



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**

A custom component has one dialog field:

-> Title

-fieldLabel = Title

-sling:resourceType = granite/ui/components/coral/foundation/form/textfield

-name = ./title

The developer needs to implement a Sling Model to perform a business logic on the authored value. The developer writes the following HTL snippet.

```
<sly data-sly-use.display="com.adobe.aem.guides.certification.core.models.HelloWorldModelImpl">
  <h1>${display.messageText}</h1>
</sly>
```

Which two implementations will support this HTL snippet? (Choose two.)

☐ A. `@Model(adaptables = Resource.class, defaultInjectionStrategy = DefaultInjectionStrategy.OPTIONAL)`
`public class HelloWorldModelImpl {`
 `@ScriptVariable`
 `private String authoredVal;`
 `private String messageText;`

```
    @PostConstruct
    public void init() {
        if (StringUtils.isNotBlank(authoredVal)) {
            setMessageText(StringUtils.join("Welcome", StringUtils.SPACE, authoredVal));
        }
    }

    public void setMessageText(String messageText) {
        this.messageText = messageText;
    }

    public String getMessageText() {
        return messageText;
    }
}
```



☐ B.

```
public void init() {
    if (StringUtils.isNotBlank(title)) {
        setMessageText(StringUtils.join("Welcome", StringUtils.SPACE, title));
    }
}

public void setMessageText(String messageText) {
    this.messageText = messageText;
}

public String getMessageText() {
    return messageText;
}

}

@Model(adaptables = SlingHttpServletRequest.class, defaultInjectionStrategy = DefaultInjectionStrategy.OPTIONAL)
public class HelloWorldModelImpl {
    @Inject
    @Via("resource")
    private String title;
    private String messageText;
```

☐ C.

```
@PostConstruct
public void init() {
    if (StringUtils.isNotBlank(title)) {
        setMessageText(StringUtils.join("Welcome", StringUtils.SPACE, title));
    }
}

public void setMessageText(String messageText) {
    this.messageText = messageText;
}

public String getMessageText() {
}

}

@Model(adaptables = Resource.class, defaultInjectionStrategy = DefaultInjectionStrategy.OPTIONAL)
public class HelloWorldModelImpl {
    @ValueMapValue
    @Named("title")
    private String authoredVal;
    private String messageText;
```

☐ D.

```
@PostConstruct
public void init() {
    if (StringUtils.isNotBlank(title)) {
        setMessageText(StringUtils.join("Welcome", StringUtils.SPACE, title));
    }
}

public void setMessageText(String messageText) {
    this.messageText = messageText;
}

public String getMessageText() {
    return messageText;
}

}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: BD



Explanation: Option B and Option D are two implementations that will support the HTL snippet. Option B 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 B also uses the @Inject annotation with the name parameter set to ".text" to inject the value of the text property into the text field. Option D uses the @Model annotation with the defaultInjectionStrategy parameter set to OPTIONAL. This allows the Sling Model to use optional injection for all fields and avoid null pointer exceptions if a property is missing. Option D also uses the @Inject annotation without any parameters to inject the value of the text property into the text field, using the field name as the default property name. 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 2

A developer needs to create a dynamic participant step where the participant is selected automatically at run time.

The developer decides to develop an OSGi service, which needs to implement the `com.day.cq.workflow.exec.ParticipantStepChooser` interface.

Which method should the developer implement from the `com.day.cq.workflow.exec.ParticipantStepChooser` interface?

- A. `String getParticipant(WorkItem workItem, WorkflowSession workflowSession, MetaDataMap metaDataMap)`
- B. `void getParticipant(WorkItem workItem, WorkflowSession workflowSession, MetaDataMap metaDataMap)`
- C. `String getDynamicParticipant(WorkItem workItem, WorkflowSession workflowSession, MetaDataMap metaDataMap)`
- D. `void getDynamicParticipant(WorkItem workItem, WorkflowSession workflowSession, MetaDataMap metaDataMap)`

Correct Answer: A

<https://medium.com/@toimrank/aem-dynamic-participant-step-1cd58e15f28>

QUESTION 3

If multiple configurations for the same PID are applicable, which configuration is applied?

- A. The last modified configuration is applied.
- B. The configuration with the highest number of matching run modes is applied.
- C. The one that occurs first in the repository is applied.
- D. A configuration factory is created and all configurations are applied.

Correct Answer: B

When multiple configurations for the same PID are applicable, the configuration with the highest number of matching runmodes is applied. This is because the runmodes act as a filter to select the most specific configuration for a given environment. If there is a tie between two or more configurations with the same number of matching runmodes, the one that occurs first in the repository is applied. References:<https://experienceleague.adobe.com/docs/experience-manager-65/deploying/configuring/configure-runmodes.html?lang=en#configuring-osgi-settings-per-runmode>

**QUESTION 4**

An AEM application requires LDAP Service integration to synchronize users/groups. Which two OSGi configuration are required for LDAP integration in AEM? (Select Two.)

- A. Apache Jackrabbit Oak AuthorizableActionProvider
- B. Apache Jackrabbit Oak Solr server provider
- C. Apache Jackrabbit Oak CUG Configuration
- D. Apache Jackrabbit Oak External Login Module
- E. Apache Jackrabbit Oak Default Sync Handler

Correct Answer: DE

Explanation: The Apache Jackrabbit Oak External Login Module and Apache Jackrabbit Oak Default Sync Handler are the two OSGi configurations that are required for LDAP integration in AEM. The External Login Module defines how AEM connects to the LDAP server and authenticates users against it. The Default Sync Handler defines how AEM synchronizes users and groups from the LDAP server to the repository.

References: <https://experienceleague.adobe.com/docs/experience-manager-65/administering/security/ldap-config.html?lang=en#ldap-integration>

QUESTION 5

AEM SPA integration provides various design models. In an application the developer chooses to use AEM as a headless CMS without using the SPA Editor SDK framework.

What would be an advantage for this design model?

- A. The content author can edit the app using AEM's content authoring experience.
- B. The developer has full control over the app.
- C. The SPA is compatible with the template editor
- D. The developer keeps control over the app by only enabling authoring in restricted areas of the app

Correct Answer: C

In an application where the developer chooses to use AEM as a headless CMS without using the SPA Editor SDK framework, the advantage would be that the developer keeps control over the app by only enabling authoring in restricted areas of the app. By limiting the use of the SPA Editor SDK, the developer can ensure that content authors do not have access to the entire application code and can only make changes within specific areas or components that have been designated for them. This provides greater security and control over the application while still allowing content authors to make necessary updates and changes to the content. Therefore, the correct answer is D.