

Free PDF ACD201 - Appian Senior Developer Newest Actualtest



Are you preparing for the Appian certification recently? Maybe the training material at your hands is wearisome and dull for you to study. Here Lead2PassExam will give you a very intelligence and interactive ACD201 study test engine. ACD201 test engine can simulate the examination on the spot. As some statistics revealed, the bad result not only due to the poor preparation, but also the anxious mood. Now, our ACD201 Simulated Test engine can make you feel the actual test environment in advance. Besides, the high quality ACD201 valid exam dumps will help you prepare well. You can must success in the ACD201 real test.

Appian ACD201 Exam Syllabus Topics:

Topic	Details
Topic 1	<ul style="list-style-type: none">Performance and Scalability: This section of the exam measures the skills of Appian Developers and covers designing performant components, building memory-efficient models, conducting database load and automated testing, analyzing server memory and logs for risks, and resolving performance concerns for end users.
Topic 2	<ul style="list-style-type: none">Data Management: This section of the exam measures the skills of Solution Architects and covers designing and documenting data models, recommending record types for optimization and security, working with relational data, and implementing complex models with supporting objects.
Topic 3	<ul style="list-style-type: none">Application Design and Development: This section of the exam measures the skills of Appian Developers and covers translating business requirements into user stories, building end-to-end applications, managing object security, applying design best practices, and ensuring user-centric experiences through proper UX design.
Topic 4	<ul style="list-style-type: none">Advanced Appian Concepts: This section of the exam measures the skills of Solution Architects and covers creating and configuring authentication methods, setting up integration objects, designing APIs, evaluating smart services and plug-ins, and applying SSO mechanisms with SAML and LDAP.

Guaranteed Appian ACD201 Passing, Valid Exam ACD201 Registration

It is a challenging exam and not a traditional exam. But complete Appian ACD201 exam preparation can enable you to crack the Appian ACD201 exam easily. For the quick and complete Appian Senior Developer (ACD201) exam preparation you can trust ACD201 Exam Practice test questions. The Appian ACD201 exam practice test questions have already helped many Appian ACD201 exam candidates in their preparation and success.

Appian Senior Developer Sample Questions (Q31-Q36):

NEW QUESTION # 31

You're developing a user input form for a financial services application.

One component of this form requires you to embed an external site (in the form of a URL) that can receive inputs from Appian and return updates as the user interacts with it.

What should you do?

- A. Use a component plug-in to achieve interaction with the embedded site, because this use case requires input-output or event parameters.
- B. Use an integration to achieve interaction with the embedded site while displaying information in the Appian out-of-the-box Web Content Component. That way you can refresh for updates and interact with the site.
- C. Use the Appian out-of-the-box Web Content Component. It can display content from a URL and provide both additional inputs, or receive updates as the user interacts with the embedded site.

Answer: A

Explanation:

A component plug-in is required because the standard Web Content Component in Appian only displays external sites and does not support direct two-way interaction or input/output event handling. A plug-in enables custom integration to allow input and receive updates as users interact with the embedded site.

NEW QUESTION # 32

You're creating a new entity-backed record type without data sync enabled. Users in the "ACME Supervisors" group must be able to access the Delete Case related action on the record summary view.

What are two valid steps that you should perform to grant users the appropriate access?

- A. Grant Viewer permissions to the underlying interface object for the Summary view.
- B. Update the record action security for "Delete Case" and include the "ACME Supervisors" group within the visibility settings.
- C. Add the "ACME Supervisors" group as a Viewer to the data store mapped to the record.
- D. Add the "ACME Supervisors" group as a member of the "Database Editors" group and grant DELETE privileges.

Answer: B,D

Explanation:

Adding "ACME Supervisors" as members of "Database Editors" with DELETE privileges allows them to perform delete operations. Updating the "Delete Case" record action security and including "ACME Supervisors" in the visibility settings ensures they have access to the related action on the record summary view.

NEW QUESTION # 33

Which two options are correct regarding record relationships when using synced records in your application?

(Choose two.)

- A. Record type relationships enforce referential integrity.
- B. Up to 20 relationships can be added on a record type with data sync enabled.
- C. Relationships can be established between any record types that have sync enabled, regardless of the source.
- D. Many-to-one relationships can be established using record relationships.

Answer: B,D

Explanation:

Many-to-one relationships can be established between record types using record relationships.

Up to 20 relationships can be added to a record type with data sync enabled, according to Appian platform limits.

NEW QUESTION # 34

You're developing a case management application. Currently, users can view a list of all cases designed using a `!forEach()` loop from a record query. Users can navigate to the case summary page which shows the case details in a two (2) column layout. Users are also able to update the case details.

During the last round of UAT testing, users reported that the tool was not intuitive to use.

Match each feedback comment to the suggested UI/UX improvements.

Note: Each UI/UX improvement will be used once, or not at all. To change your responses, you may deselect your response by clicking the blank space at the top of the selection list.

"Updating a case is time-consuming because I have to search the case list, select a case, go to 'Related Actions,' and find the correct action."
Select a match:

Use `"a!recordActionField()"` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen in "a!columnslayout()"`
Use `"a!isNativeMobile()"` to dynamically manage screen width.

"When I navigate to the landing page, the page is slow to load and shows me all cases at once."
Select a match:

Use `"a!recordActionField()"` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen in "a!columnslayout()"`
Use `"a!isNativeMobile()"` to dynamically manage screen width.

"When I access the application on my mobile device, the summary page is difficult to read as the columns become too narrow."
Select a match:

Use `"a!recordActionField()"` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen in "a!columnslayout()"`
Use `"a!isNativeMobile()"` to dynamically manage screen width.

Answer:

Explanation:

"Updating a case is time-consuming because I have to search the case list, select a case, go to 'Related Actions,' and find the correct action."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

"When I navigate to the landing page, the page is slow to load and shows me all cases at once."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

"When I access the application on my mobile device, the summary page is difficult to read as the columns become too narrow."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

"Updating a case is time-consuming because I have to search the case list, select a case, go to 'Related Actions,' and find the correct action."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

"When I navigate to the landing page, the page is slow to load and shows me all cases at once."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

"When I access the application on my mobile device, the summary page is difficult to read as the columns become too narrow."
Select a match:

Use `recordActionField()` to display relevant related actions in a grid interface.
Use record-backed read-only grids to display your data and manage paging automatically.
Use sections, box layouts, or rich text components to structure your interfaces.
Use `stackWhen` in `columnsLayout()`.
Use `isNativeMobile()` to dynamically manage screen width.

NEW QUESTION # 35

You're building an order management system.

You need to create a field that displays the total number of days it took to deliver each order, and show and sort by this data in a read-only grid. The data must be available in real-time.

Which function is the most appropriate for calculating this value?

- A. `subtractDateTime()`
- B. `customFieldDateDiff()`
- C. `calworkdays()`

Explanation:

NEW QUESTION # 36

Guaranteed ACD201 Passing: <https://www.lead2passexam.com/Appian/valid-ACD201-exam-dumps.html>

- [illegible]