

# Certified JS-Dev-101 Questions, New JS-Dev-101 Test Tutorial



What's more, part of that TestBraindump JS-Dev-101 dumps now are free: [https://drive.google.com/open?id=1QRi\\_9yMVc6wAeTmiE7cEpHb\\_sJBGTVhY](https://drive.google.com/open?id=1QRi_9yMVc6wAeTmiE7cEpHb_sJBGTVhY)

Because Salesforce JS-Dev-101 exam is concerning the future and the destiny of IT people, they pay more attention to the certification. When you decide to choosing IT industry, you have proved your ability. However, what we learn is not enough at all. Salesforce JS-Dev-101 Certification will be a big challenge for the candidates. If you decide to join our TestBraindump, we guarantee your success in the first attempt. If you fail, FULL REFUND!

Our experts are not slavish followers who just cut and paste the content into our JS-Dev-101 practice materials, all JS-Dev-101 exam questions are elaborately compiled by them. Just a small amount of money, but you can harvest colossal success with potential bright future. So we have the courage and justification to declare the number one position in this area, and choosing JS-Dev-101 Actual Exam is choosing success.

>> **Certified JS-Dev-101 Questions** <<

## New JS-Dev-101 Test Tutorial - JS-Dev-101 Test Dump

Our JS-Dev-101 study materials can improves your confidence for real JS-Dev-101 exam and will help you remember the exam questions and answers that you will take part in. You can choose the version which suits you mostly. Our JS-Dev-101 exam torrents simplify the important information and seize the focus to make you master the JS-Dev-101 Test Torrent in a short time. To gain a comprehensive understanding of our JS-Dev-101 study materials, you have to look at the introduction of our product firstly if you free download the demo of our JS-Dev-101 exam questions.

## Salesforce Certified JavaScript Developer - Multiple Choice Sample Questions (Q113-Q118):

**NEW QUESTION # 113**

Refer to code below:

```
Let productSKU = '8675309' ;
```

A developer has a requirement to generate SKU numbers that are always 19 characters long, starting with 'sku', and padded with zeros.

Which statement assigns the values sku0000000008675309 ?

- A. `productSKU = productSKU .padEnd (16. '0').padstart('sku');`
- B. `productSKU = productSKU .padEnd (16. '0').padstart(19, 'sku');`
- C. `productSKU = productSKU .padStart (16. '0').padstart(19, 'sku');`
- D. `productSKU = productSKU .padStart (19. '0').padstart('sku');`

**Answer: C**

#### NEW QUESTION # 114

A developer wants to use a module called DatePrettyPrint. This module exports one default function called printDate(). How can the developer import and use printDate()?

- A. `import printDate from '/path/DatePrettyPrint.js';`  
`DatePrettyPrint.printDate();`
- B. `import printDate from '/path/DatePrettyPrint.js';`  
`printDate();`
- C. `import DatePrettyPrint from '/path/DatePrettyPrint.js';`  
`DatePrettyPrint.printDate();`
- D. `import DatePrettyPrint() from '/path/DatePrettyPrint.js';`  
`printDate();`

**Answer: B**

Explanation:

---

Comprehensive and Detailed Explanation From Exact Extract JavaScript Knowledge ES Modules follow these rules:

Default exports are imported using:

```
import anyName from 'module'
```

The name chosen in the import does NOT need to match the exported name.

If a module exports:

```
export default function printDate() {}
```

then consuming code should import the function directly:

```
import printDate from '/path/DatePrettyPrint.js';
```

To execute it:

```
printDate();
```

Option analysis:

A incorrect:

`import DatePrettyPrint()` is invalid syntax.

Also calling `printDate()` without importing it is incorrect.

B incorrect:

A default export imported as `DatePrettyPrint` gives the function itself, not an object containing methods.

You cannot call `DatePrettyPrint.printDate()`.

C incorrect:

Imports `printDate` but then calls `DatePrettyPrint.printDate()`, which does not exist.

D correct:

Correct import of a default-exported function.

Correct direct invocation.

---

JavaScript Knowledge Reference (text-only)

Default export → imported without braces.

Importing default functions gives a callable function, not an object.

Correct syntax: `import identifier from "module"`.

#### NEW QUESTION # 115

Refer to the code snippet below:

```
Let array = [1, 2, 3, 4, 4, 5, 4, 4];
For (let i=0; i < array.length; i++){
  if (array[i] === 4) {
    array.splice(i, 1);
  }
}
```

What is the value of the array after the code executes?

- A. [1, 2, 3, 4, 5, 4, 4]
- B. [1, 2, 3, 4, 4, 5, 4]
- C. [1, 2, 3, 4, 5, 4]
- D. [1, 2, 3, 5]

**Answer: C**

### NEW QUESTION # 116

In the browser, the window object is often used to assign variables that require the broadest scope in an application. Node.js application does not have access to the window object by default.

Which two methods are used to address this ?

Choose 2 answers

- A. Assign variables to the global object.
- B. Use the document object instead of the window object.
- C. Create a new window object in the root file.
- D. Assign variables to module.exports and require them as needed.

**Answer: A**

### NEW QUESTION # 117

```
static delay = async delay => {
  return new Promise(resolve => {
    setTimeout(resolve, delay);
  });
};
static asyncCall = async () => {
  await delay(1000);
  console.log(1);
};
console.log(2);
asyncCall();
console.log(3);
```

(Assume delay and asyncCall are in scope as functions.)

What is logged to the console?

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

**Answer: A**

Explanation:

console.log(2); → logs 2.

asyncCall(); starts, but await delay(1000) pauses it until the promise resolves. No log yet from inside.

console.log(3); runs immediately → logs 3.

After ~1000ms, delay resolves, asyncCall resumes → logs 1.

Order: 2, 3, 1 → D.



myportal.utt.edu.tt, myportal.utt.edu.tt, geilebookmarks.com, aronzviv925214.theideasblog.com, myportal.utt.edu.tt,  
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt,  
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes

2026 Latest TestBraindump JS-Dev-101 PDF Dumps and JS-Dev-101 Exam Engine Free Share: [https://drive.google.com/open?id=1QRi\\_9yMVc6wAeTmiE7cEpHb\\_sJBGTvY](https://drive.google.com/open?id=1QRi_9yMVc6wAeTmiE7cEpHb_sJBGTvY)