

MAGENTO2-CERTIFIED-ASSOCIATE-DEVELOPER^{Q&As}

Magento 2 Certified Associate Developer

Pass Magento MAGENTO2-CERTIFIED-ASSOCIATE-DEVELOPER Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass2lead.com/magento-2-certified-associate-developer.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Magento
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



QUESTION 1

The constructor function for \Magento\Catalog\Model\Category contains this excerpt:

```
public function __construct(  
    // ...  
    \Magento\Store\Model\StoreManagerInterface $storeManager  
    // ...  
) { /* ... */ }
```

With the automatic dependency injection that Magento provides, how is the StoreManagerInterface resolved?

- A. If no \$storeManager is provided, Magento's code generator creates a shell concrete class based on \Magento\Store\Model\StoreManagerInterface
- B. Magento finds all classes that implement \Magento\Store\Model\StoreManagerInterface (ordered alphabetically) and injects the first class.
- C. Magento looks to the di.xml files in the entire system for a preference node for \Magento\Store\Model\StoreManagerInterface. This class is constructed and injected
- D. Magento throws an exception because you cannot instantiate an interface

Correct Answer: A

QUESTION 2

You are adding a child node to the product.info block using the XML:

```
<referenceBlock name="product.info">  
    <block class="MyCompany\MyModule\Block\Custom"  
        name="myblock"  
        as="mynewblock"  
        template="custom.phtml"/>  
</referenceBlock>
```

How will this block be rendered?

- A. Child block nodes are automatically rendered as HTML
- B. By calling \$block->getChildHtml('mynewblock') in the parent block's template
- C. The layout is invalid since block elements cannot be nested
- D. Automatically if the block class Custom implements the _toHtml method

Correct Answer: B

Reference: <https://magento.stackexchange.com/questions/49361/how-to-put-my-block-in-catalog-productview-page>

QUESTION 3

You have configured an event observer to watch the checkout_submit_all_after event using this XML:

```
<event name="checkout_submit_all_after">  
  <observer name="exampleObserver" instance="MyCompany\MyModule\Observer\CheckoutObserver" />  
</event>
```

What is the required class definition for the event observer?

- A.

```
class CheckoutObserver  
{  
  public function exampleObserver(Event\Observer $observer) { /* observer code */ }  
}
```
- B.

```
class CheckoutObserver implements Event\ObserverInterface  
{  
  public function execute(Event\Observer $observer) { /* observer code */ }  
}
```
- C.

```
class CheckoutObserver extends Event\ObserverAbstract  
{  
  public function execute(Event\Observer $observer) { /* observer code */ }  
}
```
- D.

```
class CheckoutObserver implements Event\ObserverInterface  
{  
  public function exampleObserver(Event\Observer $observer) { /* observer code */ }  
}
```

A. B. C. D.

Correct Answer: B

QUESTION 4

A merchant asks you to extend customer functionality to allow customer accounts to be associated with two or more billing addresses.

How is this implemented?

- A. By adding the attribute like customer_address_billing2 and customizing both My Account and Checkout functionality to use that new attribute
- B. By changing the System Configuration setting: Customer>Allow multiple billing addresses to Yes
- C. By altering the customer_entity table, adding the field billing_address2, and customizing both My Account and Checkout functionality to use that new field

D. This is out-of-the box functionality

Correct Answer: A

QUESTION 5

You are working on a new entity called vendor. You implemented the model, resource model and collection. You want to ensure that standard model events will be fired for your model, so an observer can be created for the events vendor_save_after, vendor_save_commit_after and others.

How do you do that?

- A. Create an entry in etc/di.xml and add the argument eventPrefix with the value of vendor
- B. Declare the \$_eventPrefix property in your vendor model and set it to vendor
- C. You must implement all appropriate methods and fire the events manually
- D. Ensure that the primary key in the corresponding table is named vendor_id

Correct Answer: B

Reference: <https://medium.com/@kangpaper/crud-models-in-magento-2-4e031861909b>

[MAGENTO2-CERTIFIED-ASSOCIATE-DEVELOPER VCE Dumps](#)

[MAGENTO2-CERTIFIED-ASSOCIATE-DEVELOPER Exam Questions](#)

[MAGENTO2-CERTIFIED-ASSOCIATE-DEVELOPER Braindumps](#)