

Salesforce

CRT-600 Exam

Javascript Developer I

Questions & Answers (Demo Version - Limited Content)

Thank you for Downloading CRT-600 exam PDF Demo

Get Full File:

<https://www.dumpsfire.com/crt-600-dumps/>

Version: 5.0

Question: 1

Refer to the code below:

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

Answer: B, D

Question: 2

A developer needs to test this function:

```
01 const sum3 = (arr) => (
02   if (!arr.length) return 0,
03   if (arr.length === 1) return arr[0],
04   if (arr.length === 2) return arr[0] + arr[1],
05   return arr[0] + arr[1] + arr[2],
06 );
```

Which two assert statements are valid tests for the function?

Choose 2 answers

- A. console.assert(sum3(1, '2')) == 12);
- B. console.assert(sum3(0)) == 0);

- C. `console.assert(sum3(-3, 2)) == -1;`
- D. `console.assert(sum3('hello', 2, 3, 4)) === NaN;`

Answer: A, C

Question: 3

Which statement phrases successfully?

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

Answer: D

Question: 4

Refer to the code below:

```
01 let car1 = new promise((_, reject) =>
02 setTimeout(reject, 2000, "Car 1 crashed in"));
03 let car2 = new Promise(resolve => setTimeout(resolve, 1500, "Car 2
completed"));
04 let car3 = new Promise(resolve => setTimeout (resolve, 3000, "Car 3
Completed"));
05 Promise.race([car1, car2, car3])
06 .then(value => (
07 let result = $(value) the race. `;
08 ))
09 .catch( arr => (
10 console.log("Race is cancelled.", err);
11 ));
```

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

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

Answer: C

Question: 5

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

Question: 6

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

Question: 7

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: B, C, E

Question: 8

Given the code below:

```
01 function GameConsole (name) {
02 this.name = name;
03 }
04
05 GameConsole.prototype.load = function(gamename) {
06 console.log( `${this.name} is loading a game : ${gamename} ...` );
07 )
08 function Console 16 Bit (name) {
09 GameConsole.call(this, name) ;
10 }
11 Console16bit.prototype = Object.create ( GameConsole.prototype) ;
12 //insert code here
13 console.log( `${this.name} is loading a cartridge game : ${gamename} ...` );
14 }
15 const console16bit = new Console16bit(' SNEGeneziz ');
16 console16bit.load(' Super Nonic 3x Force ');
```

What should a developer insert at line 15 to output the following message using the method ?

> SNEGeneziz is loading a cartridge game: Super Monic 3x Force . . .

- A. Console16bit.prototype.load(gamename) = function() {
- B. Console16bit.prototype.load = function(gamename) {
- C. Console16bit= Object.create(GameConsole.prototype).load = function (gamename) {
- D. Console16bit.prototype.load(gamename) {

Answer: B

Question: 9

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

Question: 10

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: C, D

Thank You for trying CRT-600 PDF Demo

<https://www.dumpsfire.com/crt-600-dumps/>

Start Your CRT-600 Preparation

[Limited Time Offer] Use Coupon " **Fire20** " for extra 20% discount on the purchase of PDF file. Test your CRT-600 preparation with actual exam questions