

# Latest Version: 10.1

## 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 data

**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**