

Salesforce.B2B Commerce Developer.by.Jack.27q

Number: B2B Commerce Developer

Passing Score: 800

Time Limit: 120

File Version: 2.0

Exam Code: B2B Commerce Developer

Exam Name: Salesforce Accredited B2B Commerce Developer

Exam A

QUESTION 1

A user wants to have a customized experience for adding items to the cart. The user also wants the mini cart module to reflect changes to the state of the cart afterwards. How should this requirement be fulfilled?

- A. Leverage the Addto Cart Global API which add items to the cart and also refreshes the page with the new data.
- B. Trigger the global cartChange' event and then trigger changeMiniCart' event after the Add to Cart Action on the custom button.
- C. Write a custom Remote Action to refresh the Mini Cart and refresh the Cart Line item count on the Cart Link in the header.
- D. Trigger the global cartChange' event after the Add to Cart Action on the custom button.

Correct Answer: D

Section:

Explanation:

To have a customized experience for adding items to the cart and also update the mini cart module, the requirement can be fulfilled by triggering the global "cartChange" event after the Add to Cart Action on the custom button. This event will notify all the components that are listening to it that the cart has changed and they should refresh their data accordingly. The mini cart module is one of these components, so it will update its state based on the new cart data. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Events

QUESTION 2

A user wants to leverage a three column layout on a page. The user also wants to move the mini-cart widget from the right to the center column. How can this requirement be fulfilled?

- A. Gross Layout Override
- B. Subscriber Template
- C. Page Include
- D. HandleBar Template Override

Correct Answer: A

Section:

Explanation:

To leverage a three column layout on a page and move the mini-cart widget from the right to the center column, the requirement can be fulfilled by creating a Gross Layout Override. This is a custom Visualforce page that overrides the default layout of a page and allows changing its structure and content. The user can create a Gross Layout Override that uses a three column layout and places the mini-cart widget in the center column. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Gross Layout Overrides

QUESTION 3

What are the templating, Javascript, and CSS frameworks what the cloudcraze managed package leverages?

- A. Angularjs, Backbonejs, and handlebarsjs
- B. Bootstrap, Backbonejs, and handlebarsjs
- C. Bootstrap, Angularjs, and Backbonejs
- D. Angularjs, react.js, and handlebarsjs

Correct Answer: B

Section:**Explanation:**

The templating, JavaScript, and CSS frameworks that the cloudcraze managed package leverages are Bootstrap, Backbone.js, and Handlebars.js. Bootstrap is a CSS framework that provides responsive design and layout components. Backbone.js is a JavaScript framework that provides models, views, collections, and events for building single-page applications. Handlebars.js is a templating engine that allows generating HTML from JSON data. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Front-End Development

QUESTION 4

What are three advantages of using ccLog over the Salesforce standard System.debug class? (3 answers)

- A. There is no need to use string concatenation to easily tag log statements with a subject.
- B. ccLog can debug syntax errors found in the JavaScript.
- C. There is no need to create a User Trace Flag.
- D. Append #ccLog=<Logging Token Name> to the end of the storefront URL in order to get logs in the inspector console.
- E. There is no need to manually set a cookie to debug with the Site Guest User.

Correct Answer: A, D, E

Section:**Explanation:**

Three advantages of using ccLog over the Salesforce standard System.debug class are:

There is no need to use string concatenation to easily tag log statements with a subject. ccLog allows passing a subject parameter to the log method, which will prepend the subject to the log message. For example, `ccLog.log('This is a message', 'Subject')` will log `[Subject] This is a message`.

There is no need to create a User Trace Flag. ccLog can be enabled by setting the value of `CO.logToken` to true in `CCAdmin`, which will activate logging for all users who access the storefront.

There is no need to manually set a cookie to debug with the Site Guest User. ccLog can be enabled for the Site Guest User by appending `#ccLog=<Logging Token Name>` to the end of the storefront URL in order to get logs in the inspector console. For example, `https://my-storefront.com/#ccLog=debug` will enable logging for the Site Guest User with the debug level. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Logging

QUESTION 5

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

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

Correct Answer: B, C, E

Section:**Explanation:**

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

Reference: B2B Commerce and D2C Commerce Developer Guide, Hooks, Process Builder

QUESTION 6

What are three ways to test the value of Page Label on any Salesforce B2B Commerce Community Page? (3 answers)

- A. Access the source HTML for the page via the browser developer tools.
- B. Execute `CCRZ.pagevars.pageLabels['PAGE_LABEL_NAME']` in the JavaScript console.
- C. Execute `CCRZ.processPageLabelMap('PAGE_LABEL_NAME')` in the JavaScript console.
- D. Enable the 'display page label names' in cc admin.
- E. Execute `(('PAGE_LABEL_NAME'))` in the JavaScript console.

Correct Answer: B, C, D

Section:

Explanation:

Three ways to test the value of Page Label on any Salesforce B2B Commerce Community Page are:

Execute `CCRZ.pagevars.pageLabels['PAGE_LABEL_NAME']` in the JavaScript console. This will return the value of the page label with the given name from the `pagevars` object, which contains all the page labels that are used on the page.

Execute `CCRZ.processPageLabelMap('PAGE_LABEL_NAME')` in the JavaScript console. This will return the value of the page label with the given name from the `pageLabelMap` object, which contains all the page labels that are defined in CCAdmin.

Enable the 'display page label names' in cc admin. This will display the name of each page label next to its value on the storefront pages, which can help identify and verify the page labels. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Page Labels

QUESTION 7

What are two guidelines for logging that are used within the core Salesforce B2B Commerce product? (2 answers)

- A. Items or data within computational intensive loops should be logged.
- B. The close method of `ccrz.ccLog` must be called at the end of the remote action.
- C. No calls to `ccrz.ccLog` can be made before `cc_CallContext.initRemoteContext` is executed.
- D. It is okay to log any data on the server that is already logged on the client side.

Correct Answer: B, C

Section:

Explanation:

Two guidelines for logging that are used within the core Salesforce B2B Commerce product are:

The close method of `ccrz.ccLog` must be called at the end of the remote action. This method will flush the log messages to the browser console or to a custom object, depending on the logging mode. If this method is not called, the log messages may not be displayed or saved properly.

No calls to `ccrz.ccLog` can be made before `cc_CallContext.initRemoteContext` is executed. This method will initialize the call context object, which contains information such as the current user, cart, storefront, and configuration settings. These information are required for logging, so calling `ccrz.ccLog` before initializing the call context will result in an error. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Logging

QUESTION 8

What is a best practice when passing query parameters from user interface to an apex controller?

- A. Query parameters should be properly sanitized by using `JSINHTMLCODE` within the VisualForce Page or Component.
- B. String parameters should be trimmed using `String.trim()`.

- C. Query parameters should be passed only to Salesforce B2B Commerce classes that you are extending.
- D. Query parameters should be stored on a backbone model prior to passing them to the server

Correct Answer: A

Section:

Explanation:

A best practice when passing query parameters from user interface to an apex controller is to query parameters should be properly sanitized by using JSINHTMLENCODE within the VisualForce Page or Component. This function will encode any special characters in the query parameters to prevent cross-site scripting (XSS) attacks or SOQL injection attacks. For example, `ccrz.ccRemoteActions.getProducts('{!JSINHTMLENCODE(searchTerm)}')` will encode the searchTerm parameter before passing it to the apex controller. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, Security

QUESTION 9

What is a method to resolve if the current storefront customer is a Salesforce B2B Commerce guest user in an apex class?

- A. `ccrz.cc_CallContext.currUser.isGuest`
- B. `ccrz.cc_CallContext.isGuest`
- C. `UserInfo.getUserType()`
- D. `... UserType`

Correct Answer: B

Section:

Explanation:

A method to resolve if the current storefront customer is a Salesforce B2B Commerce guest user in an apex class is to use `ccrz.cc_CallContext.isGuest`. This property will return true if the current user is a guest user, or false otherwise. For example, `if(ccrz.cc_CallContext.isGuest){ // do something for guest user }` will execute some logic only for guest users. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, Call Context

QUESTION 10

What is a valid way of referencing the CC Cart Object whose API name is `E_Cart__c` in a SOQL query?

- A. `_Cart__c`
- B. `c.E_Cart__c`
- C. `ccrz__E_Cart__c`
- D. `cloudcraze__E_Cart__c`

Correct Answer: C

Section:

Explanation:

A valid way of referencing the CC Cart Object whose API name is `E_Cart__c` in a SOQL query is to use `ccrz__E_Cart__c`. This is the transformed name of the object that is used by the Salesforce B2B Commerce framework. All custom objects and fields that are part of the cloudcraze managed package have the prefix `ccrz__` in their API names. For example, `SELECT Id, Name FROM ccrz__E_Cart__c` will query the CC Cart Object records. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, Query Transformation

QUESTION 11

What is a valid way of referencing the global `cc_api_CartExtension` apex class via subscriber code?

- A. `ccrz__cc_api_CartExtension`

- B. c__cc_api_CartExtension
- C. cloudcraze.cc_api_CartExtension
- D. ccrz.cc_api_CartExtension

Correct Answer: D

Section:

Explanation:

A valid way of referencing the global cc_api_CartExtension apex class via subscriber code is to use ccrz.cc_api_CartExtension. This is the name of the class that is defined in the cloudcraze managed package. The class is global, so it can be accessed by subscriber code. The other options are either invalid or incorrect. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, API Classes

QUESTION 12

What is default behavior for how the Salesforce B2B Commerce Global APIs transform Salesforce data?

- A. Fields names are returned using the Salesforce naming convention.
- B. Fields names are returned with c.' prepended in their name.
- C. Fields names are returned with a lowercase first letter, camelcase convention
- D. Fields names can be mapped to any naming convention desired

Correct Answer: C

Section:

Explanation:

The default behavior for how the Salesforce B2B Commerce Global APIs transform Salesforce data is to return field names with a lowercase first letter, camelcase convention. For example, the field name ccrz__E_Product__c in Salesforce will be transformed to eProduct in the API. This is done to follow the JavaScript naming convention and to avoid conflicts with the standard Salesforce fields and relationships. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Query Transformation

QUESTION 13

What is essential for a Salesforce B2B Commerce theme to show up in the theme section in CC Admin?

- A. The theme needs to be set as a Custom Setting in Salesforce.
- B. The theme needs to be set in the Configuration Settings.
- C. The theme needs to have 'theme' in the name of the Static Resource.
- D. The theme needs to be referred to in the head element on the page

Correct Answer: C

Section:

Explanation:

An essential requirement for a Salesforce B2B Commerce theme to show up in the theme section in CC Admin is that the theme needs to have "theme" in the name of the Static Resource. For example, a theme named "MyTheme" will not appear in CC Admin, but a theme named "MyTheme_theme" will. This is how the framework identifies which static resources are themes and which are not. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Themes

QUESTION 14

What is the difference between Gross Layout Overrides and Subscriber Templates?

- A. Gross Layout Overrides allow modification to CSS of a page, while Subscriber Templates allows for modification of the entire page including header and footer.

- B. Subscriber Templates allows for modification of the header, the footer and the content in between them. Gross Layout Overrides only allow for modification of the header and footer.
- C. Subscriber Templates allow for modification of the header and the footer, while Gross Layout Overrides allow for modification everything inside the header and footer.
- D. Gross Layout Overrides allow for the modification of the footer, while Subscriber Templates allow for modification of everything inside the header and footer.

Correct Answer: C

Section:

Explanation:

The difference between Gross Layout Overrides and Subscriber Templates is that Subscriber Templates allow for modification of the header and the footer, while Gross Layout Overrides allow for modification everything inside the header and footer. Subscriber Templates are custom Visualforce pages that can replace the standard header and footer components on a page. Gross Layout Overrides are custom Visualforce pages that can replace the default layout of a page, such as the number of columns, the position of widgets, or the content of sections. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, Subscriber Templates, Gross Layout Overrides

QUESTION 15

What is the recommended method for increasing the number of required autocomplete characters that are typed before autocomplete works?

- A. Override and extend the autoComplete method in cc_hk_Catalog.
- B. Override the autoComplete.search_input.minLength value in the CCRZ.uiProperties file
- C. Override the autocomplete template and create and register a new handlebars helper.
- D. Update the...minLength property in CC Admin, then build and activate a new cache.

Correct Answer: B

Section:

Explanation:

The recommended method for increasing the number of required autocomplete characters that are typed before autocomplete works is to override the autoComplete.search_input.minLength value in the CCRZ.uiProperties file. This file contains various properties that control the behavior and appearance of the user interface components. The autoComplete.search_input.minLength property specifies the minimum number of characters that must be entered before the autocomplete feature is triggered. The default value is 3, but it can be changed to any desired value by overriding it in the CCRZ.uiProperties file. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, UI Properties

QUESTION 16

What is true regarding adding more Configuration Settings to Salesforce B2B Commerce?

- A. Metadata can be added to existing modules, but you cannot add new modules.
- B. Configuration settings can only be extended through API's
- C. Select 'New' in your storefront's Configuration Settings and create a custom setting.
- D. More modules and metadata can be added to Salesforce B2B Commerce.

Correct Answer: D

Section:

Explanation:

More modules and metadata can be added to Salesforce B2B Commerce by creating custom configuration settings. Configuration settings are custom settings that store various values and parameters that affect the functionality and appearance of the storefront. They are organized into modules, which group related settings together. To create a custom configuration setting, the user needs to create a custom setting record in Salesforce and specify its module, name, value, and description. The custom setting will then appear in CCAdmin under the specified module. Salesforce Reference: B2B Commerce and D2C Commerce Developer Guide, Configuration Settings

QUESTION 17

Which method is used to override when extending the Salesforce B2B Commerce logic providers?

- A. doLogic
- B. process
- C. doAction
- D. fetch

Correct Answer: C

Section:

Explanation:

The method that is used to override when extending the Salesforce B2B Commerce logic providers is doLogic. This method is responsible for executing the business logic for each logic provider class, such as validating inputs, performing calculations, applying rules, or updating data. By overriding this method, the user can modify or extend the existing logic or add their own custom logic. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Logic Classes

QUESTION 18

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

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

Correct Answer: C

Section:

Explanation:

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

Reference: B2B Commerce and D2C Commerce Developer Guide, View Class

QUESTION 19

Which method signature is used in the Global API's?

- A. Changes based on API and Method name
- B. ccrz.cc_Output (ccrz.cc_Input input)
- C. Map<String, Object>
- D. List<List<Object>>

Correct Answer: B

Section:

Explanation:

The method signature that is used in the Global API's is ccrz.cc_Output (ccrz.cc_Input input). This signature indicates that the Global API methods take a single input parameter of type ccrz.cc_Input and return an output of type ccrz.cc_Output. These are custom classes that are defined in the cloudcraze managed package and contain various properties and methods for handling the input and output data. For example, ccrz.ccServiceProduct.getProducts(ccrz.cc_Input input) is a Global API method that takes an input object and returns an output object containing product data. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, API Classes, cc_Input Class, cc_Output Class

QUESTION 20

Which out of the box Salesforce B2B Commerce page can give instructions to web crawlers from accessing specific Salesforce B2B Commerce pages?

- A. CCCat?SiteMap
- B. cc_RobotsTxt
- C. CCSiteIndex
- D. CCPage

Correct Answer: B

Section:

Explanation:

The out of the box Salesforce B2B Commerce page that can give instructions to web crawlers from accessing specific Salesforce B2B Commerce pages is cc_RobotsTxt. This is a Visualforce page that generates a robots.txt file, which is a text file that tells web crawlers which pages or files they can or can't request from a site. The page uses the configuration settings CO.RobotsTxtAllow and CO.RobotsTxtDisallow to specify which paths are allowed or disallowed for web crawlers. For example, User-agent: * Disallow: /CCCart will instruct web crawlers to not access the CCCart page. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Robots.txt File

QUESTION 21

Which Salesforce B2B Commerce object needs to have a record added when defining a new Subscriber Pages to be rendered in a CC Page?

- A. CC Storefront Association
- B. CC Admin
- C. CC Page Sections
- D. CC Subscriber Pages

Correct Answer: D

Section:

Explanation:

The Salesforce B2B Commerce object that needs to have a record added when defining a new Subscriber Page to be rendered in a CC Page is CC Subscriber Pages. This is a custom object that stores information about the Subscriber Pages, such as the name, description, URL, and Visualforce page. To create a new Subscriber Page, the user needs to create a new record in this object and specify the required fields. The user can then select the Subscriber Page from the CC Page Settings configuration in CCAdmin. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Subscriber Pages

QUESTION 22

Which service method should be overridden in order to allow 'without sharing' queries?

- A. ccrz.ccService.queryWithoutSharing()
- B. ccrz.ccAPI.queryService()
- C. ccrz.ccService.query()
- D. ccrz.ccService.initSVCDAO()

Correct Answer: D

Section:

Explanation:

The service method that should be overridden in order to allow "without sharing" queries is ccrz.ccService.initSVCDAO. This method is responsible for initializing the service data access object (SVCDAO) that is used by the service class to perform queries. By overriding this method, the user can specify the sharing mode of the SVCDAO, which will determine whether the queries respect or ignore the sharing rules of the current user. For example, ccrz.ccService.initSVCDAO(ccrz.ccAPI.SZ_WITHOUTSHARING) will initialize the SVCDAO with the without sharing mode. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Service Classes, ccService Class

QUESTION 23

Which three actions are applicable when extending a default Salesforce B2B Commerce page via a page include? (3 answers)

- A. Create a Service Class override to query the new page include.
- B. Create the VisualForce page you wish to include to the Salesforce B2B Commerce page.
- C. Prepend 'c.' to the name of the page referenced in the configuration setting.
- D. Create a configuration setting for enabling the page include and assigning the new page include via CC admin.
- E. Build and activate a new configuration cache setting via CC admin.

Correct Answer: B, D, E

Section:

Explanation:

Three actions that are applicable when extending a default Salesforce B2B Commerce page via a page include are:

Create the VisualForce page you wish to include to the Salesforce B2B Commerce page. This page will contain the custom content or logic that you want to add to the existing page. For example, you can create a VisualForce page named MyPageInclude that displays some additional information or functionality on the product detail page.

Create a configuration setting for enabling the page include and assigning the new page include via CC Admin. This setting will specify which page include you want to use for which page and section. For example, you can create a configuration setting named CO.MyPageInclude with the value PDV.BodyIncludeBegin.MyPageInclude, which means you want to use MyPageInclude as the body include begin for the product detail view (PDV) page.

Build and activate a new configuration cache setting via CC Admin. This action will refresh the cache and apply the changes to the storefront. You need to do this whenever you make any changes to the configuration settings or VisualForce pages. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Page Includes

QUESTION 24

Which three actions are applicable when modifying the number of steps required in the Salesforce Commerce Checkout flow? (3 answers)

- A. Perform a template override on the Checkout page.
- B. Add a page include to the checkout page.
- C. Build and activate a new configuration cache setting via CC admin.
- D. Set the value of the configuration setting defined as CO.useDef to TRUE
- E. Set the value of the configuration setting defined as CO.overrideFlow to TRUE.

Correct Answer: A, C, E

Section:

Explanation:

Three actions that are applicable when modifying the number of steps required in the Salesforce Commerce Checkout flow are:

Perform a template override on the Checkout page. This action will allow you to change the structure and content of the Checkout page, such as adding or removing sections, widgets, or fields. For example, you can override the checkout.handlebars template and modify it according to your requirements.

Set the value of the configuration setting defined as CO.overrideFlow to TRUE. This setting will enable you to use your own custom checkout flow instead of the default one. You need to set this value to true before you can modify the checkout flow.

Set the value of the configuration setting defined as CO.useDef to TRUE. This setting will enable you to use a single-page checkout flow instead of a multi-step checkout flow. You need to set this value to true if you want to reduce the number of steps in the checkout flow to one. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Checkout Flow

QUESTION 25

Which three attributes are true regarding Subscriber pages? (3 answers)

- A. Subscriber pages can include additional standard Salesforce B2B Commerce components such as featured products, category tree, and mini cart.
- B. All the user interface components must be created manually.
- C. Subscriber pages allow customers to quickly create new custom pages for their storefront.
- D. Out of the Box, Subscriber Pages are completely blank HTML pages.
- E. Standard Salesforce B2B Commerce components are automatically included on the page e.g. Header links, images, menu items, containers, etc.

Correct Answer: A, C, D

Section:

Explanation:

Three attributes that are true regarding Subscriber Pages are:

Subscriber Pages can include additional standard Salesforce B2B Commerce components such as featured products, category tree, and mini cart. These components can be added to the Subscriber Pages by using the `CCRZ.pagevars.pageLabels` and `CCRZ.pagevars.pageSections` objects, which contain the page labels and page sections that are defined in `CCAdmin`.

Subscriber Pages allow customers to quickly create new custom pages for their storefront. These pages can be used to display any content or functionality that is not available in the default pages, such as FAQs, testimonials, or promotions.

Out of the Box, Subscriber Pages are completely blank HTML pages. These pages do not have any predefined layout or content, so the user can customize them according to their requirements. The user can use Visualforce, HTML, CSS, JavaScript, or any other web technologies to create their own Subscriber Pages. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Subscriber Pages

QUESTION 26

Which three pages should be enabled for the Guest user profile for a storefront to have anonymous checkout? (3 answers)

- A. `CCPaymentInfo`
- B. `CheckoutNew`
- C. `OrderView`
- D. `Checkout`
- E. `OrderConfirmation`

Correct Answer: A, B, E

Section:

Explanation:

Three pages that should be enabled for the Guest user profile for a storefront to have anonymous checkout are:

`CCPaymentInfo`: This page allows the guest user to enter their payment information, such as credit card number, expiration date, and security code. The page also displays the order summary and total amount.

`CheckoutNew`: This page allows the guest user to enter their shipping and billing information, such as name, address, phone number, and email. The page also displays the cart items and shipping options.

`OrderConfirmation`: This page displays the confirmation message and order number after the guest user places their order. The page also provides a link to view the order details or print the invoice. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, Anonymous Checkout

QUESTION 27

Which three statements are true about Global API versioning? (3 answers)

- A. Calling in with an API version set to lower than 1 will result in an exceptional case where the exception class `ccrz.BelowMinAPIVersionException` will be returned to callers.
- B. There is no need to pass `API_VERSION` to the Global APIs, and based on the Salesforce B2B Commerce Managed Package version, Global APIs are able to figure out what version of the API to use.
- C. The API version is scoped at the Class API level and NOT at the method level.
- D. Minimum `API_VERSION` is 1 and the Maximum API version follows the releases. E.g. The maximum was 4 as of Salesforce B2B Commerce Release-4.5, 5 as of Salesforce B2B Commerce Release 4.6, etc.
- E. Calling in with an API version set to more than current maximum will result in exception case where the exception class `ccrz.ExceedsMaxAPIVersionException` will be returned to callers.

Correct Answer: A, D, E

Section:**Explanation:**

Three statements that are true about Global API versioning are:

Calling in with an API version set to lower than 1 will result in an exceptional case where the exception class `ccrz.BelowMinAPIVersionException` will be returned to callers. This exception indicates that the API version is not supported by the framework and the caller should use a higher API version.

The API version is scoped at the Class API level and NOT at the method level. This means that all the methods in a class will use the same API version that is specified by the caller. For example, if the caller passes an API version of 4 to `ccrz.ccServiceProduct.getProducts()`, then all the other methods in `ccrz.ccServiceProduct` will also use API version 4.

Calling in with an API version set to more than current maximum will result in exception case where the exception class `ccrz.ExceedsMaxAPIVersionException` will be returned to callers. This exception indicates that the API version is not supported by the framework and the caller should use a lower API version. Salesforce

Reference: B2B Commerce and D2C Commerce Developer Guide, API Versioning