



**Free Questions for B2B-Commerce-Administrator by  
actualtestdumps**

**Shared by Santiago on 12-12-2023**

**For More Free Questions and Preparation Resources**

**Check the Links on Last Page**

## Question 1

---

**Question Type:** MultipleChoice

---

What needs to be done first, in order to modify the out-of-the-box checkout flow template?

### Options:

---

- A- Clone the existing checkout flow
- B- Change the permissions on the existing checkout flow
- C- Rename the existing Checkout Flow in place
- D- Delete the existing flow

### Answer:

---

A

## Question 2

---

**Question Type:** MultipleChoice

---

A developer is working on an existing checkout information built against the Lightning Web Runtime (LWR) and wants to implement a custom child checkout component to modify out-of-the-box functionality.

Which interface must the developer implement for the child component?

**Options:**

---

- A- CheckoutSavable
- B- CheckoutInterface
- C- CustomCheckout
- D- CheckoutStep

**Answer:**

---

D

**Explanation:**

---

To implement a custom child checkout component to modify out-of-the-box functionality in an LWR checkout flow, the developer must implement the CheckoutStep interface. This interface defines the contract between the child component and the parent Checkout component, and allows the child component to communicate with the checkout flow. The developer must implement the methods of this interface, such as getStepName, getStepLabel, getStepStatus, validateStep, and so on. The developer can also use the

@wire(CheckoutApi) decorator to access the checkout API methods and data.Reference:

[Customize Your Checkout Flow for a B2B Store \(LWR\)](#)

[CheckoutStep Interface](#)

## Question 3

---

**Question Type: MultipleChoice**

---

Which three components should a developer use on the product page to replace the out-of-the-box Product Detail Card component?

### Options:

---

- A- Product Detail Breadcrumbs component
- B- Product Fields Short component
- C- Product Field Long component
- D- Product Detail Purchase Options component
- E- Product Detail Image Gallery component

**Answer:**

---

B, C, E

**Explanation:**

---

To replace the out-of-the-box Product Detail Card component on the product page, a developer can use three components: Product Fields Short component, Product Field Long component, and Product Detail Image Gallery component. These components are part of the standard Lightning web components that are available for B2B Commerce stores. They allow the developer to customize the product page layout and display different product fields and images<sup>3</sup>. The Product Fields Short component displays short text fields from the product object, such as name, SKU, or price<sup>4</sup>. The Product Field Long component displays long text fields from the product object, such as description or specifications<sup>5</sup>. The Product Detail Image Gallery component displays a carousel of product images from the product object or from a custom field<sup>6</sup>. The other options are not suitable for replacing the Product Detail Card component, as they serve different purposes. The Product Detail Breadcrumbs component displays the navigation path from the home page to the current product page<sup>7</sup>. The Product Detail Purchase Options component displays a button to add the product to the cart or to request a quote<sup>8</sup>. Reference:<sup>3</sup>: Lightning Web Components for B2B Commerce - Trailhead,<sup>4</sup>: Product Fields Short Component - Salesforce Developers,<sup>5</sup>: Product Field Long Component - Salesforce Developers,<sup>6</sup>: Product Detail Image Gallery Component - Salesforce Developers,<sup>7</sup>: Product Detail Breadcrumbs Component - Salesforce Developers,<sup>8</sup>: Product Detail Purchase Options Component - Salesforce Developers

## Question 4

---

**Question Type:** MultipleChoice

---

A developer has written the logic to import products from an enterprise resource planning (ERP) platform into B2B storefront products. The imported

products are in Salesforce, but they are not visible in the store.

What did the developer forget to assign to the imported products?

### Options:

---

- A- Entitlement policy
- B- Storefront
- C- Account
- D- Promotion

### Answer:

---

B

### Explanation:

---

To make the imported products visible in the store, the developer must assign them to a storefront. A storefront is a record that defines the properties and settings of a B2B Commerce store, such as its name, URL, theme, currency, and language. A storefront also has a relationship with the Product2 object, which allows the developer to specify which products are available for the store. The developer can

assign products to a storefront by using the Storefronts related list on the product detail page, or by using the Products related list on the storefront detail page<sup>1</sup>.The other options are not required to make the products visible in the store, although they may affect other aspects of the product display or functionality.Reference:1: Create a Storefront - Trailhead<sup>2</sup>

## Question 5

---

**Question Type:** MultipleChoice

---

Which two log levels does a developer need to include to debug platform events?

### Options:

---

- A- Apex Code
- B- Database
- C- Callout
- D- Workflow

### Answer:

---

A, D

### Explanation:

---

To debug platform events, a developer needs to include two log levels: Apex Code and Workflow. The Apex Code log level captures the execution of Apex triggers that subscribe to platform events. The Workflow log level captures the execution of processes and flows that subscribe to platform events. By setting these log levels to FINEST, the developer can see detailed information about the platform event messages, such as their fields, values, replay IDs, and delivery statuses<sup>5</sup>. The other log levels are not relevant for debugging platform events, as they capture different aspects of system activity, such as database operations or callouts<sup>6</sup>. Reference:<sup>5</sup>: Set Up Debug Logs for Event Subscriptions - Platform Events Developer Guide<sup>7</sup>,<sup>6</sup>: Debug Log Levels - Salesforce Help

## Question 6

---

### Question Type: MultipleChoice

---

What are two ways a developer should ensure that a store verifies shipping charges by using an external service?

### Options:

---

- A-** Create a flow using an action to retrieve shipping charges from an external service and update the Cart Delivery Group Methods.
- B-** Create an Apex class implementing the `sfdc_checkout.CartShippingCharges` interface to retrieve shipping charges from an external



services and register it as as the ..Calculation integration in the store administration.

**C-** Create an Apex class to retrieve shipping charges from an external service and update the Cart Delivery group methods.

**D-** Create a trigger to retrieve shipping charges from an external service and update the Cart Delivery Group Methods.

### **Answer:**

---

B, C

### **Explanation:**

---

To ensure that a store verifies shipping charges by using an external service, a developer can use two ways:

Create an Apex class implementing the `sfdc_checkout.CartShippingCharges` interface to retrieve shipping charges from an external service and register it as the Shipping Calculation integration in the store administration. The `sfdc_checkout.CartShippingCharges` interface defines the methods that a custom shipping calculation class must implement to calculate shipping charges based on the cart items, delivery groups, and shipping methods. The developer can use this interface to call an external service API and return the shipping charges for each delivery group. The developer must also register the custom Apex class as the Shipping Calculation integration in the store administration, so that it can be used by the store to verify shipping charges during checkout<sup>1</sup>.

Create an Apex class to retrieve shipping charges from an external service and update the Cart Delivery Group Methods. The Cart Delivery Group Methods object stores the shipping methods and charges for each delivery group in a cart. The developer can create an Apex class that queries this object and updates the shipping charges based on the response from an external service API. The developer can invoke this Apex class from a trigger, a process, or a flow that runs when a cart is updated or submitted<sup>2</sup>. Reference: <sup>1</sup>: Add a Shipping Calculation Service for Commerce Stores - Salesforce Help<sup>3</sup>, <sup>2</sup>: CartDeliveryGroupMethod Object Reference - Salesforce Help<sup>4</sup>

## Question 7

---

**Question Type:** MultipleChoice

---

A product is assigned to the entitlement policy but is missing from the Price Book related to the Buyer Group.

The entitlement policy has View products and View prices in catalog checked.

How will the product behave on the B2B Portal?

### Options:

---

- A-** The product will not appear on the Portal but can be searched since it is part of the entitlement policy.
- B-** The product will appear on the Portal with Price Unavailable status and can also be added to the cart.
- C-** The product will not appear on the Portal and therefore cannot be added to the cart.
- D-** The product will appear on the Portal with Price Unavailable status but cannot be added to the cart.

### Answer:

---

C

## **Explanation:**

---

The product will not appear on the Portal and therefore cannot be added to the cart. This is because the product is not assigned to the price book that is related to the buyer group. The entitlement policy only determines which products the buyer group can see, but the price book determines which products the buyer group can buy. If a product is not in the price book, it will not be displayed on the Portal, regardless of the entitlement policy settings. Reference:

[Organize How Customers Access Your Commerce Store](#)

[Set Up Pricing for Commerce Stores](#)

## **Question 8**

---

**Question Type:** MultipleChoice

---

A developer is on a tight timeline and needs to implement a Lightning web component which can read, create and modify single records. What is the recommended path forward?

**Options:**

---

- A- Use base components
- B- Write custom functions against a wire adapter
- C- Create an Apex Controller
- D- Use Lightning Data Service

**Answer:**

---

A

**Explanation:**

---

The recommended path forward for a developer who needs to implement a Lightning web component that can read, create, and modify single records is to use base components. Base components are prebuilt Lightning web components that provide common functionality and user interface elements, such as forms, buttons, inputs, outputs, and more. Some of the base components, such as lightning-record-edit-form, lightning-record-form, and lightning-record-view-form, use Lightning Data Service to access and manipulate Salesforce data without requiring any Apex code. These components can handle CRUD operations on single records, as well as validation, layout, and error handling<sup>3</sup>. Using base components can save time and effort for the developer, as well as ensure consistency and performance across the application<sup>4</sup>. Reference:<sup>3</sup>: Work with Salesforce Records - Trailhead<sup>5</sup>,<sup>4</sup>: Use Base Lightning Components - Trailhead<sup>6</sup>

## Question 9

---

**Question Type: MultipleChoice**

---

An administrator has just provided a developer with a completely new org and a username. Assuming the username is me@my-new-org.org, what is the correct set of steps to authorize the org for Command Line

Interface (CLI) access so the developer can deploy Lightning web components?

**Options:**

---

- A-** Run the command: 'sfdx force:login -r 'https://login.salesforce.com"' and supply the credentials in the browser when it opens.
- B-** Run the command 'sfdx force:auth:web:login -a 'https://login.salesforce.com and then supply the credentials in the browser when it opens.
- C-** Run the command: 'sfdx force:auth:web:login -r 'https://login.salesforce.com and then supply the credentials in the browser when it opens
- D-** Run the command 'sfdx force:auth:web:login -r 'https://login.salesforce.com' -username='me@my-new-org.org'

**Answer:**

---

C

**Explanation:**

---

The correct set of steps to authorize the org for Command Line Interface (CLI) access so the developer can deploy Lightning web components is to run the command: 'sfdx force:auth:web:login -r "https://login.salesforce.com"' and then supply the credentials in the browser when it opens. This command will open a browser window and prompt the user to log in to the org using their username and password. The -r flag specifies the login URL for the org, which is https://login.salesforce.com for production or developer orgs. The command will also create an alias for the org, which can be used to reference it in other CLI commands<sup>1</sup>. Reference: <sup>1</sup>: Authorize an Org - Trailhead<sup>2</sup>

## Question 10

---

**Question Type: MultipleChoice**

---

A developer has made a component with a lightning combobox in the following markup. To handle changes on the combobox, what should replace ?

```
<template>
  <lightning-combobox
    name="billingAddressSelector"
    label={title}
    options={selectableAddresses}
    value={initialSelectedAddressId}
    class="slds-p-bottom_medium"
    onchange=<CHANGE_EVENT>
    required={billingAddressRequired}
  >
  </lightning-combobox>
</template>
```

### Options:

---

- A- {eventhandleChange}
- B- Javascript:void(0);handleChange();
- C- {handleChange()}
- D- {handleChange}

### Answer:

---

D

### Explanation:

---

To handle changes on the combobox, the developer should use {handleChange} as the value for the onchange attribute. This syntax binds the handleChange function defined in the JavaScript file of the component to the change event fired by the combobox. The developer can then access the selected value from the event object passed to the function and perform any logic or action based on it. Reference:

[lightning-combobox - documentation](#)

Handle Events in Lightning Web Components

## Question 11

---

**Question Type:** MultipleChoice

---

Which two components in a B2B store template should a developer use to customize a storefront page? 01m 52s

**Options:**

---



**A-** My Lists

**B-** Product List

**C-** Order List

**D-** Address List

### **Answer:**

---

B, D

### **Explanation:**

---

To customize a storefront page, the developer can use the following components in a B2B store template:

**Product List.** This component displays a list of products that match the search criteria or category selection of the buyer. The developer can customize the layout, filters, sorting, pagination, and actions of this component using the Product List configuration module in CC Admin.

**Address List.** This component displays a list of addresses that are associated with the buyer's account. The developer can customize the appearance, behavior, and validation of this component using the Address List configuration module in CC Admin. Reference:

Customize Your Storefront with Aura Components

[Product List Configuration Module](#)

[Address List Configuration Module](#)



**To Get Premium Files for B2B-Commerce-Administrator Visit**

**<https://www.p2pexams.com/products/b2b-commerce-administrator>**

**For More Free Questions Visit**

**<https://www.p2pexams.com/salesforce/pdf/b2b-commerce-administrator>**

