

# 権威のあるAP-202復習範囲と高品質のAP-202資格勉強



あなたはSalesforceのAP-202の資料を探すのに悩んでいますか。心配しないでください。私たちを見つけるのはあなたのSalesforceのAP-202試験に合格する保障からです。数年以来IT認証試験のためのソフトを開発している我々Jpexamチームは国際的に大好評を博しています。我々はSalesforceのAP-202のような重要な試験を準備しているあなたに一番全面的で有効なヘルプを提供します。

SalesforceのAP-202試験に関する権威のある学習教材を見つけられないで、悩んでいますか？世界中での各地の人々はほとんどSalesforceのAP-202試験を受験しています。SalesforceのAP-202の認証試験の高品質の資料を提供しているユニークなサイトはJpexamです。もし君はまだ心配することがあったら、私たちのSalesforceのAP-202問題集を購入する前に、一部分のフリーな試験問題と解答をダウンロードして、試用してみることができます。

>> AP-202復習範囲 <<

## 更新のSalesforce AP-202復習範囲は主要材料 & 有効的なAP-202: B2B Commerce for Developers Accredited Professional

Jpexamについてどのくらい知っているのですか。JpexamのAP-202試験問題集を利用したことがありますか。あるいは、知人からJpexamを聞いたことがありますか。IT認定試験に関連する参考書のプロな提供者として、Jpexamは間違いなくあなたが今まで見た最高のサイトです。なぜこのように確かめるのですか。それはJpexamのように最良のAP-202試験参考書を提供してあなたに試験に合格させるだけでなく、最高品質のサービスを提供してあなたに100%満足させることもできるサイトがないからです。

## Salesforce B2B Commerce for Developers Accredited Professional 認定 AP-202 試験問題 (Q35-Q40):

### 質問 # 35

In which two ways can events fired from Lightning web components be handled?

- A. Attaching handlers to DOM elements
- B. Adding callbacks to components
- C. Listening for all possible events at the document root
- D. Programmatically adding event listeners

正解: A、D

解説:

Two ways that events fired from Lightning web components can be handled are programmatically adding event listeners and attaching handlers to DOM elements. Programmatically adding event listeners is a way of handling events by using JavaScript code to register functions that are invoked when an event occurs. The developer can use methods such as `addEventListener` or `@wire` to add event listeners to components or services that fire events. Attaching handlers to DOM elements is a way of handling events by using HTML attributes to bind functions that are invoked when an event occurs. The developer can use attributes such as `onclick` or `onchange` to attach handlers to DOM elements that fire events. Adding callbacks to components is not a valid way of handling events fired from Lightning web components, as it is not related to event handling, but rather to asynchronous programming. Listening for all possible events at the document root is not a valid way either, as it is not efficient or recommended for event handling, as it can cause performance issues or conflicts with other event listeners. Salesforce [Lightning Web Components Developer Guide: Handle Events], [Lightning Web Components Developer Guide: Communicate with Events]

### 質問 # 36

Which method needs to be implemented when rendering a Salesforce B2B Commerce view in order to have it called after rendering has finished?

- A. `onRender()`
- B. `postRender()`
- C. There are no methods called on the view after rendering has finished
- D. `afterRender()`

正解: B

解説:

The method that needs to be implemented when rendering a Salesforce B2B Commerce view in order to have it called after rendering has finished is `postRender`. This method is an optional hook that can be defined by the user to perform any actions or logic that depend on the view being rendered, such as initializing widgets, binding events, or updating the user interface. The method will be called by the render method of the `CCRZ.View` class, which is the base class for all views in the framework. Salesforce B2B Commerce and D2C Commerce Developer Guide, View Class

### 質問 # 37

A developer has made a component with a lightning combobox in the follow! markup. To handle changes on the combobox, what should replace `<CHANGE FVENT>`?

- A. `{handleChange()}`
- B. `{event.handleChange}`
- C. `{handleChange}`
- D. `javascript:void(0);handleChange();`

正解: C

解説:

To handle changes on the combobox, the developer should replace `<CHANGE EVENT>` with `{handleChange}`. The `handleChange` is a function that is defined in the JavaScript file of the Lightning web component and is invoked when the value of the combobox changes. The developer can use this function to perform custom logic or actions based on the selected value of the

combobox, such as updating other components or properties, calling Apex methods or services, or firing events. The developer can use the onchange attribute to bind the handleChange function to the combobox element in the HTML file of the Lightning web component. The onchange attribute is an HTML attribute that specifies a function to be executed when the value of an element changes. {event.handleChange} is not a valid way to handle changes on the combobox, as it is not a valid syntax for binding a function to an element. javascript:void(0);handleChange(); is not a valid way either, as it is not a valid syntax for binding a function to an element. {handleChange()} is not a valid way either, as it is not a valid syntax for binding a function to an element. Salesforce Lightning Web Components Developer Guide: Handle Events, Lightning Web Components Developer Guide: Communicate with Properties, Lightning Web Components Developer Guide: lightning-combobox

#### 質問 # 38

Which three files comprise the essential pieces of a Lightning Web Component that is named myComponent?

- A. myComponent.html
- B. myNewComponent.css
- C. myComponent.js
- D. myComponent.aura
- E. myComponent.js-meta.xml

正解: A、C、E

解説:

A Lightning Web Component (LWC) consists of a minimum of three core files: the template file (.html), the JavaScript file (.js), and the metadata configuration file (.js-meta.xml). These files are essential for defining the structure, functionality, and configuration of the component. For more information, review the Salesforce LWC documentation: Salesforce LWC Documentation.

#### 質問 # 39

What are three ways to implement custom post Order processing? (3 answers)

- A. Modify or add custom Cart formula fields to handle logic.
- B. Use Process builder to implement business processes that execute when an Order record is created.
- C. Extend cc\_hk\_invoice to handle custom business logic post Order processing
- D. Use cc\_hk\_Order.placeTarget to define a new Order Confirmation page which executes additional business logic.
- E. Use a Salesforce workflow rule that executes when an Order record is created.

正解: B、C、D

解説:

Three ways to implement custom post Order processing are:

Extend cc\_hk\_invoice to handle custom business logic post Order processing. This hook allows modifying the invoice data or performing additional actions after an Order is placed. For example, the hook can send an email notification or update a custom field on the invoice record.

Use cc\_hk\_Order.placeTarget to define a new Order Confirmation page which executes additional business logic. This hook allows specifying a custom Visualforce page that will be displayed after an Order is placed. The custom page can include additional business logic or user interface elements.

Use Process Builder to implement business processes that execute when an Order record is created. Process Builder is a tool that allows creating workflows and actions based on criteria and conditions. For example, Process Builder can create a task, send an email, or update a record when an Order record is created. Salesforce B2B Commerce and D2C Commerce Developer Guide, Hooks, Process Builder

#### 質問 # 40

.....

常に Salesforce AP-202 試験に参加する予定があるお客様は「こちらの問題集には、全部で何問位、掲載されておりますか?」といった質問を提出しました。心配なくて我々 Jpexam の Salesforce AP-202 試験問題集は実際試験のすべての問題種類をカバーします。70% の問題は解説がありますし、試験の内容を理解しやすいと助けます。

AP-202 資格勉強: [https://www.jpexam.com/AP-202\\_exam.html](https://www.jpexam.com/AP-202_exam.html)

