Question 1

An Adobe Commerce developer wants to override the following Layout XML file in the theme ExampleCorp/orange.

app/design/frontend/ExampleCorp/blank/Vendor_Module/layout/catalog_product_view.xml

What path would the developer use inside the layout directory of the theme to override the file?

Options:

A- /override/ExampleCorp/blank/catalog_product_view.xml

B- /override/theme/ExampleCorp/blank/catalog_product_view.xml

C- /catalog_product_view.xml

Answer:

C

Explanation:

To override a layout XML file from a parent theme, the developer just needs to place the modified file in the same path relative to the layout directory of the child theme. In this case, the file would be app/design/frontend/ExampleCorp/orange/Vendor_Module/layout/catalog_product_view.xml. The override directory is not used for overriding layout files, but for overriding templates and web assets. Reference: [Layout instructions], [Override templates and layout files]

Question 2

Question Type: **MultipleChoice**

The merchant needs to create a new website, and is need modify a template the third party vendor's, because the customer is different. The file is found in a module here: app/code/Vendor/Module

Keep it simple in your mind!

Options:

A- Create another layout for the new website and configure new file.phtml.
app/code/Vendor/Module/view/frontend/templates/file.phtml

B- Create a new module for extends layout.xml and include new file.phtml.
app/code/Vendor/Module_Two/view/frontend/templates/file.phtml

C- Create a new theme, define a new website and customize in app/design.

app/design/frontend/Custom/Theme/Vendor_Module/templates/file.phtml

Answer:

C

Explanation:

The best way to customize a template file from a third-party module is to create a new theme that inherits from the parent theme and override the template file in the app/design/frontend/Custom/Theme/Vendor_Module/templates directory. This way, the customization is isolated from the original module and can be applied to a specific website or store view. Creating another layout file or a new module would not be as simple or flexible as creating a new theme. Reference:Frontend development guide, [Create a theme], [Theme inheritance]

Question 3

Question Type: MultipleChoice

An Adobe Commerce developer needs to improve the time of first render of the pages. How would the developer achieve this?

Options:

A- Use the quick static files deployment strategy

B- Enable CSS critical path

C- Enable CSS file merging

D- Enable JavaScript minification

Answer:

B

Explanation:

CSS critical path is a feature that improves the time of first render of the pages by inlining the CSS rules that are required to render the above-the-fold content of the page. This reduces the number of requests and bytes that need to be downloaded before the page is rendered. CSS critical path can be enabled in the Admin Panel by navigating to Stores > Configuration > Advanced > Developer > CSS Settings and setting Enable CSS Critical Path to Yes. Reference:Adobe Commerce Developer Documentation,Adobe Inc.

Question 4

Question Type: **MultipleChoice**

In which mode would the stylesheet customizations display immediately after you reload a page in a browser?

Options:

A- server- side compilation mode

B- less-side compilation mode

C- client-side compilation mode

Answer:

C

Explanation:

In client-side compilation mode, the stylesheet customizations will display immediately after reloading a page in a browser. This is because in this mode, the LESS files are compiled into CSS files by the browser using JavaScript. This mode is useful for development and debugging purposes, as it allows quick changes and previews of the styles. However, this mode is not recommended for production, as it can affect the performance and compatibility of the site. The other modes, server-side compilation and LESS compilation, require running commands or tools to compile the LESS files into CSS files on the server side, which can take some time and delay the display of the customizations. Reference: [Compilation modes], [Compile LESS]

Question 5

An Adobe Commerce developer has been asked to add text to an email template that supports translations. Which two options would they use during their implementation? (Choose two.)

Options:

A- {{translations 'Lorem Ipsum is simply dummy text of the printing'}}

B- {{translations '%items items' items='numItems'}}

C- {{trans 'Lorem Ipsum is simply dummy text of the printing'}}

D- {{trans '%items items' items='numItems'}}

Answer:

B, D

Explanation:

To add text to an email template that supports translations, the developer should use the {{trans}} directive with the text enclosed in double quotes. For example:

{{trans "Lorem Ipsum is simply dummy text of the printing"}}

This will render the text as it is, or translate it if a translation file is available for the current locale. If the text contains a variable, the developer should use a placeholder with a percent sign and pass the variable name as an argument. For example:

{{trans "%items items" items="numItems"}}

This will render the text with the value of numItems replacing the %items placeholder, or translate it if a translation file is available for the current locale. The {{translations}} directive is not valid and will not work. Reference: [Translate email templates], [Email template syntax]

Question 6

Question Type: **MultipleChoice**

An Adobe Commerce Developer is adding a new page layout to the theme directory within a custom theme. Which file needs to be created to register the new page layout?

Options:

A- app/design/frontend/<VendorName>/<ThemeName>/layouts. xml

B- app/design/frontend/<VendorName>/<ThemeName>/Magento_Theme/layouts.xnil

C- app/design/frontend/<VendorName>/<ThemeName>/Magento_Theme/layout/layouts.xml

Answer:

C

Explanation:

To register a new page layout in a custom theme, the developer needs to create a layouts.xml file in the app/design/frontend/<VendorName>/<ThemeName>/Magento_Theme/layout directory. The layouts.xml file should contain the <layout> element with the id, label, and file attributes. The id attribute is used to reference the layout in other layout files, the label attribute is used to display the layout name in the admin panel, and the file attribute is used to specify the path to the layout file relative to the web directory of the theme. The app/design/frontend/<VendorName>/<ThemeName>/layouts.xml and app/design/frontend/<VendorName>/<ThemeName>/Magento_Theme/layouts.xml files are not valid and will not work. Reference: [Create a new page layout], [layouts.xml]

Question 7

Question Type: MultipleChoice

An Adobe Commerce developer needs to add CMS content above products on a specific category page via the Admin Panel. Where would the developer need to put the content in order to use Display Modes functionality and display it on the category?

Options:

A- Widget

B- CMS Page

C- CMS Block

Answer:

C

Explanation:

CMS Blocks are the best option to add CMS content above products on a specific category page via the Admin Panel. CMS Blocks are reusable pieces of content that can be inserted into any page or layout using widgets or layout XML. CMS Blocks can be assigned to specific categories using the Display Settings tab in the category edit page. The developer can choose the Display Mode for the category, which determines how the CMS Block and the products are displayed on the category page. For example, the developer can choose Static block and products to show both the CMS Block and the products, or Static block only to show only the CMS Block and no products. Reference:Adobe Commerce Developer Documentation,Adobe Inc.

Question 8

An Adobe Commerce developer has been asked to move a block called country from the container sidebar to the container content, the block has to be the last child on the content container.

Which layout instruction would be used to move the block?

Options:

A- <move element = 'country' destination='content' after='-'/>

B- <move element='country-element' destination='content-element'/>

C- <move element='country' destination='content' after="last-child'/>

Answer:

A

Explanation:

To move a block from one container to another, the developer needs to use the <move> layout instruction with the element attribute specifying the name of the block and the destination attribute specifying the name of the container. The after attribute can be used to

position the block relative to other blocks in the same container. The value "-" means that the block will be placed after all other blocks, i.e., as the last child. The value "last-child" is not valid for the after attribute and will not work. The element and destination attributes should use the names of the blocks and containers, not their aliases or classes. Reference: [Layout instructions], [Move an element]

Question 9

An Adobe Commerce developer needs to debug an issue, where the path of the block template was invalid and the warning was added to a log file. Which mode are errors only written to a log file and not displayed?

Options:

A- developer only

B- default and production

C- developer and default

Answer:

B

Explanation:

The default and production modes are the modes where errors are only written to a log file and not displayed on the screen. This is done to prevent exposing sensitive information to users and attackers. The default mode is the mode that Adobe Commerce runs in by default if no other mode is specified. The production mode is the mode that Adobe Commerce runs in when it is deployed to a live site. The developer can use the following command to check the current mode:

bin/magento deploy:mode:show

The other two options are incorrect because they display errors on the screen as well as writing them to a log file. The developer mode is the mode that Adobe Commerce runs in when it is under development or testing. The developer mode enables enhanced debugging and error reporting features. Reference:Adobe Commerce Developer Documentation,Adobe Inc.

Question 10

Question Type: **MultipleChoice**

An Adobe Commerce developer is working on a custom knockout UI component and they need to add the text Happy Birthday. to be translated inside an .html template.

How would the developer add the text?

Options:

A-

B-

C- <!-- ko il8n = 'Happy Birthday.' --><!-- /ko -->

Answer:

B

Explanation:

To add the text Happy Birthday. to be translated inside an .html template, the developer should use the i18n binding. This binding allows the developer to specify the text as a string literal and translate it using the Magento translation mechanism. For example:

This will render the text as it is, or translate it if a translation file is available for the current locale. The i18n binding can also accept variables or expressions as arguments. For example:

This will render the text with the value of name variable, or translate it if a translation file is available for the current locale. The Mil8n and il8n bindings are not valid and will not work, as they are misspelled and do not match the knockout binding syntax. Reference: [Knockout bindings], [i18n binding]

Question 11

An Adobe Commerce developer has created a system configuration field:

```
<section id="module" type="text" sortOrder="10" showInDefault="1" showInWebsite="1" showInStore="1">
    <label>Vendor Module</label>
    <tab>vendor</tab>
    <resource>Vendor_Module::store_config</resource>
    <group id="general" type="text" sortOrder="0" showInDefault="1" showInWebsite="1" showInStore="1">
        <label>General</label>
        <field id="enable" translate="label" type="select" sortOrder="0" showInDefault="1" showInStore="1"
            <label>Enable</label>
            <source_model>Magento\Config\Model\Config\Source\Yesno</source_model>
        </field>
    </group>
</section>
```

Using Layout XML, how can the visibility of a block be controlled by a system configuration?

Options:

A- <block name='block.name' template='Vendor_Module::template.phtml=" ifconfig='vendor/general/enable' />

B- <block name='block.name' template="Vendor_Module: :template.phtml"ifconfig='module/general/enable' />

C- <block name='block.name' template='Vendor_Module::template.phtml' ifconfig='general/module/enable' />

Answer:

A

Explanation:

To control the visibility of a block using a system configuration, the developer should use the ifconfig attribute in the <block> tag. The ifconfig attribute should specify the path to the system configuration field that determines whether the block is visible or not. For example:

<block name="block.name" template="Vendor_Module::template.phtml" ifconfig="vendor/general/enable" />

This will render the block only if the vendor/general/enable system configuration field is set to true. The module/general/enable and general/module/enable paths are not valid and will not work, as they do not match the system configuration field defined in the image. Reference: [Layout instructions], [System configuration]

Question 12

Question Type: **MultipleChoice**

an Adobe commerce developer wants to override the core Magento UI library dropdowns in your theme. Which is the correct way to achieve this?