

Sitecore.Sitecore-XM-Cloud-Developer.v2024-08-03.q17

Exam Code:	Sitecore-XM-Cloud-Developer
Exam Name:	Sitecore XM Cloud Developer Certification Exam
Certification Provider:	Sitecore
Free Question Number:	17
Version:	v2024-08-03
# of views:	291
# of Questions views:	170
https://www.exam-tests.com/Sitecore-XM-Cloud-Developer-exam/Sitecore.Sitecore-XM-Cloud-Developer.v2024-08-03.q17.html	

NEW QUESTION: 1

A developer is creating a component that has the following requirements:

*Image Left, Title, Call to action

*Image Right, Title, Call to action

*Image Centered, Call to action

The component should be able to display the same content in all three options. What is the best way to build this?

- A. Three separate components with the compatible renderings set
- B. One component with logic in the component to change the view
- C. One component with three Headless variants
- D. One component with rendering parameters to control the layout

Answer: D (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers¹, rendering parameters are a way to define custom properties for a component that can be changed by the content author in the XM Cloud Pages editor. Rendering parameters allow you to create flexible and reusable components that can adapt to different scenarios and layouts. To create a component with rendering parameters, you need to follow these steps²:

Create a data template that defines the rendering parameters for your component. For example, you can create a data template with a field called Layout that has three options: Left, Right, and Center.

Create a component that uses the data template as its rendering parameter template. For example, you can create a component called Image and Text that has an image, a title, and a call to action button.

In the component's code, use the rendering parameter value to control the layout of the component. For example, you can use conditional logic or CSS classes to change the position of the image based on the Layout value.

Register the component in the Components builder and add it to the Components library. In the XM Cloud Pages editor, drag and drop the component to the page and use the Properties panel to change the rendering parameter value. For example, you can select Left, Right, or Center from the Layout drop-down menu.

2:Create a component with rendering parameters | Sitecore Documentation1:XM Cloud Documentation for Developers - Sitecore

NEW QUESTION: 2

When a developer selects the base templates for a new data template, what happens if those base templates contain a field with the same name?

- A. The field will not be displayed on the data item.
- B. Duplicate field names will appear on the data item.
- C. XM Cloud will add a suffix to the field names to identify them.
- D. The fields will be merged on the data item.

Answer: D (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers¹, a template inherits sections and fields from its base templates. When there are several base templates associated with a data template, fields and sections from inherited templates are merged. To configure the base templates for a data template, you can use the Base Templates dialog box in the Template Manager or the Content Editor¹.

NEW QUESTION: 3

A developer needs to create a site for a company and must define the data structures in Sitecore to create items and content. What should the developer use to define the data structures?

- A. Templates
- B. Rendering parameters
- C. Renderings
- D. Components data sources

Answer: A (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers¹, templates are the fundamental concept in Sitecore that determine the structure and behavior of content items. Templates define the fields, sections, and parameters that make up each item. Templates also enable inheritance, which allows items to share common fields and properties from their base templates. There are different types of templates in Sitecore, such as data templates, branch templates, and command templates².

To create the data structures for a site, the developer should use data templates, which define the fields used to control how data is entered. Data templates form the framework

around which items are built. Sitecore associates a data template with every item in the content tree. The field types determine the editing controls shown to the user and they contain the raw values that are stored in the database².

The other options are not correct:

Rendering parameters are custom properties for a component that can be changed by the content author in the XM Cloud Pages editor. Rendering parameters allow the content author to customize the appearance and behavior of the component, but they do not define the data structure of the content item³.

Renderings are items that define the metadata and parameters for a component.

Renderings link a component to a data template, a headless variant, and a rendering parameter template. Renderings are used to create and manage components in the Components builder, but they do not define the data structure of the content item⁴.

Component data sources are content items that provide the content for a component.

Component data sources can have fields, such as text, image, or link, that can be edited by the content author in the XM Cloud Pages editor. Component data sources are based on data templates, but they are not templates themselves⁴.

²:Data definition and template overview | Sitecore Documentation³:Create a component with rendering parameters | Sitecore Documentation⁴:Create a component that uses a data source item | Sitecore Documentation

¹:XM Cloud Documentation for Developers - Sitecore

NEW QUESTION: 4

A developer changed the CSS grid system after a site was created. Which of the following can happen due to references to the previous grid system?

- A.** The changes break the layout of the site.
- B.** The Grid Setup item and its dependencies were automatically updated.
- C.** A new custom theme with a grid definition item was created.
- D.** The new grid was automatically added as one of the grid options.

Answer: ([SHOW ANSWER](#))

Changing the CSS grid system after a site has been created can lead to broken layouts due to references to the previous grid system. This is because grid systems have specific rules, classes, and options that are not automatically updated or compatible with the new system, requiring manual changes to ensure the layout functions correctly.

References: The Sitecore XM Cloud documentation provides insights into the implications of changing the grid system, emphasizing the need for manual updates to avoid breaking the site layout¹.

NEW QUESTION: 5

A developer has set up a local XM Cloud development environment and would like to connect Sitecore Pages to their locally running XM Cloud instance. However, after

following the necessary steps, they're facing issues with the connection. Of the options below, what is the most likely potential reason for this issue?

- A. The browser developer tools console does not support the local storage functionality required for this connection.
- B. The value of the local storage entry "Sitecore.Pages.LocalXmlCloudUrl" should be set to <https://pages.sitecorecloud.io>.
- C. The value of the local storage entry "Sitecore.PagesXmlCloud" should be set to enabled.
- D. The environment variable "SITECORE_Pages_CORS_Allowed_Origins" is not correctly configured to include pages.sitecorecloud.io.

Answer: D (LEAVE A REPLY)

NEW QUESTION: 6

A developer needs to configure a rendering in order to use dynamic placeholders. Which of the following steps is required? Select all that apply.

- A. Include the IDynamicPlaceholder base template in the Rendering Parameters template.
- B. Link the placeholder settings item to the rendering item.
- C. Define the placeholder key using a question mark (?) in the placeholder settings item.
- D. In the component TSX file, set a unique placeholder key value that has not yet been defined.

Answer: A,B,C (LEAVE A REPLY)

To configure a rendering for the use of dynamic placeholders in Sitecore XM Cloud, the following steps are required:

Include the IDynamicPlaceholder base template in the Rendering Parameters template to enable dynamic assignment of IDs to the placeholder key.

Link the placeholder settings item to the rendering item to ensure that the dynamic placeholders are correctly associated with the rendering.

Define the placeholder key using a wildcard in the placeholder settings item, which allows for the generation of unique placeholder keys for every component on the page.

References: The Sitecore XM Cloud documentation provides a walkthrough on configuring components to use dynamic placeholders, detailing the steps required to set up dynamic placeholders¹. This includes adding the `IsRenderingsWithDynamicPlaceholders` property to the component item, including the `IDynamicPlaceholder` base template in the Rendering Parameters template, defining the placeholder key using a wildcard, and linking the placeholder settings item to the component¹.

NEW QUESTION: 7

What is the `sitecore.json` file in a development solution?

- A. The environment variables file for the Sitecore Docker deployment
- B. The configuration file for the Sitecore development tooling
- C. The solution integration file needed for deploying code changes to the environment
- D. A custom configuration file defining a solution for headless environments

Answer: C (LEAVE A REPLY)

The `sitecore.json` file is located in the root folder of the solution and it comes with the XM Cloud Foundation Head starter kit¹. In here we can configure what `*.module.json` files should be considered for serialization configuration, the plugins with its particular versions and some general settings for the serialization¹.

References:

Setup Sitecore Content Serialization

Sitecore Content Serialization structural overview

NEW QUESTION: 8

A developer is working with Sitecore's Authoring and Management API to manage their Sitecore content using GraphQL. They want to explore and interact with the API using the GraphQL integrated development environment (IDE). Which of the following statements is correct about using the GraphQL IDE?

- A. A developer needs to be in the `sitecore\Admin` role to access the GraphQL IDE.
- B. A developer needs to be in the `sitecore\Sitecore Client Users` role to access the GraphQL IDE.
- C. The GraphQL IDE provides read-only access to the API.
- D. The GraphQL IDE is only available for non-production environments to ensure secure interactions.

Answer: B (LEAVE A REPLY)

Access to the GraphQL IDE for exploring and managing Sitecore content via the Authoring and Management API requires a developer to have the `sitecore\Sitecore Client Users` role. This role grants the necessary permissions to use the IDE for various operations, not just read-only access.

References: The Sitecore XM Cloud documentation specifies the role requirements for using the GraphQL IDE¹. It also provides guidance on setting up and authoring content with the GraphQL playgrounds, which are part of the IDE².

NEW QUESTION: 9

A developer needs to log in to the XM Cloud Deploy app to create a project using a starter template but requires the correct permissions. Which of the following should happen?

- A. An Organization Admin or Organization Owner must update their role.
- B. An Organization Admin or Organization Owner must update their role using the XM Cloud Deploy app.
- C. Only the Organization Owner can update their role in the Sitecore Cloud Portal.
- D. Only the Organization Owner can update their role using the XM Cloud Deploy app.

Answer: A (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers¹, to use the XM Cloud Deploy app, you must be an Organization Admin or Organization Owner in your Sitecore Cloud Portal organization. An Organization Admin or Organization Owner can grant a team

member access to the XM Cloud Deploy app by changing their organization role to Admin2. This can be done in the Sitecore Cloud Portal, not in the XM Cloud Deploy app.

2:Invite team members to your Sitecore Cloud Portal organization1:XM Cloud

Documentation for Developers

- Sitecore

NEW QUESTION: 10

A developer is tasked with creating an item using the Sitecore Authoring and Management GraphQL API.

Which of the following GraphQL mutations is the correct way to create a new item?

- A. createOrUpdateItem
- B. createTemplateItem
- C. createItem
- D. updateItem

Answer: (SHOW ANSWER)

The correct GraphQL mutation to create a new item in Sitecore XM Cloud is createItem.

This mutation allows developers to specify the necessary details such as the item's name, template ID, parent ID, language, and fields to create a new content item within the Sitecore content tree.

References:The usage of the createItem mutation is documented in the Sitecore XM Cloud Developer's Guide, which provides examples and explanations for authoring operations, including item creation1. Additionally, the Sitecore Stack Exchange provides insights into the available mutations for item management, confirming the use of createItem for creating new items2.

NEW QUESTION: 11

Why should a developer assign security to roles instead of users when setting up security on a site?

- A. Security roles can assign individual rights and permissions to be configured by users.
- B. Role-based security allows multiple users to have the same username.
- C. Security can be managed more efficiently as the rights and permissions apply to a group of users.
- D. Individual users with a role can set and configure their own item security permissions.

Answer: C (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers1, security roles define how different users access different types of records in Sitecore. To control access to data and resources, you can create or modify security roles and change the security roles that are assigned to your users. A user can have multiple security roles. Security role privileges are cumulative. Users are granted the privileges that are available in each role that's assigned to them2.

This makes it easier for you to manage your security system because you do not necessarily have to assign access rights for each item in your content tree. Instead, you only need to assign and manage the access rights on the parent items and then specify whether their descendants can inherit these access rights³.

NEW QUESTION: 12

Which of the following is part of the out-of-the-box technology stack for XM Cloud?

- A. jQuery
- B. Python
- C. Svelte Kit
- D. React

Answer: D (LEAVE A REPLY)

XM Cloud includes a modern technology stack that supports various frameworks and libraries for development. React is part of this stack, as it is commonly used in conjunction with Sitecore's headless services and JSS (JavaScript Services) for building dynamic user interfaces.

References: The Sitecore XM Cloud documentation mentions the use of modern frameworks and libraries, including React, as part of its technology stack¹. This aligns with the industry's move towards more flexible and modular front-end development practices.

NEW QUESTION: 13

When a developer deploys to XM Cloud, which resources are included in the provisioning step?

- A. Content Management instance, Editing Host, and Experience Edge
- B. Content Management instance and Experience Edge
- C. Content Management instance and Vercel application
- D. Content Management instance and Content Delivery instance

Answer: A (LEAVE A REPLY)

During the provisioning step of a deployment to XM Cloud, the resources that are included are the Content Management instance, Editing Host, and Experience Edge. These components are essential for managing content, providing an editing interface, and delivering content to end-users, respectively.

References: The Sitecore XM Cloud documentation details the deployment process and the resources that are provisioned during deployment, which includes the Content Management instance, Editing Host, and Experience Edge¹.

NEW QUESTION: 14

How can a developer differentiate between XM Cloud Components and Headless SXA Components?

- A. XM Cloud Components offer drag-and-drop construction of pages. Headless SXA Components require manual coding and can only be added to a page by a developer.

B. XM Cloud Components are a low/no code builder for creating components that can consume data from multiple data sources. Headless SXA components require code that consumes data from the same data sources.

C. In XM Cloud Components, the Styles library ensures visual consistency across components. Headless SXA Components emphasize dynamic content mapping.

D. XM Cloud Components provide responsive designs for different device types. Headless SXA Components are exclusively focused on static layouts.

Answer: B (LEAVE A REPLY)

XM Cloud Components provide a low-code or no-code solution for building components, allowing for the consumption of data from various sources without extensive coding. On the other hand, Headless SXA Components are designed for developers to code and consume data, typically from the same data sources, offering a more hands-on approach to component creation.

References: The distinction between XM Cloud Components and Headless SXA Components is outlined in the Sitecore XM Cloud documentation, which describes the capabilities and intended use cases for each type of component¹².

NEW QUESTION: 15

A multinational clothing company wants to create a structured online presence for its various brands and countries. They want to ensure that the different brands and countries have their own dedicated content within the same Sitecore instance. Which approach should they take to achieve this using XM Cloud's content architecture?

A. Use separate instances of XM Cloud for each brand and country to maintain separation.

B. Create a single site collection and site for all brands and countries and also manage their resources manually.

C. Create individual site collections for each brand and a site for each country.

D. Create a separate site for each brand and use a language per country.

Answer: C (LEAVE A REPLY)

XM Cloud's content architecture supports the creation of individual site collections for each brand, with separate sites for each country. This approach allows a multinational company to maintain a structured online presence, ensuring dedicated content for different brands and countries within the same Sitecore instance.

References: The Sitecore XM Cloud documentation outlines the ability to create headless site collections and sites, supporting multiple brands for a single company or multiple languages or locations for a single brand¹. This is part of the SXA content architecture, which includes site collections and sites, allowing for multiple sites on a single instance of Sitecore¹.

NEW QUESTION: 16

Which of the following tasks are performed by the publishing pipeline to create an accurate snapshot of the content available for publishing from the Content Management instance?
Select all that apply.

- A. Calculating additional entities to publish
- B. Calculating dependencies
- C. Archiving expired content
- D. Resolving dependencies

Answer: A,B,D (LEAVE A REPLY)

The publishing pipeline in Sitecore XM Cloud performs several tasks to ensure an accurate snapshot of content is available for publishing from the Content Management instance. These tasks include calculating additional entities to publish, calculating dependencies, and resolving dependencies. These steps are crucial for maintaining the integrity and consistency of the published content.

References: The Sitecore XM Cloud documentation outlines the publishing pipeline process, which includes the calculation and resolution of entities and dependencies as part of creating a publishing manifest¹.

Valid Sitecore-XM-Cloud-Developer Dumps shared by BraindumpsPass.com for Helping Passing Sitecore-XM-Cloud-Developer Exam! BraindumpsPass.com now offer the **newest Sitecore-XM-Cloud-Developer exam dumps**, the BraindumpsPass.com Sitecore-XM-Cloud-Developer exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Sitecore-XM-Cloud-Developer dumps with Test Engine here:

<https://www.braindumpsPass.com/Sitecore/Sitecore-XM-Cloud-Developer-practice-exam-dumps.html> (52 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 17

If a developer wants to limit the serialization of items under a Navigation item to just the item itself and then one step below, what property should the developer add to the includes to indicate this limitation?

- A. Scope
- B. Nothing-this is the default.
- C. Limit
- D. Name

Answer: A (LEAVE A REPLY)

According to the Sitecore XM Cloud Documentation for Developers¹, the scope property is used to control the depth of the serialization tree. The scope property can have one of the following values²:

Self- only the item itself is serialized, not its descendants.

Children- the item and its immediate children are serialized, not the grandchildren or further descendants.

Subtree- the item and all its descendants are serialized, regardless of the depth.

To limit the serialization of items under a Navigation item to just the item itself and then one step below, the developer should add the scope property with the value Children to the includes section of the serialization configuration file. For example:

includes:

-name:Navigation

path:/sitecore/content/MySite/Navigation

scope:Children

2:The YAML serialization format | Sitecore Documentation3:The XM Cloud build configuration | Sitecore Documentation1:XM Cloud Documentation for Developers - Sitecore

Valid Sitecore-XM-Cloud-Developer Dumps shared by BraindumpsPass.com for Helping Passing Sitecore-XM-Cloud-Developer Exam! BraindumpsPass.com now offer the **newest Sitecore-XM-Cloud-Developer exam dumps**, the BraindumpsPass.com Sitecore-XM-Cloud-Developer exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Sitecore-XM-Cloud-Developer dumps with Test Engine here:

<https://www.braindumpspass.com/Sitecore/Sitecore-XM-Cloud-Developer-practice-exam-dumps.html> (52 Q&As Dumps, **40%OFF** Special Discount: **Exam-Tests**)