



AD0-E134^{Q&As}

Adobe Experience Manager Developer Exam

Pass Adobe AD0-E134 Exam with 100% Guarantee

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

<https://www.pass4itsure.com/ad0-e134.html>

100% Passing Guarantee
100% Money Back Assurance

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

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





QUESTION 1

An AEM as a Cloud Service implementation customer wants content to be replicated as soon as the On Time and Off Times are reached.

What must the developer configure?

- A. Enable Auto Replicate via On Off Trigger Configuration
- B. Enable On Time or Off Time via Page Properties
- C. Configure the Publish content tree via Workflow Models

Correct Answer: A

Explanation: The Auto Replicate via On Off Trigger Configuration is a feature that allows content to be replicated as soon as the On Time and Off Times are reached. This feature can be enabled by creating an OSGi configuration for the

com.day.cq.wcm.core.impl.servlets.OnOffTimeServlet service. This service listens for changes in the On Time and Off Time properties of pages and triggers replication accordingly.

References: <https://experienceleague.adobe.com/docs/experience-manager-65/deploying/configuring/replication.html?lang=en#on-off-time>
<https://experienceleague.adobe.com/docs/experience-manager-65/deploying/configuring/replication.html?lang=en#auto-replication>

replication.html?lang=en#auto-replication

QUESTION 2

Refer to the exhibit.

```
<div class="list">
<ul data-sly-
list="currentPage.listChildren" >< li > This is a sample HTML snippet < /li >< /ul >< /div >< div class = "r
{currentPage.listChildren}">
<li>This is a sample HTML snippet</li>
</ul>
</div>
```

The current page has three children.

What is the final rendered html output for the code snippet?



```
<div class='list'>
<ul>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
</ul>
</div>
A. <div class='repeat'>
<ul>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
</ul>
</div>

<div class='list'>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
</div>
B. <div class='repeat'>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
</div>

<div class='list'>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
</div>
C. </ul>
</div>
<div class='repeat'>
<ul>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
</ul>
</div>

<div class="list">
<ul>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
<li>This is a sample HTL snippet</li>
</ul>
</div>
<div class="repeat">
<ul>
D. <li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
<ul>
<li>This is a sample HTL snippet</li>
</ul>
</div>
```



A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: D

QUESTION 3

A developer needs to create sling models for two fields name and occupations. The dialog has two fields, name - a single value field, and occupations - a multi value field. The following code is included in sling models inherited from interface com.adobe.aem.guides.wknd.core.models.Byline



```

package com.adobe.aem.guides.wknd.core.models.impl;
.....
public class BylineImpl implements Byline {
    .....
    @Override
    public List<String> getOccupations() {
        if (occupations != null) {
            Collections.sort(occupations);
            return new ArrayList<String>(occupations);
        } else {
            return Collections.emptyList();
        }
    }
    .....
}

```

- Ⓐ
- ```

<div data-sly-use.byline="com.adobe.aem.guides.wknd.core.models.Byline"
 data-sly-use.placeholderTemplate="core/wcm/components/commons/v1/templates.html"
 data-sly-test.hasContent="${!byline.empty}"
 class="cmp-byline">

 <h2 class="cmp-byline__name">${byline.name}</h2>
 <p class="cmp-byline__occupations">${byline.occupations @ join=', '}</p>
</div>

```
- Ⓑ
- ```

<div data-sly-use.byline="com.adobe.aem.guides.wknd.core.models.Byline.impl"
    data-sly-use.placeholderTemplate="core/wcm/components/commons/v1/templates.html"
    data-sly-test.hasContent="${!byline.empty}"
    class="cmp-byline">

    <h2 class="cmp-byline__name">${byline.name}</h2>
    <p class="cmp-byline__occupations">${byline.occupations @ join=', '}</p>
</div>

```
- Ⓒ
- ```

<div data-sly-use.byline="com.adobe.aem.guides.wknd.core.models.Byline"
 data-sly-use.placeholderTemplate="core/wcm/components/commons/v1/templates.html"
 data-sly-test.hasContent="${!byline.empty}"
 class="cmp-byline">

 <h2 class="cmp-byline__name">${byline.name}</h2>
 <p class="cmp-byline__occupations">${byline.occupations }</p>
</div>

```
- Ⓓ
- ```

<div data-sly-use.byline="com.adobe.aem.guides.wknd.core.models.Byline"
    data-sly-use.placeholderTemplate="core/wcm/components/commons/v1/templates.html"
    data-sly-test.hasContent="${!byline.empty}"
    class="cmp-byline">

    <h2 class="cmp-byline__name">${byline.name @ join=', '}</h2>
    <p class="cmp-byline__occupations">${byline.occupations @ join=', '}</p>
</div>

```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

Explanation: Option C is the correct implementation for the Sling Model. Option C uses the @Model annotation with the adaptables parameter set to Resource.class. This allows the Sling Model to adapt from a resource object and access its properties using the ValueMap interface. Option C also uses the @Inject annotation with the name parameter set to "/name" and "/occupations" to inject the values of the name and occupations properties into the name and occupations fields. Option C also uses the @Named annotation with the value parameter set to "byline" to specify the name of the Sling Model that can be used in HTL scripts. References:



<https://sling.apache.org/documentation/bundles/models.html><https://experienceleague.adobe.com/docs/experience-manager-htl/using-htl/htl-block-statements.html?lang=en#use>

QUESTION 4

In a non-optimized website, the final HTML generated for a typical page by publish instance includes a relatively large number of elements that refer to other script files loaded from AEM environment. The developer wants to minimize these network calls by combining all required client library code into a single file to reduce the number of back-and-forth requests on page load.

Which step should a developer take to solve this issue?

- A. Embed the required libraries into an app-specific client library using the allowProxy property of the cq:ClientLibraryFolder node
- B. Add the categories property of the cq:ClientLibraryFolder node into an app-specific client library folder
- C. Embed the required libraries into an app-specific client library using the dependencies property of the cq:ClientLibraryFolder node
- D. Embed the required libraries into an app-specific client library using the embed property of the cq:ClientLibraryFolder node

Correct Answer: C

Explanation: The embed property of the cq:ClientLibraryFolder node allows embedding code from a client library into another client library. At runtime, the generated JS and CSS files of the embedding library include the code of the embedded library. This reduces the number of network calls and improves performance. Embedding code is useful for providing access to libraries that are stored in secured areas of the repository. References:
<https://experienceleague.adobe.com/docs/experience-manager-cloud-service/content/implementing/developing/fullstack/clientlibs.html?lang=en#embed><https://experienceleaguecommunities.adobe.com/t5/a-dobe-experience-manager/embed-property-in-client-libs/m-p/426858>

QUESTION 5

Which tool should a developer use to look up Adobe Identity Management System (IMS) users by email and return their IMS IDs?

- A. Developer Console
- B. User Mapping Tool
- C. IMS Lookup Tool
- D. Cloud Acceleration

Correct Answer: C

Explanation: The IMS Lookup Tool is a tool that allows looking up Adobe Identity Management System (IMS) users by email and returning their IMS IDs. The IMS Lookup Tool is available in the Cloud Manager UI under the Tools menu. The

IMS Lookup Tool can be used to find the IMS IDs of users who need to be added to Cloud Manager programs or



environments.

References: <https://experienceleague.adobe.com/docs/experience-manager-cloud-service/implementing/using-cloud-manager/ims-lookup-tool.html?lang=en>
[https://experienceleague.adobe.com/docs/experience-manager-cloud-service/](https://experienceleague.adobe.com/docs/experience-manager-cloud-service/implementing/using-cloud-manager/adding-users-to-programs-and-environments.html?lang=en)
[implementing/using-cloud-manager/adding-users-to-programs-and-environments.html?lang=en](https://experienceleague.adobe.com/docs/experience-manager-cloud-service/implementing/using-cloud-manager/adding-users-to-programs-and-environments.html?lang=en)

[AD0-E134 Study Guide](#)

[AD0-E134 Exam Questions](#)

[AD0-E134 Braindumps](#)