



## Prepare Smart for Success Free Adobe AD0-E716 Exam Questions and Answers

Ready to pass faster? Grab free and updated Adobe Commerce Developer with Cloud Add-on exam PDF questions now. Get authentic AD0-E716 dumps packed with verified answers and secure your certification success with PrepBolt AD0-E716 exam pdf questions and answers.

Thank you for Downloading AD0-E716 exam PDF Demo

<https://prepbolt.com/AD0-E716.html>

QUESTIONS & ANSWERS  
**DEMO VERSION**  
*(LIMITED CONTENT)*

# Question 1

---

Question Type: MultipleChoice

---

An Adobe Commerce developer was asked to provide additional information on a quote. When getting several quotes, the extension attributes are returned, however, when getting a single quote it fails to be returned.

What is one reason the extension attributes are missing?

## Options:

- A- The developer neglected to add `collection='true'` to their attribute in `etc/extension_attributes.xml` file. `<attribute code='my_attributes' type='MyVendor\MyModule\Api\Data\AttributeInterface' collection='true' />`
- B- The developer neglected to provide a plugin `On Magento\Quote\Api\CartRepositoryInterface::get`.
- C- The developer neglected to implement an observer on the `collection_load_after` event.

## Answer:

---

A

## Explanation:

---

The extension attributes are missing because the `collection='true'` attribute is not set in the `etc/extension_attributes.xml` file. This attribute tells Magento that the extension attributes should be returned when the quote is retrieved.

To fix this issue, the developer needs to add the `collection='true'` attribute to the `my_attributes` extension attribute.

Once the `collection='true'` attribute is set, the extension attributes will be returned when the quote is retrieved.

# Question 2

---

Question Type: MultipleChoice

---

An Adobe Commerce developer is asked to create a new payment method for their project. This

project has administrators who use the backend to manage customer information and occasionally place orders. When testing the new payment method on the frontend everything worked as expected, however, the payment method is missing in the admin.

What is a possible reason for this?

### Options:

---

- A- In the module di.xml, there were no default 3DS verification types configured as a VirtualType.
- B- In the module config.xmi, the boolean value for can\_capture was set to false.
- C- In the module config.xmi, the node can\_use\_internal was not set to true.

### Answer:

---

C

## Question 3

---

Question Type: MultipleChoice

---

An Adobe Commerce developer has been asked to modify the PageBuilder slider content type to allow a new custom content type (other than slide) to be assigned as a child. The developer has already created the new content type called improved\_slide in their module. They now need to create a new view/adminhtml/pagebuilder/content\_type/slider.xml file in their module to allow the new content type to be a child of slider content types.

What is the correct xml to accomplish this?

A)



B)



C)



### Options:

---

- A- Option A
- B- Option B
- C- Option C

## Answer:

---

B

## Explanation:

---

The following XML will allow the new content type to be a child of slider content types:

```
<type>slider</type>  
  
<children>  
  
<type>improved_slide</type>  
  
</children>
```

Use code with caution. <https://bard.google.com/faq>

This XML will tell Magento that the slider content type can have improved\_slide content types as children.

## Question 4

---

Question Type: MultipleChoice

---

An Adobe Commerce developer is creating a new console command to perform a complex task with a lot of potential terminal output. If an error occurs, they want to provide a message that has higher visibility than some of the other content that may be appearing, so they want to ensure it is highlighted in red (as seen in the screenshot):



How can they customize the appearance of this message?

## Options:

---

- A- Call the `setDecorationType(Stype)` method On the `Symfony\Console\Output\OutputInterface` Object before Calling `writeln()`.
- B- Wrap the output content in tags like `<error>`, `<info>`, or `<comment>`.
- C- Throw a new `commandException` with the desired message passed as an argument.

## Answer:

---

### Explanation:

---

The developer can customize the appearance of the error message by calling the `setDecorationType()` method on the `Symfony\Console\Output\OutputInterface` object before calling `writeln()`. The `setDecorationType()` method takes a single argument, which is the type of decoration that the developer wants to use. In this case, the developer wants to use the `STYPE_ERROR` decoration, which will highlight the message in red.

Here is an example of how to customize the appearance of the error message:

```
$output = new Symfony\Console\Output\ConsoleOutput();  
$output->setDecorationType(Symfony\Console\Output\OutputInterface::STYPE_ERROR);  
$output->writeln('This is an error message.');
```

The output of this code will be an error message that is highlighted in red.

## Question 5

---

Question Type: MultipleChoice

---

An Adobe Commerce developer wants to create a product EAV attribute programmatically which should appear as WYSIWYG in the admin panel. They have made sure that `wysiwyg_enabled` has been set to true, however, the attribute is not appearing as WYSIWYG in the admin panel.

What would be a possible reason?

### Options:

---

- A- The `is_html_allowed_on_front` Option is Set to false.
- B- The input type is not set to text.
- C- The input type is not set to textarea.

### Answer:

---

C

### Explanation:

---

The `input_type` attribute of a product EAV attribute specifies the type of input field that will be used to

enter the value of the attribute in the admin panel. The textarea input type is used for WYSIWYG fields. If the input\_type attribute is not set to textarea, then the attribute will not appear as WYSIWYG in the admin panel.

To fix this, the developer should set the input\_type attribute to textarea.

## Question 6

---

Question Type: MultipleChoice

---

A merchant is experiencing performance issues on integration environments of their Adobe Commerce Cloud Pro plan and wants to upgrade to Enhanced Integration Environments.

What are the steps necessary prior to redeploying in order to upgrade to Enhanced Integration Environments?

### Options:

---

- A- 1. Limit the number of Integration branches to two  
2. Submit a support ticket requesting the upgrade
- B- 1. Limit the number of Integration branches to three  
2. Set the ENV.ENVIRONMENT in .magento.env.yaml to ENHANCEDINTEGRATION
- C- 1. Limit the number of Integration branches to four  
2. Configure integration environments in the cloud GUI and set the Enhanced switch to On

### Answer:

---

A

### Explanation:

---

The steps necessary prior to redeploying in order to upgrade to Enhanced Integration Environments are to limit the number of integration branches to two and to submit a support ticket requesting the upgrade. Enhanced Integration Environments are an improved version of integration environments that offer better performance, stability, and security. They have a limit of four active branches at a time, but only two branches can be migrated from standard integration environments. The developer needs to delete or deactivate any extra branches before requesting the upgrade from Adobe support. Verified Reference: [Magento 2.4 DevDocs]

# Thank You for trying AD0-E716 PDF Demo

To try our AD0-E716 practice exam software visit  
link below

<https://prepbolt.com/AD0-E716.html>

## Start Your AD0-E716 Preparation

Use Coupon "SAVE50" for extra 50% discount on the purchase of  
Practice Test Software. Test your AD0-E716 preparation with actual  
exam questions.