

Sie können so einfach wie möglich - JS-Dev-101 bestehen!



P.S. Kostenlose 2026 Salesforce JS-Dev-101 Prüfungsfragen sind auf Google Drive freigegeben von Pass4Test verfügbar:
<https://drive.google.com/open?id=1Q8T8pV8qLKU1e-I8b4Zl8pbz7mYxoX-v>

Heute, wo das Internet schnell entwickelt, ist es ein übliches Phänomen, Online-Ausbildung zu wählen. Pass4Test ist eine der vielen Online-Ausbildungswebsites. Pass4Test hat langjährige Erfahrungen und kann den Kandidaten die Lernmaterialien von guter Qualität zur Salesforce JS-Dev-101 Zertifizierungsprüfung bieten, um ihre Bedürfnisse abzudecken.

Die Produkte von PassTest sind für diejenigen, die sich an der Salesforce JS-Dev-101 Zertifizierungsprüfung beteiligen, geeignet. Die Schulungsmaterialien von Pass4Test enthalten nicht nur Trainingsmaterialien zur Salesforce JS-Dev-101 Zertifizierungsprüfung, um Ihre Fachkenntnisse zu konsolidieren, sondern auch die genauen Prüfungsfragen und Antworten. Wir versprechen, dass Sie die Salesforce JS-Dev-101 Zertifizierungsprüfung beim ersten Versuch mit einer hohen Note bestehen können.

>> JS-Dev-101 Examengine <<

JS-Dev-101 Schulungsangebot & JS-Dev-101 Fragenpool

In Bezug auf die Salesforce JS-Dev-101 Zertifizierungsprüfung ist die Zuverlässigkeit nicht zu ignorieren. Die Schulungsmaterialien zur JS-Dev-101 Zertifizierungsprüfung von Pass4Test werden besonders entworfen, um Ihre Effizienz zu erhöhen. Unsere Website hat weltweit die höchste Erfolgsquote.

Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 Prüfungsfragen mit Lösungen (Q58-Q63):

58. Frage

```
01 class Student {
02 constructor(name) {
03 this._name = name;
04 }
05 displayGrade() {
06 console.log(`${this._name} got 70% on test.`);
07 }
08 }
09
10 class GraduateStudent extends Student {
11 constructor(name) {
12 super(name);
13 this._name = "Graduate Student " + name;
14 }
15 displayGrade() {
16 console.log(`${this._name} got 100% on test.`);
17 }
18 }
```

```
19 let student = new GraduateStudent("Jane");
20 student.displayGrade();
...

```

What is the console output?:

- A. Uncaught ReferenceError
- **B. Better student Jackie got 100% on test.**
- C. Better student Jackie got 70% on test.
- D. Jackie got 70% on test.

Antwort: B

59. Frage

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.function.fullName()`
- B. `cat.get.fullName`
- **C. `cat.fullName`**
- D. `cat.fullName()`

Antwort: C

Begründung:

`fullName` is defined as a getter:

```
get fullName() { ... }
```

Getters are accessed like properties, not like functions.

So:

```
cat.fullName; // "Fancy Whiskers"
```

No parentheses.

Why others are wrong:

A: `cat.fullName()` tries to call the returned string as a function.

B and C: These properties (`get`, `function`) do not exist; they are misinterpretations of syntax.

60. Frage

A developer has an `isDeg` function that takes one argument, `pts`. They want to schedule the function to run every minute.

What is the correct syntax for scheduling this function?

- A. `setInterval(isDeg('cat'), 60000);`
- **B. `setInterval(isDeg, 60000, 'cat');`**
- C. `setTimeout(isDeg('cat'), 60000);`
- D. `setTimeout(isDeg, 60000, 'cat');`

Antwort: B

Begründung:

`setInterval(fn, delay, arg1, arg2, ...)` calls `fn` every `delay` ms, passing extra arguments.

To call `isDeg` every 60 seconds with `'cat'`:

```
setInterval(isDeg, 60000, 'cat');
```

Why others are wrong:

A/D: `isDeg('cat')` is called immediately, and its return value is scheduled, not the function.

C: `setTimeout` runs once after 60 seconds, not repeatedly.

61. Frage

Refer to the code below:

```
01 function myFunction(reassign) {
02 let x = 1;
03 var y = 1;
04
05 if(reassign) {
06 let x = 2;
07 var y = 2;
08 console.log(x);
09 console.log(y);
10 }
11
12 console.log(x);
13 console.log(y);
14 }
```

What is displayed when myFunction(true) is called?

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

Antwort: B

Begründung:

Comprehensive and Detailed Explanation From Exact JavaScript knowledge:

This question tests understanding of let (block scope) and var (function scope) in JavaScript.

Initial declarations in the function:

```
let x = 1; // line 2
```

```
var y = 1; // line 3
```

Here:

x is declared with let, so it is block-scoped to the function body.

y is declared with var, so it is function-scoped to the entire function.

Inside the if (reassign) block (and since reassign is true, we enter it):

```
if (reassign) {
```

```
let x = 2; // line 6
```

```
var y = 2; // line 7
```

```
console.log(x); // line 8
```

```
console.log(y); // line 9
```

```
}
```

Detailed behavior:

let x = 2; on line 6 creates a new block-scoped variable x that exists only inside the if block. It does not change the outer x declared on line 2.

var y = 2; on line 7 declares y with var again, but var is function-scoped. This effectively reassigns the same y defined on line 3 for the entire function. After this line, y is 2 everywhere in the function.

Now, inside the if block:

console.log(x); (line 8) logs the inner block-scoped x, which is 2.

console.log(y); (line 9) logs y, which is the function-scoped y that was set to 2.

So the first two outputs are:

2

2

After the if block, execution continues:

```
console.log(x); // line 12
```

```
console.log(y); // line 13
```

Outside the if block:

The block-scoped let x = 2; no longer exists; it was only visible inside the if block.

The outer let x = 1; (line 2) is still in scope and has not been changed.

Thus:

console.log(x); (line 12) logs the outer x, which is still 1.

console.log(y); (line 13) logs y which, due to var y = 2; inside the if, is now 2 for the whole function.

Therefore, when myFunction(true) is called, the output in order is:

2 (inner x in if)

2 (function-scoped y after reassignment)

1 (outer x after if)

2 (function-scoped y remains 2)

This corresponds to:

JavaScript knowledge / study guide reference concepts:

let declarations and block scope

var declarations and function scope

Shadowing of variables with let inside a block

Re-declaration and reassignment of var within a function

Execution order of statements and console output

62. Frage

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

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

Antwort: B

63. Frage

.....

Wenn wir am Anfang die Fragenkataloge zur Salesforce JS-Dev-101 Zertifizierungsprüfung bieten, haben wir niemals geträumt, dass wir so einen guten Ruf bekommen können. Wir geben Ihnen die unglaubliche Garantie. Wenn Sie die Produkte von Pass4Test für Ihre Salesforce JS-Dev-101 Zertifizierungsprüfung benutzen, versprechen wir Ihnen, die Prüfung 100% zu bestehen.

JS-Dev-101 Schulungsangebot: <https://www.pass4test.de/JS-Dev-101.html>

Mit den Lernmaterialien von Pass4Test JS-Dev-101 Schulungsangebot können Sie 100% die Prüfung bestehen, Salesforce JS-Dev-101 Examengine Mit ihr können Sie alle Probleme lösen, Salesforce JS-Dev-101 Examengine In Wirklichkeit sind zahlreiche Prüflinge in der Prüfung durchgefallen, Mit einem Salesforce JS-Dev-101 Zertifikat kann der Berufstätige in der IT-Branche eine bessere Berufsmöglichkeit haben, Viele berühmte IT-Firmen stellen ihre Angestellte laut dem Salesforce JS-Dev-101 Zertifikat ein.

Ich werde mit den Mädchen zurückkehren oder gar nicht, Und ich kenne JS-Dev-101 eine Frau, welche sich denselben Namen gegeben hat, Mit den Lernmaterialien von Pass4Test können Sie 100% die Prüfung bestehen.

Seit Neuem aktualisierte JS-Dev-101 Examfragen für Salesforce JS-Dev-101 Prüfung

Mit ihr können Sie alle Probleme lösen, In Wirklichkeit sind zahlreiche Prüflinge in der Prüfung durchgefallen, Mit einem Salesforce JS-Dev-101 Zertifikat kann der Berufstätige in der IT-Branche eine bessere Berufsmöglichkeit haben.

Viele berühmte IT-Firmen stellen ihre Angestellte laut dem Salesforce JS-Dev-101 Zertifikat ein.

- JS-Dev-101 Zertifikatsdemo JS-Dev-101 Testking JS-Dev-101 Kostenlos Downloaden Öffnen Sie die Webseite ▶ www.echtfage.top ◀ und suchen Sie nach kostenloser Download von JS-Dev-101 JS-Dev-101 Prüfungen
- JS-Dev-101 Trainingsmaterialien: Salesforce Certified JavaScript Developer - Multiple Choice - JS-Dev-101 Lernmittel - Salesforce JS-Dev-101 Quiz Suchen Sie auf ▶ www.itzert.com nach ▶ JS-Dev-101 und erhalten Sie den kostenlosen Download mühelos JS-Dev-101 Online Test
- JS-Dev-101 Online Prüfungen JS-Dev-101 Online Prüfungen JS-Dev-101 Testing Engine Suchen Sie auf

