

DUMPS ARENA

Salesforce Certified JavaScript Developer I

Salesforce JavaScript-Developer-I

Version Demo

Total Demo Questions: 15

Total Premium Questions: 271

Buy Premium PDF

<https://dumpsarena.co>

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

QUESTION NO: 1

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 two.)

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

ANSWER: C D

Explanation:

Reference: <https://www.telerik.com/blogs/what-is-the-point-of-promises>

QUESTION NO: 2

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 ws.

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

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

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

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

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

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

A.

B. C.

D.

Answer: B**Explanation:**

A. C.

D.

ANSWER: A**QUESTION NO: 3**

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: B D**QUESTION NO: 4**

Which two options are core Node.js modules?

Choose 2 answers

- A. worker
- B. istream
- C. exception
- D. http

ANSWER: B D**QUESTION NO: 5**

A developer has an ErrorHandler module that contains multiple functions.

What kind of export be leverages so that multiple functions can be used?

- A. Named
- B. All
- C. Multi
- D. Default

ANSWER: A

QUESTION NO: 6

Refer to the code below:

```
01 function execute() {  
02   return new Promise((resolve, reject) => reject());  
03 }  
04 let promise = execute();  
05  
06 promise  
07   .then(() => console.log('Resolved1'))  
08   .then(() => console.log('Resolved2'))  
09   .then(() => console.log('Resolved3'))  
10   .catch(() => console.log('Rejected'))  
11   .then(() => console.log('Resolved4'));
```

What is the result when the Promise in the execute function is rejected?

- A. Resolved1 Resolved2 Resolved3 Resolved4
- B. Rejected
- C. Rejected Resolved
- D. Rejected1 Rejected2 Rejected3 Rejected Rejected Rejected4

ANSWER: C

QUESTION NO: 7 - (SIMULATION)

Refer the code below.

```
x=3.14;  
function myfunction() {  
"use strict";  
y=x;  
}  
z=x;  
myFunction();
```

ANSWER: See the answer below in explanation:

Explanation:

Z is equal to 3.14

Use strict has effect only on line 5.

Line 5 throws an error

QUESTION NO: 8

A developer implements a function that adds a few values.

```
01 function sum(num) {  
02   if (num === undefined) {  
03     num = 0;  
04   }  
05   return function(num2, num3) {  
06     if (num3 === undefined) {  
07       num3 = 0;  
08     }  
09     return num + num2 + num3;  
10   }  
11 }
```

Which three options can the developer invoke for this function to get a return value of 10? (Choose three.)

- A. sum(5) (5)
- B. sum() (10)
- C. sum(5, 5) ()

- D. sum(10) ()
- E. sum() (5, 5)

ANSWER: A C E

QUESTION NO: 9

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

QUESTION NO: 10

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:

"Hello"

Display

The developer wrote the javascript code below:

```
Const button = document.querySelector('button');  
button.addEventListener('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.getElementById('input');
- C. Replace line 02 with button.addEventListener("onclick", function() {
- D. Replace line 02 with button.addCallback("click", function() {

ANSWER: A

QUESTION NO: 11

A developer writes the code below to return a message to a user attempting to register a new username. If the username is available, a variable named msg is declared and assigned a value on line 03.

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

What is the value of msg when getAvailableabilityMessage ("newUserName") is executed and get Availability ("newUserName") returns true?

- A. "msg is not defined"
- B. "newUserName"
- C. "User-name available"
- D. undefined

ANSWER: C**QUESTION NO: 12**

A developer creates an object where its properties should be immutable and prevent properties from being added or modified.

Which method should be used to execute this business requirement ?

- A. Object.const()
- B. Object.eval()
- C. Object.lock()
- D. Object.freeze()

ANSWER: D**QUESTION NO: 13**

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: A C**QUESTION NO: 14**

Refer to the string below:

```
const str = 'Salesforce';
```

Which two statements result in the word 'Sales'?

Choose 2 answers

- A. `str.substr(1, 5);`
- B. `str.substr (0, 5);`
- C. `str.substring (1, 5);`
- D. `str.substring (0, 5);`

ANSWER: B D

QUESTION NO: 15

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

Which two promises are rejected? (Choose two.)

- A. `new Promise(() => {throw 'Cool error here'}).then((null, error => console.error(error)));`
- B. `new Promise((resolve, reject) => {throw 'Cool error here'}).catch(error => console.error(error));`
- C. `Promise.reject('Cool error here').then(error => console.error(error));`
- D. `Promise.reject('Cool error here').catch(error => console.error (error));`

ANSWER: B D

Explanation:

Reference: <https://www.javascripttutorial.net/es6/promise-error-handling/>