

Authorized JS-Dev-101 Test Dumps & JS-Dev-101 New Real Exam



What's more, part of that Actual4dump JS-Dev-101 dumps now are free: https://drive.google.com/open?id=1SKYA04_3uLvPymt5kSxp18vDEaK2IGex

Actual4dump's exam dumps guarantee your success with a promise of returning back the amount you paid. Such an in itself is the best proof of the unique quality of our product and its ultimate utility for you. Try JS-Dev-101 Dumps and ace your upcoming JS-Dev-101 certification test, securing the best percentage of your academic career. If you didn't pass JS-Dev-101 exam, we guarantee you will get full refund.

They have years of experience in Actual4dump JS-Dev-101 exam preparation and success. So you can trust Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 dumps and start Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 exam preparation right now. The Actual4dump is quite confident that the Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 valid dumps will not ace your Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 Exam Preparation but also enable you to pass this challenging Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 exam with flying colors. The Actual4dump is one of the top-rated and leading Salesforce Certified JavaScript Developer - Multiple Choice JS-Dev-101 test questions providers.

>> **Authorized JS-Dev-101 Test Dumps** <<

JS-Dev-101 New Real Exam & JS-Dev-101 Latest Exam Testking

It is understandable that different people have different preference in terms of JS-Dev-101 study guide. Taking this into consideration, and in order to cater to the different requirements of people from different countries in the international market, we have prepared three kinds of versions of our JS-Dev-101 Preparation questions in this website, namely, PDF version, APP online and software version, and you can choose any one of them as you like. You will our JS-Dev-101 exam dumps are the best!

Salesforce JS-Dev-101 Exam Syllabus Topics:

Topic	Details
-------	---------

Topic 1	<ul style="list-style-type: none"> • Server Side JavaScript: Covers Node.js implementations, CLI commands, core modules, and package management solutions for given scenarios.
Topic 2	<ul style="list-style-type: none"> • Variables, Types, and Collections: Covers declaring and initializing variables, working with strings, numbers, dates, arrays, and JSON, along with understanding type coercion and truthy • falsy evaluations.
Topic 3	<ul style="list-style-type: none"> • Objects, Functions, and Classes: Covers function, object, and class implementations to meet business requirements, along with the use of modules, decorators, variable scope, and execution flow.
Topic 4	<ul style="list-style-type: none"> • Debugging and Error Handling: Covers proper error handling techniques and the use of the console and breakpoints to debug code.

Salesforce Certified JavaScript Developer - Multiple Choice Sample Questions (Q38-Q43):

NEW QUESTION # 38

A developer at Universal Containers creates a new landing page based on HTML, CSS, and JavaScript TO ensure that visitors have a good experience, a script named `personalizeContext` needs to be executed when the webpage is fully loaded (HTML content and all related files), in order to do some custom initialization.

Which statement should be used to call `personalizeWebsiteContent` based on the above business requirement?

- A. `document.addEventListener("DOMContentLoaded", personalizeWebsiteContext);`
- B. `window.addEventListener("load",personalizeWebsiteContext);`
- C. `window.addEventListener("onload", personalizeWebsiteContext);`
- D. `document.addEventListener("onDOMContentLoaded", personalizeWebsiteContext);`

Answer: B

NEW QUESTION # 39

Refer to the code below:

```
const searchText = 'Yay! Salesforce is amazing!';
let result1 = searchText.search(/sales/i);
let result2 = searchText.search(/sales/);
console.log(result1);
console.log(result2);
```

After running this code, which result is displayed on the console?

- A. 5
-1
- B. 5
0
- C. 5
undefined
- D. true
false

Answer: A

Explanation:

Comprehensive and Detailed Explanation From Exact Extract JavaScript Knowledge:

String: "Yay! Salesforce is amazing!"

Index positions:

'Y' at 0, 'a' at 1, 'y' at 2, '!' at 3, space at 4, 'S' at 5, 'a' at 6, 'l' at 7, 'e' at 8, 's' at 9, etc.

Substring "Sales" starts at index 5.

String.prototype.search with a regex returns the index of the first match or -1 if there is no match.

`searchText.search(/sales/i);`

/sales/i is case-insensitive because of the i flag.
It matches "Sales" beginning at index 5.
So result1 is 5.
searchText.search(/sales/);
/sales/ is case-sensitive.
It requires lowercase "sales".
The text has "Sales" with uppercase S, so this does not match.
search returns -1 when there is no match.
So result2 is -1.
Console output:
First log: 5
Second log: -1
Option D matches this.
Concepts: regex search, case sensitivity vs i flag, String.prototype.search return values.

NEW QUESTION # 40

A developer creates a simple webpage with an input field. When a user enters text and clicks the button, the actual value must be displayed in the console:

HTML:

```
<input type="text" value="Hello" name="input">
```

```
<button type="button">Display</button>
```

JavaScript:

```
01 const button = document.querySelector('button');
```

```
02 button.addEventListener('click', () => {
```

```
03 const input = document.querySelector('input');
```

```
04 console.log(input.getAttribute('value'));
```

```
05 });
```

When the user clicks the button, the output is always "Hello".

What needs to be done to make this code work as expected?

- A. Replace line 02 with `button.addCallback("click", function() {`
- B. Replace line 03 with `const input = document.getElementByIdName("input");`
- C. Replace line 02 with `button.addEventListener("onclick", function() {`
- **D. Replace line 04 with `console.log(input.value);`**

Answer: D

Explanation:

Comprehensive and Detailed Explanation From Exact Extract JavaScript Knowledge `getAttribute('value')` This returns the initial HTML attribute, not the live, updated value.

Even if the user edits the text, the value attribute remains "Hello" because HTML attributes do not update dynamically.

Input elements have a property value that always reflects the live current text inside the field.

So:

`input.value`

returns the user-entered value.

Therefore, line 04 must use the property, not the attribute:

```
console.log(input.value);
```

This ensures the updated input value is displayed.

JavaScript knowledge references (text-only)

HTML attributes are static and retrieved using `getAttribute()`.

DOM element properties (like `value`) represent the current live state.

Input value changes update the `.value` property, not the attribute.

NEW QUESTION # 41

A developer wants to create a simple image upload using the File API.

HTML:

```
<input type="file" onchange="previewFile()">
<img src="" height="200" alt="Image preview..." />
```

JavaScript:

```
01 function previewFile() {
02   const preview = document.querySelector('img');
03   const file = document.querySelector('input[type=file]').files[0];
04   // line 4 code
05   reader.addEventListener("load", () => {
06     preview.src = reader.result;
07   }, false);
08   // line 8 code
09 }
```

Which code in lines 04 and 08 allows the selected local image to be displayed?

- A. 04 `const reader = new File();`
08 `if (file) reader.readAsDataURL(file);`
- B. 04 `const reader = new FileReader();`
08 `if (file) URL.createObjectURL(file);`
- C. 04 `const reader = new FileReader();`
08 `if (file) reader.readAsDataURL(file);`

Answer: C

Explanation:

Comprehensive and Detailed Explanation From Exact Extract JavaScript Knowledge The File API in browsers provides the FileReader object to read file contents selected from `<input type="file">`.

Important knowledge points:

`new FileReader()` creates a file-reading object.

`.readAsDataURL(file)` reads a file and produces a Base64 URL string.

The "load" event fires when the file has finished reading.

`reader.result` contains the data URL after reading completes.

Therefore, the correct implementation must:

Create a FileReader instance:

```
const reader = new FileReader();
```

Call:

```
reader.readAsDataURL(file);
```

Use the load event handler to assign the image preview:

```
preview.src = reader.result;
```

Option B is the only option that matches valid JavaScript File API usage.

Option A is incorrect because File is not a constructor for reading files.

Option C is incorrect because `URL.createObjectURL(file)` must be assigned directly as a URL, not used with `reader.result`.

JavaScript Knowledge Reference (text-only)

The file-reading interface in browsers is FileReader.

`readAsDataURL()` loads files as Base64 data URLs.

The load event indicates when the reader has finished and `reader.result` is available.

NEW QUESTION # 42

Refer to the code:

```
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 `runParallel()`?

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

Answer: A,B

Explanation:

Comprehensive and Detailed Explanation From Exact Extract JavaScript Knowledge Facts about JavaScript Promises:

`runParallel()` is declared `async`, which means it always returns a Promise.

Promises are consumed using `.then()`, `.catch()`, and `.finally()`.

There is no `.done()` method in native JavaScript Promises.

The `async` keyword cannot be placed before a function call (option A is invalid syntax).

Analysis of each option:

A. `async runParallel().then(data);`

Invalid syntax. `async` cannot prefix an expression.

B. Valid. Calls the function and attaches a `.then()` handler.

C. Invalid. `.done()` is not part of JavaScript Promise API.

D. Valid. Calls `runParallel()` and chains `.then()`.

Therefore the correct answers are B and D.

JavaScript Knowledge Reference (text-only)

`async` functions return Promises.

Promises use `.then()` to retrieve resolved values.

`.done()` is not part of the standard Promise interface.

NEW QUESTION # 43

.....

Our professions endeavor to provide you with the newest information with dedication on a daily basis to ensure that you can catch up with the slight changes of the JS-Dev-101 test. Therefore, our customers are able to enjoy the high-productive and high-efficient users' experience. In this circumstance, as long as your propose and demand are rational, we have the duty to guarantee that you can enjoy the one-year updating system for free. After purchasing our JS-Dev-101 Test Prep, you have the right to enjoy the free updates for one year long after you buy our JS-Dev-101 exam questions.

JS-Dev-101 New Real Exam: <https://www.actual4dump.com/Salesforce/JS-Dev-101-actualtests-dumps.html>

- Salesforce Certified JavaScript Developer - Multiple Choice latest study torrent - JS-Dev-101 actual prep exam Immediately open ➡ www.verifeddumps.com and search for ▶ JS-Dev-101 ◀ to obtain a free download JS-Dev-101 Free Sample Questions
- Customizable JS-Dev-101 Exam Mode JS-Dev-101 Free Test Questions JS-Dev-101 Reliable Test Sims Easily obtain (JS-Dev-101) for free download through 「 www.pdfvce.com 」 Exam JS-Dev-101 Papers
- Salesforce Certified JavaScript Developer - Multiple Choice latest study torrent - JS-Dev-101 actual prep exam 🕒 Search for ➡ JS-Dev-101 on 🌟 www.prep4away.com 🌟 immediately to obtain a free download Reliable JS-Dev-101 Exam Blueprint
- Regularly updated as per the updates by the Salesforce JS-Dev-101 Search for “JS-Dev-101 ” and obtain a free download on www.pdfvce.com JS-Dev-101 Valid Test Sample
- JS-Dev-101 Accurate Prep Material JS-Dev-101 Free Sample Questions JS-Dev-101 New Study Notes Search for JS-Dev-101 and obtain a free download on ➡ www.vce4dumps.com Reliable JS-Dev-101 Exam Blueprint
- Pdfvce Offers JS-Dev-101 PDF Dumps With Refund Policy Search for 🌟 JS-Dev-101 🌟 and download it for free on ➡ www.pdfvce.com website JS-Dev-101 Online Version
- Salesforce Certified JavaScript Developer - Multiple Choice latest study torrent - JS-Dev-101 actual prep exam Open “

www.troytecdumps.com” and search for ▶ JS-Dev-101 ◀ to download exam materials for free □ Study Guide JS-Dev-101 Pdf

- Latest JS-Dev-101 Dumps Ebook □ Valid JS-Dev-101 Test Discount □ Trustworthy JS-Dev-101 Exam Torrent □ Search for ☀ JS-Dev-101 □ ☀ □ and easily obtain a free download on ▷ www.pdfvce.com ◁ □ Valid JS-Dev-101 Test Discount
- JS-Dev-101 Online Version □ Reliable JS-Dev-101 Test Forum □ JS-Dev-101 Sample Exam □ Download “ JS-Dev-101 ” for free by simply searching on 「 www.dumpsmaterials.com 」 □ Latest JS-Dev-101 Dumps Ebook
- Pass-Sure Authorized JS-Dev-101 Test Dumps Covers the Entire Syllabus of JS-Dev-101 □ Search on □ www.pdfvce.com □ for 【 JS-Dev-101 】 to obtain exam materials for free download □ JS-Dev-101 Free Sample Questions
- Complete Authorized JS-Dev-101 Test Dumps - Guaranteed Salesforce JS-Dev-101 Exam Success with High-quality JS-Dev-101 New Real Exam □ Search for □ JS-Dev-101 □ and easily obtain a free download on □ www.pdfdumps.com □ □ Reliable JS-Dev-101 Test Pass4sure
- bookmarkinglive.com, ihannakqsh508204.blog-kids.com, cecilymjqc844308.wikiannouncement.com, bookmarkuse.com, sidneybycq532522.ourcodeblog.com, zakarianzmf041759.snack-blog.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, kamailioasterisk.com, socialfactories.com, francesgerp622815.theideasblog.com, Disposable vapes

P.S. Free & New JS-Dev-101 dumps are available on Google Drive shared by Actual4dump: https://drive.google.com/open?id=1SKYA04_3uLvPymt5kSxp18vDEaK2IGex