

New AP-202 Braindumps Ebook, AP-202 Customized Lab Simulation



Our AP-202 exam training’ developers to stand in the perspective of candidate, fully consider their material basis and actual levels of knowledge, formulated a series of scientific and reasonable learning mode, meet the conditions for each user to tailor their learning materials. What’s more, our AP-202 Guide questions are cheap and cheap, and we buy more and deliver more. The more customers we buy, the bigger the discount will be. In order to make the user a better experience to the superiority of our AP-202 actual exam guide, we also provide considerate service,

In addition to the comprehensive Salesforce AP-202 practice exams, our product also includes B2B Commerce for Developers Accredited Professional (AP-202) PDF questions developed by our team to help you get prepared in a short time. Our Prepare for your B2B Commerce for Developers Accredited Professional (AP-202) PDF format works on all smart devices without limits of time and place.

>> New AP-202 Braindumps Ebook <<

Pass Guaranteed Quiz 2026 High Pass-Rate AP-202: New B2B Commerce for Developers Accredited Professional Braindumps Ebook

Our company abides by the industry norm all the time. By virtue of the help from professional experts, who are conversant with the regular exam questions of our latest AP-202 exam torrent we are dependable just like our AP-202 test prep. They can satisfy your

knowledge-thirsty minds. And our AP-202 Quiz torrent is quality guaranteed. By devoting ourselves to providing high-quality practice materials to our customers all these years we can guarantee all content is of the essential part to practice and remember.

Salesforce B2B Commerce for Developers Accredited Professional Sample Questions (Q137-Q142):

NEW QUESTION # 137

What are two purposes of the Shadow DOM in a Lightning web component?

- A. It encapsulates the internal document object model (DOM) structure of a web component
- B. It allows direct access to the document object model of the component
- C. It allows older JavaScript libraries to manipulate the tagging structure
- D. It allow components to be shared while protecting them from being manipulated by arbitrary code

Answer: A,D

Explanation:

According to the Lightning Web Components Developer Guide, Shadow DOM is a standard that encapsulates the internal document object model (DOM) structure of a web component. Encapsulating the DOM gives developers the ability to share a component and protect the component from being manipulated by arbitrary HTML, CSS, and JavaScript. Shadow DOM also provides style and behavior isolation for a web component, which means that the styles and scripts defined inside a component do not affect the rest of the page, and vice versa. Therefore, the purposes of the Shadow DOM in a Lightning web component are to encapsulate the internal DOM structure of a web component (A) and to allow components to be shared while protecting them from being manipulated by arbitrary code (B). The other options are incorrect because Shadow DOM does not allow direct access to the document object model of the component, nor does it allow older JavaScript libraries to manipulate the tagging structure (D). Lightning Web Components Developer Guide, Understand the Shadow DOM Unit

NEW QUESTION # 138

A developer has created a custom Lightning web component for the Cart page that needs to react to changes to cart items from the standard cart component.

How should the developer implement the custom component so changes to cart items and quantities are reflected?

- A. Listen for events on the lightning_commerce_cartChanged channel with the Lightning Event
- B. Add a listener for the cartItemUpdate Lightning event.
- C. Add an event listener for the cartchanged DOM (Document Object Model) event.
- D. Subscribe to events on the lightning_commerce_cartChanged channel using the Lightning

Answer: D

Explanation:

The developer should implement the custom component so that it subscribes to events on the lightning_commerce_cartChanged channel using the Lightning Message Service. This is the recommended way to communicate between custom and standard components on the Cart page, as it allows the custom component to receive updates from the standard cart component whenever the cart items or quantities change. The Lightning Message Service is a pub/sub mechanism that enables components to communicate across the DOM, regardless of their namespace, technology, or source. The lightning_commerce_cartChanged channel is a predefined channel that carries information about the cart state, such as the cart ID, the cart items, and the cart total. Custom Component APIs, Lightning Message Service, Communicate Across the DOM with Lightning Message Service Learn more

1developer.salesforce.com2developer.salesforce.com

NEW QUESTION # 139

A developer is working in Visual Studio Code on a previously deployed project which is rather large and deployments are time consuming. The developer wants to know if a CSS file containing small changes was actually deployed to the org. What is one way this can be accomplished?

- A. Right-click the folder for the component and choose Diff Styles Against Org
- B. Right-click the folder for the component and choose Diff Files Against Org
- C. Click the Tools menu and select Diff Styles Against Org...

- **D. Right-click the CSS file and choose Diff File Against Org**

Answer: D

Explanation:

To know if a CSS file containing small changes was actually deployed to the org, one way that a developer can accomplish this is by right-clicking the CSS file and choosing Diff File Against Org. Diff File Against Org is an option that allows the developer to compare a local file with its remote version in the org using Salesforce CLI commands. The developer can use Visual Studio Code to execute these commands by right-clicking on files or folders in the project and choosing from various diff options. Right-clicking the CSS file and choosing Diff File Against Org allows the developer to see the differences between the local CSS file and the remote CSS file in the org side by side in Visual Studio Code. This way, the developer can verify if their changes were deployed successfully or not. Clicking the Tools menu and selecting Diff Styles Against Org... is not a valid way to know if a CSS file was deployed to the org, as there is no such option in Visual Studio Code or Salesforce CLI. Right-clicking the folder for the component and choosing Diff Styles Against Org is not a valid way either, as there is no such option in Visual Studio Code or Salesforce CLI. Right-clicking the folder for the component and choosing Diff Files Against Org is not a valid way either, as it will compare all the files in the folder, not just the CSS file, which may not be efficient or necessary. Salesforce [Salesforce CLI Command force:source:diff], [Salesforce Developer Tools for Visual Studio Code]

NEW QUESTION # 140

Which two steps are necessary to enable Salesforce B2B Commerce logging in the managed package?

- A. Turn On the Checkbox "Cloudcraze Logging" in CC Admin.
- **B. Ensure the value saved in the Logging token field is appended to the ccLog query parameter.**
- C. Set a cookie with the Id of the user accessing the storefront in CC Admin
- **D. Ensure you save a value in the Logging Token input field in the Global Settings section of CC Admin.**

Answer: B,D

Explanation:

To enable Salesforce B2B Commerce logging in the managed package, you need to do two steps. First, you need to save a value in the Logging Token input field in the Global Settings section of CC Admin. This value can be any string that you choose, such as "debug". Second, you need to ensure that the value saved in the Logging token field is appended to the ccLog query parameter in the URL of the storefront page that you want to debug. For example, if your logging token is "debug", then your URL should look like this: <https://my-storefront.com/?ccLog=debug>. This will enable logging for that page only. You do not need to turn on the checkbox "Cloudcraze Logging" in CC Admin, as this is an old setting that is no longer used. You also do not need to set a cookie with the Id of the user accessing the storefront in CC Admin, as this is not required for logging. Salesforce [B2B Commerce Developer Guide: Logging]

NEW QUESTION # 141

For which two reasons is it preferable to extend the Salesforce B2B Commerce remote invocation object instead of using the standard Salesforce remote action invocation manager (2 answers)

- **A. The Salesforce B2B Commerce logger cannot be utilized in standard remote actions**
- B. A standard remote action will not have access to Salesforce B2B Commerce objects.
- **C. The APEX method called by the remote action will be passed as a Salesforce B2B Commerce context object.**
- D. Salesforce B2B Commerce includes do not support standard Salesforce remote actions.

Answer: A,C

Explanation:

It is preferable to extend the Salesforce B2B Commerce remote invocation object instead of using the standard Salesforce remote action invocation manager for two reasons:

The APEX method called by the remote action will be passed as a Salesforce B2B Commerce context object, which contains useful information such as the current user, cart, storefront, and configuration settings. This can simplify the development and testing of the remote action.

The Salesforce B2B Commerce logger can be utilized in the remote action, which allows logging messages and errors to the debug log or to a custom object. This can facilitate debugging and troubleshooting of the remote action.

