }  
05 let output = [];  
06  
07 for (let something in obj) {  
08   output.push(something);  
09 }  
10  
11 console.log(output);
```

What is the output line 11?

- A. [1,2]
- B. ["bar","foo"]
- C. ["foo","bar"]
- D. ["foo:1","bar:2"]

Answer: C

### NEW QUESTION 70

Refer to the code below:

Let foodMenu1 =['pizza', 'burger', 'French fries']; Let finalMenu = foodMenu1; finalMenu.push('Garlic bread');  
What is the value of foodMenu1 after the code executes?

- A. [ 'pizza', 'Burger', 'French fires', 'Garlic bread']
- B. [ 'pizza', 'Burger', 'French fires']
- C. [ 'Garlic bread', 'pizza', 'Burger', 'French fires' ]
- D. [ 'Garlic bread']

Answer: B

### NEW QUESTION 72

A developer uses a parsed JSON string to work with user information as in the block below:

```
01 const userInfo = {  
02   "id": "user-01",  
03   "email": "user01@universalcontainers.demo",  
04   "age": 25
```

Which two options access the email attribute in the object? Choose 2 answers

- A. userInfo("email")
- B. userInfo.get("email")
- C. userInfo.email
- D. userInfo(email)

Answer: AC

### NEW QUESTION 76

Given the code below:

```
const delay = async delay => {
  return new Promise((resolve, reject) => {
    setTimeout(resolve, delay);
  });
};

const callDelay = async () => {
  const yup = await delay(1000);
  console.log(1);
};

console.log(2);
callDelay();
console.log(3);
```

What is logged to the console?

- A. 1 2 3
- B. 1 3 2
- C. 2 1 3
- D. 2 3 1

**Answer: D**

#### NEW QUESTION 78

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 83

Refer to code below: console.log(0);

setTimeout(() => ( console.log(1);

));

console.log(2); setTimeout(() => { console.log(3);

), 0);

console.log(4);

In which sequence will the numbers be logged?

- A. 01234
- B. 02431
- C. 02413
- D. 13024

**Answer: C**

#### NEW QUESTION 87

Refer to the code:

```
01 function Animal(size, type) {
02   this.size = size || 'small';
03   this.type = type || 'Animal';
04   this.canTalk = false;
05 }
06
07 let Pet = function(size, type, name, owner) {
08   Animal.call(this, size, type);
09   this.name = name;
10   this.owner = owner;
11 }
12
13 Pet.prototype = Object.create(Animal.prototype);
14 let pet1 = new Pet();
```

Given the code above, which three properties are set pet1? Choose 3 answers:

- A. Name

- B. canTalk
- C. Type
- D. Owner
- E. Size

**Answer:** BCE

**NEW QUESTION 89**

Refer to the code below:

```
for(let number =2 ; number <= 5 ; number += 1 ) {
// insert code statement here
}
```

The developer needs to insert a code statement in the location shown. The code statement has these requirements:

- \* 1. Does require an import
- \* 2. Logs an error when the boolean statement evaluates to false
- \* 3. Works in both the browser and Node.js

Which meet the requirements?

- A. `assert (number % 2 === 0);`
- B. `console.error(number % 2 === 0);`
- C. `console.debug(number % 2 === 0);`
- D. `console.assert(number % 2 === 0);`

**Answer:** B

**NEW QUESTION 91**

Universal Containers (UC) just launched a new landing page, but users complain that the website is slow. A developer found some functions any that might cause this problem. To verify this, the developer decides to execute everything and log the time each of these three suspicious functions consumes.

Which function can the developer use to obtain the time spent by every one of the three functions?

```
01 console.time('Performance');
02
03 maybeAHeavyFunction();
04
05 thisCouldTakeTooLong();
06
07 orMaybeThisOne();
08
09 console.endTime('Performance');
```

- A. `consol`
- B. `timeLog ()`
- C. `console.timeStamp ()`
- D. `console.trace()`
- E. `console.getTime ()`

**Answer:** A

**NEW QUESTION 92**

```
console.log("start");
Promise.resolve("Success") .then(function(value){
console.log('Success');
});
console.log('End');
```

What is the output after the code executes successfully?

- A. EndStart Success
- B. StartSuccess End
- C. StartEnd Success
- D. Success Start End

**Answer:** C

**NEW QUESTION 93**

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 94

A developer implements and calls the following code when an application state change occurs: `Const onStateChange =innerPageState) => { window.history.pushState(newPageState, ' ', null); }`

If the back button is clicked after this method is executed, what can a developer expect?

- A. A navigate event is fired with a state property that details the previous application state.
- B. The page is navigated away from and the previous page in the browser's history is loaded.
- C. The page reloads and all Javascript is reinitialized.
- D. A popstate event is fired with a state property that details the application's last state.

**Answer: B**

#### NEW QUESTION 95

A test has a dependency on `database.query`. During the test the dependency is replaced with an object called `database` with the method, `query`, that returns an array. The developer needs to verify how many times the method was called and the arguments used each time.

Which two test approaches describe the requirement? Choose 2 answers

- A. Integration
- B. Black box
- C. White box
- D. Mocking

**Answer: CD**

#### NEW QUESTION 99

A developer wrote a fizzbuzz function that when passed in a number, returns the following:

'Fizz' if the number is divisible by 3. 'Buzz' if the number is divisible by 5.

'Fizzbuzz' if the number is divisible by both 3 and 5. Emptystring if the number is divisible by neither 3 or 5.

Which two test cases will properly test scenarios for the fizzbuzz function? Choose 2 answers

- A. `let res = fizzbuzz(5); console.assert ( res === ' ' );`
- B. `let res = fizzbuzz(15);console.assert ( res === ' fizzbuzz ' )`
- C. `let res = fizzbuzz(Infinity); console.assert ( res === ' ' )`
- D. `let res = fizzbuzz(3); console.assert ( res === ' buzz ' )`

**Answer: BCD**

#### NEW QUESTION 101

Refer to the code below:

```
new Promise((resolve, reject) => { const fraction = Math.random();
if( fraction >0.5) reject("fraction > 0.5, " + fraction); resolve(fraction);
})
.then(() =>console.log("resolved"))
.catch((error) => console.error(error))
.finally(() => console.log(" when am I called?"));
```

```
new Promise((resolve, reject) => {
  const fraction = Math.random();
  if( fraction >0.5) reject("fraction > 0.5, " + fraction);
  reject(fraction);
})
.then(() =>console.log("resolved"))
.catch((error) => console.error(error))
.finally(() => console.log(" when am I called?"));
0.024493713600408196 VM1560:7
when am I called? VM1560:8
Promise {<fulfilled>: undefined}
```

```
new Promise((resolve, reject) => {
  const fraction = Math.random();
  if( fraction >0.5) reject("fraction > 0.5, " + fraction);
  resolve(fraction);
})
.then(() =>console.log("resolved"))
.catch((error) => console.error(error))
.finally(() => console.log(" when am I called?"));
resolved VM1349:6
when am I called? VM1349:8
Promise {<fulfilled>: undefined}
```

When does Promise.finally on line 08 get called?

- A. When rejected
- B. When resolved and settled
- C. When resolved
- D. When resolved or rejected

Answer: D

**NEW QUESTION 106**

A developer creates a simple webpage with an input field. When a user enters text in the input field and clicks the button, the actual value of the field must be displayed in the console. Here is the HTML file content:

```
<input type = " text" value="Hello" name ="input">
<button type ="button" >Display </button>
The developer wrote the javascript code below:
Const button = document.querySelector('button'); button.addEvenListener('click', () => (
Const input = document.querySelector('input'); console.log(input.getAttribute('value'));
```

When the user clicks the button, the output is always "Hello". What needs to be done make this code work as expected?

- A. Replace line 04 with console.log(input .value);
- B. Replace line 03 with const input = document.getElementByName('input');
- C. Replace line 02 with button.addEventListener("onclick", function() {
- D. Replace line 02 with button.addCallback("click", function() {

Answer: A

**NEW QUESTION 108**

Refer to the following code:

```
01 function test(val) {
02   if (val === undefined) {
03     return 'Undefined value!';
04   }
05   if (val === null) {
06     return 'Null value!';
07   }
08   return val;
09 }
10
11 let x;
12
13 test(x);
```

What is returned by the function call on line 13?

- A. Undefined
- B. Line 13 throws an error.
- C. 'Undefined values!'
- D. 'Null value!'

**Answer:** A

**NEW QUESTION 110**

Which three options show valid methods for creating a fat arrowfunction? Choose 3 answers

- A. `x => ( console.log(' executed ' ) ; )`
- B. `[ ] => ( console.log(' executed ' ) ; )`
- C. `( ) => ( console.log(' executed ' ) ; )`
- D. `X,y,z => ( console.log(' executed ' ) ; )`
- E. `(x,y,z) => ( console.log(' executed ' ) ; )`

**Answer:** AE

**NEW QUESTION 115**

Refer to the code below:

```
Const resolveAfterMilliseconds = (ms) => Promise.resolve ( setTimeout (( => console.log(ms), ms ));
Const aPromise = await resolveAfterMilliseconds(500); Const bPromise = await resolveAfterMilliseconds(500); Await aPromise, wait bPromise;
What is the result of runningline 05?
```

- A. aPromise and bPromise run sequentially.
- B. Neither aPromise or bPromise runs.
- C. aPromise and bPromise run in parallel.
- D. Only aPromise runs.

**Answer:** B

**NEW QUESTION 120**

Which statement parses successfully?

- A. JSO
- B. `parse ("foo");`
- C. `JSON.parse ("foo");`
- D. `JSON.parse ("foo");`
- E. `JSON.parse ("foo");`

**Answer:** A

**NEW QUESTION 122**

Which statement accurately describes the behaviour of the async/ await keywords ?

- A. The associated class contains some asynchronous functions.
- B. The associated function will always return a promise
- C. The associated function can only be called via asynchronous methods
- D. The associated sometimes returns a promise.

**Answer:** B

**NEW QUESTION 124**

Refer to the code below:

```
01 function changeValue(param) {
02   param = 5;
03 }
04 let a = 10;
05 let b = a;
06
07 changeValue(b);
08 const result = a + ' - ' + b;
```

What is the value of result when the code executes?

- A. 10-10
- B. 5-5
- C. 10-5
- D. 5-10

**Answer:** A

**NEW QUESTION 127**

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 131

Given a value, which three options can a developer use to detect if the value is NaN? Choose 3 answers !

- A. value == NaN
- B. Object.is(value, NaN)
- C. value === Number.NaN
- D. value !== value
- E. Number.isNaN(value)

**Answer:** AE

#### NEW QUESTION 132

Refer to the following array: Let arr = [ 1,2, 3, 4, 5];  
Which three options result in x evaluating as [3, 4, 5] ? Choose 3 answers.

- A. Let x= arr.filter (( a) => (a<2));
- B. Let x=arr.splice(2,3);
- C. Let x= arr.slice(2);
- D. Let x= arr.filter((a) => ( return a>2 ));
- E. Let x = arr.slice(2,3);

**Answer:** BCD

#### NEW QUESTION 136

A developer has code that calculates a restaurant bill, but generates incorrect answers while testing the code:

```
function calculateBill ( items ) { let total = 0;  
total += findSubTotal(items); total += addTax(total);  
total += addTip(total); return total;  
}
```

Which option allows the developer to step into each function execution within calculateBill?

- A. Using the debugger command on line 05.
- B. Using the debugger command on line 03
- C. Calling the console.trace (total) method on line 03.
- D. Wrapping findSubtotal in a console.log() method.

**Answer:** A

#### NEW QUESTION 139

Which two console logs outputs NaN? Choose 2 answers

- A. console.log(10/ Number('5'));
- B. console.log(parseInt('two'));
- C. console.log(10/ "five");
- D. console.log(10/0);

**Answer:** BC

#### NEW QUESTION 140

In which situation should a developer include a try .. catch block around their function call ?

- A. The function has an error that shouldnot be silenced.
- B. The function results in an out of memory issue.
- C. The function might raise a runtime error that needs to be handled.
- D. The function contains scheduled code.

**Answer:** C

#### NEW QUESTION 141

A developer has two ways to write a function: Option A:

```
function Monster() { This.growl = () => { Console.log ("Grr!");  
}  
}
```

Option B:

```
function Monster() {}; Monster.prototype.growl =() => { console.log("Grr!");  
}
```

After deciding on an option, the developer creates 1000 monster objects. How many growl methods are created with Option A Option B?

- A. 1 growl method is created for Option
- B. 1000 growl methods are created for Option B.
- C. 1000 growl method is created for Option
- D. 1 growl methods are created for Option B.
- E. 1000 growl methods are created regardless of which option is used.

F. 1 growl method is created regardless of which option is used.

**Answer: B**

**NEW QUESTION 145**

Given the following code: Counter = 0;  
 const logCounter = () => { console.log(counter);  
 );  
 logCounter(); setTimeout(logCounter, 1100); setInterval(() => {  
 Counter++ logCounter();  
 }, 1000);  
 What is logged by the first four log statements?

- A. 0 0 1 2
- B. 0 1 2 3
- C. 0 1 1 2
- D. 0 1 2 2

**Answer: C**

**NEW QUESTION 148**

Which three statements are true about promises ? Choose 3 answers

- A. The executor of a new Promise runs automatically.
- B. A Promise has a .then() method.
- C. A fulfilled or rejected promise will not change states .
- D. A settled promise can become resolved.
- E. A pending promise can become fulfilled, settled, or rejected.

**Answer: BCE**

**NEW QUESTION 150**

A developer is leading the creation of a new browser application that will serve a single page application. The team wants to use a new web framework Minimalist.js. The Lead developer wants to advocate for a more seasoned web framework that already has a community around it. Which two frameworks should the lead developer advocate for? Choose 2 answers

- A. Vue
- B. Angular
- C. Koa
- D. Express

**Answer: BD**

**NEW QUESTION 153**

Refer to the code below:  
 Async funct on functionUnderTest(isOK) { If (isOK) return 'OK' ;  
 Throw new Error('not OK');  
 )  
 Which assertion accurately tests the above code?

- A. Console.assert (await functionUnderTest(true), ' OK ')
- B. Console.assert (await functionUnderTest(true), ' not OK ')
- C. Console.assert (awaitfunctionUnderTest(true), ' not OK ')
- D. Console.assert (await functionUnderTest(true), 'OK')

**Answer: D**

**NEW QUESTION 157**

Given the JavaScript below:  
 1 function filterDOM (searchString) {  
 2 const parsedSearchString = searchString && searchString.toLowerCase() ;  
 03 document.quesrySelectorAll(' .account' ) . forEach(account => (  
 04 const accountName = account.innerHTML.toLowerCase();  
 05 account. Style.display = accountName.includes(parsedSearchString) ? /\*Insert code\*/;  
 06 ));  
 07 }

Which code should replace the placeholder comment on line 05 to hide accounts that do not match the search string?

- A. ' name ' : ' block '
- B. ' Block ' : ' none '
- C. ' visible ' : ' hidden '
- D. ' hidden ' : ' visible '

**Answer: B**

**NEW QUESTION 158**

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 159

A developer wants to define a function log to be used a few times on a single-file JavaScript script.

```
01 // Line 1 replacement
02 console.log("LOG:", logInput);
03 }
```

Which two options can correctly replace line 01 and declare the function for use? Choose 2 answers

- A. function log(logInput) {
- B. const log(logInput) {
- C. const log = (logInput) => {
- D. function log = (logInput) {

**Answer:** AC

#### NEW QUESTION 161

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 162

.....

## **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](#)**