Prepare Smart for Success Free Adobe AD0-E722 Exam Questions and Answers

Ready to pass faster? Grab free and updated Adobe Commerce Architect Master Exam PDF questions now. Get authentic AD0-E722 dumps packed with verified answers and secure your certification success with PrepBolt AD0-E722 exam pdf questions and answers.

Thank you for Downloading AD0-E722 exam PDF Demo

https://prepbolt.com/AD0-E722.html

QUESTIONS & ANSWERS

DEMO VERSION

(LIMITED CONTENT)

Question 1

Question Type: MultipleChoice

An Architect wants to create an Integration Test that does the following:

* Adds a product using a data fixture

* Executes $this->someLogic->execute($product) on the product

* Checks if the result is true.

$this->someLogic has the correct object assigned in the setup() method.

Product creation and the tested logic must be executed in the context of two different store views with IDs of 3 and 4, which have been created and are available for the test.

How should the Architect meet these requirements?

Options:

A- Create two test classes with one test method each. Use the @magentoExecuteinstoreContext 3 and $MagentoExecuteinStoreContext 4 annotations on the class level.
B- Create one test class with two test methods. Use the emagentostorecontext 3 annotation in one method and amagentostorecontext 4 in the other one.
C- Create one test class with one test method. Use the \Magento\TestFramework\store\ExecuteinstoreContext class once in the fixture and another time in the test.

Answer:

C

Explanation:

To create an integration test that executes different logic in different store views, the Architect needs to do the following steps:

Create one test class that extends \Magento\TestFramework\TestCase\AbstractController or \Magento\TestFramework\TestCase\AbstractBackendController, depending on the type of controller being tested1.

Create one test method that uses the @magentoDataFixture annotation to specify the data fixture file that creates the product2.

Use the \Magento\TestFramework\Store\ExecuteInStoreContext class to execute the fixture and the tested logic in different store views. This class has a method called executeInStoreContext, which takes two parameters: the store ID and a callable function.The callable function will be executed in the context of the given store ID, and then the original store ID will be restored3. For example:

PHPAI-generated code. Review and use carefully.More info on FAQ.

```
public function testSomeLogic()

{

// Get the product from the fixture

$product = $this->getProduct();

// Get the ExecuteInStoreContext instance from the object manager

$executeInStoreContext =
$this->_objectManager->get(\Magento\TestFramework\Store\ExecuteInStoreContext::class);

// Execute the fixture in store view 3

$executeInStoreContext->executeInStoreContext(3, function () use ($product) {

// Do some operations on the product in store view 3

});

// Execute the tested logic in store view 4

$result = $executeInStoreContext->executeInStoreContext(4, function () use ($product) {

// Call the tested logic on the product in store view 4

return $this->someLogic->execute($product);

});

// Assert that the result is true

$this->assertTrue($result);

}
```

Integration tests | Magento 2 Developer Documentation

Data fixtures | Magento 2 Developer Documentation

Magento\TestFramework\Store\ExecuteInStoreContext | Magento 2 Developer Documentation

Question 2

Question Type: MultipleChoice

A client has multiple warehouses where orders can be fulfilled. The cost of shipping goods from each warehouse varies by day, due to the number of workers available. The Architect needs to make sure that when an order is shipped, it is shipped from the lowest cost warehouse that is open.

How should this functionality be implemented?

Options:

A- Create a new class as a preference for Magento\inventoryShipping\piugin\Sales\shipment\AssignSourceCodeToShipmentPlugin to set the lowest-cost warehouse on a shipment.

B- Create a new class implementing Magento\invtntorysourceSelectionApi\Modei\sourceSelectioninterfacece. which returns open warehouses sorted by cost.

C- Create an after plugin On Hagento\InventoryDistanceBasedSourceSelection\Hodel\Algorithms\DistanceBasedAlgorithto sort to Warehouse sources by cost

Answer:

B

Explanation:

According to the Adobe Commerce documentation, the Source Selection Interface is the main interface for implementing custom source selection algorithms. The interface defines a method called execute(), which takes a list of items to be shipped and a stock ID as parameters, and returns a SourceSelectionResultInterface object, which contains the recommended sources and quantities for each item. The Architect can create a new class that implements this interface and provides the logic for finding the lowest-cost warehouse that is open for each item. The Architect can then register the new class as an option for the source selection algorithm in the di.xml file of the custom module.

Source Selection Algorithm | Adobe Commerce Developer Guide

Source Selection Interface | Adobe Commerce Developer Guide

Question 3

Question Type: MultipleChoice

A merchant is utilizing an out-of-the-box Adobe Commerce application and asks to add a new reward card functionality for customers. During the code review, the Adobe Commerce Architect notices the reward_card_number attribute setup created for this functionality is causing the customer attribute to be unavailable in the My account/My rewards page template.

What should be added to set the customer attribute correctly?

Options:

A- group property should be added with a value of 1

B- system property should be added with a value of true

C- scope property should be added with a value of global

Answer:

A

Explanation:

The group property determines which section of the customer account the attribute belongs to. By setting the group property to 1, the reward_card_number attribute will be assigned to the default group, which is the Account Information section.This will make the attribute available in the My account/My rewards page template.Reference: https://experienceleague.adobe.com/docs/commerce-admin/customers/customer-accounts/attributes/attribute-properties.html?lang=en#group1

Question 4

Question Type: MultipleChoice

The development of an Adobe Commerce website is complete. The website is ready to be rolled out on the production environment.

An Architect designed the system to run in a distributed architecture made up of multiple backend webservers that process requests behind a Load Balancer.

After deploying the system and accessing the website for the first time, users cannot access the Customer Dashboard after logging in. The website keeps redirecting users to the sign-in page even though the users have successfully logged in The Architect determines that the session is not being

saved properly.

In the "app/etc/env.php", the session is configured as follows:

What should the Architect do to correct this issue?

Options:

A- Update the session host value to a shared Redis instance

B- increase the session size with the command config:set system/security/max_session_size_admin

C- Utilize the Remote Storage module to synchronize sessions between the servers

Answer:

A

Explanation:

Option A is correct because updating the session host value to a shared Redis instance in the ''app/etc/env.php'' file will allow the session to be saved properly and prevent users from being redirected to the sign-in page after logging in. Redis is a fast and reliable in-memory data store that can be used for session storage in Magento 2. By using a shared Redis instance, the session data can be accessed by any of the backend web servers behind the load balancer, regardless of which server handled the initial request.This ensures that the user's session is maintained and consistent across different servers1.

Option B is incorrect because increasing the session size with the command config:set system/security/max_session_size_admin will not solve the issue of session not being saved properly. This command only affects the admin session size limit, not the customer session size limit.Moreover, this command does not address the root cause of the issue, which is that the session data is not shared among the backend web servers2.

Option C is incorrect because utilizing the Remote Storage module to synchronize sessions between the servers is not a viable solution for this issue. The Remote Storage module is a feature of Magento Commerce Cloud that allows storing media files and other static content on a remote storage service such as AWS S3 or Azure Blob Storage.This module does not support synchronizing sessions between servers, as sessions are dynamic and transient data that need to be stored in a fast and accessible data store such as Redis3.

1: Use Redis for session storage | Adobe Commerce Developer Guide

2: Security | Adobe Commerce User Guide

3: Remote storage | Adobe Commerce Developer Guide

Question 5

Question Type: MultipleChoice

An Adobe Commerce Architect notices that the product price index takes too long to execute. The store is configured with multiple websites and dozens of customer groups.

Which two ways can the Architect shorten the full price index execution time? (Choose two.)

Options:

A- Set mage_ihdexer_threads_COUNT environment variable to enable parallel mode

B- Move catalog_Price_index indexer to another custom indexer group

C- Enable price index customer group merging for products without tier prices

D- Set Customer Share Customer Accounts Option to Global

E- Edit customer groups to exclude websites that they are not using

Answer:

A, C

Explanation:

The product price index can be optimized by using parallel mode and customer group merging. Parallel mode allows the indexer to run multiple threads simultaneously, which can speed up the indexing process. Customer group merging reduces the number of rows in the price index table by merging customer groups that have the same product prices. This can improve the performance of the price index queries and reduce the index size.Reference:Indexing optimization,Price index customer group merging