



Free Questions for [ACD101](#) by [go4braindumps](#)

Shared by [Rosales](#) on [22-07-2024](#)

For More Free Questions and Preparation Resources

[Check the Links on Last Page](#)

Question 1

Question Type: MultipleChoice

You write an expression that checks the validity of an email address.

Which three scenarios should you configure as test cases? (Choose three.)

Options:

- A- An invalid email address that is missing the @-character: 'john.doeexample.com'.
- B- An invalid email address: null.
- C- A valid email address: 'jane.doe@example.com'.
- D- The mail server is unavailable.

Answer:

A, B, C

Explanation:

When testing the validity of an email address in Appian, you should consider various scenarios to ensure robust validation. This includes checking for the presence of the @-character, which is essential for a valid email format (A), testing how the expression handles null values (B), and verifying that a correctly formatted email address is validated as such (C). The availability of the mail server (D) is not relevant to the format validation of an email address within an expression. Reference: Appian Documentation - Expression Language Syntax

Question 2

Question Type: MultipleChoice

ACME Automobile uses Appian to manage their vehicle fleet. Vehicle records can have a status of either "active" or "inactive".

Users are primarily concerned with active vehicles and want to see only those records by default when viewing the Vehicle records list. However, it is important for users to be able to see the unfiltered list of Vehicle records on demand to address occasional auditing requests from managers.

Which configuration supports the desired Vehicle record list behavior?

Options:

- A- Visibility on the Status column in the Vehicle record list set with conditional logic.
- B- A source filter set to exclude vehicles with status 'inactive'.
- C- A user filter for the status field with a default option corresponding to 'active'.

Answer:

C

Explanation:

To achieve the behavior where users see only 'active' vehicle records by default but can also view all records when needed, you should configure a user filter for the status field on the Vehicle record list. This user filter should have a default value set to 'active', which will filter the list to only show active records initially. However, users will still have the option to adjust the filter to see all records, thus accommodating occasional auditing requests. Reference: Appian Documentation - Record List Filters and User Filters

Question 3

Question Type: MultipleChoice

You receive a bug ticket that states "After selecting a value for the drop-down field, the value disappears."

You investigate and notice that when you select the drop-down, the proper choice labels display. When you select an option, the value updates properly in the corresponding rule input.

What is the issue and how can you fix this bug?

Options:

- A-** The value parameter is improperly configured on the drop-down component. You need to map the value to the proper rule input or variable.
- B-** The user group for the lookup table is incorrect. You need to add the user to the proper group.
- C-** The choice labels parameter of the drop-down field is not configured as a list. You need to wrap the value with curly brackets.

Answer:

A

Explanation:

The described bug typically occurs when the value parameter of the drop-down component is not correctly mapped to the corresponding rule input or variable that is supposed to hold the selected value. To fix the issue, you should ensure that the drop-down's value parameter is correctly mapped so that the selected option is retained and displayed properly. Reference: Appian Documentation - Dropdown Field Component

Question 4

Question Type: MultipleChoice

Review the following code snippet:

```
displayvalue(1, {0, 1, 2}, {"Low", "Medium", "High"}, "Unknown")
```

The definition of displayvalue is:

Tries to match a value in a given array with a value at the same index in a replacement array and returns either the value at the same index or a default value if the value is not found.

What does this code snippet return?

Options:

A- Low

B- Medium

C- High

Answer:

B

Explanation:

The `displayvalue()` function matches a given value with an array and returns the value at the same index from a replacement array or a default value if not found. In the snippet `displayvalue(1, {0, 1, 2}, {'Low', 'Medium', 'High'}, 'Unknown')`, the value 1 is matched in the array {0, 1, 2} at index 1. The function then returns the value at index 1 in the array {'Low', 'Medium', 'High'}, which is 'Medium'. Reference: Appian Expression Language Documentation - Functions

Question 5

Question Type: MultipleChoice

Which Appian feature can help the implementation team analyze the event log data of an existing process?

Options:

- A-** RPA
- B-** Process Mining
- C-** Workflow

Answer:

B

Explanation:

vent logs. It uses the data from process models to give insights into process performance, bottlenecks, and compliance with the designed process flow. This feature is valuable for understanding the actual performance of business processes and for identifying areas for improvement. Reference: Appian Documentation - Process Mining

Question 6

Question Type: MultipleChoice

Which statement about local variables is valid?

Options:

A- The data type of a local variable is determined by its value.

B- Local variables can only store primitive data types, such as numbers and strings.

C- Local variables must have an initial value set when defining them.

Answer:

A

Explanation:

In Appian, the data type of a local variable is inferred from the value it is set to. Unlike some other programming languages where the data type must be explicitly declared, Appian determines the data type automatically based on the initial value assigned to the local variable. Local variables in Appian are quite flexible and can store various types of data, including complex data types, not just primitive ones. Reference: Appian Documentation - Local Variables

Question 7

Question Type: MultipleChoice

Which set of out-of-the-box features is only available when data sync is enabled on a record type?

Options:

A- Generate record actions

Define record type object security

Add custom record fields

B- Define record type relationships

Add custom record fields

Configure record-level security

C- Define record type relationships

Add hidden record fields

Configure record-level security

Answer:

C

Explanation:

Data sync enables additional features for record types in Appian. With data sync enabled, you can define relationships between different record types, add fields to a record type that do not appear in the source database (hidden fields), and configure record-level security to control access to individual records based on user roles or other criteria. These features are part of the enhanced functionality provided by data sync to ensure efficient data management and security within Appian applications. Reference: Appian Documentation - Record Type Features and Data Sync

Question 8

Question Type: MultipleChoice

Review the following code snippet:

```
index({"a", "b", "c"}, 1, "x")
```

Which value is returned?

Options:

A- a

B- b

C- c

D- x

Answer:

D

Explanation:

The `index()` function in Appian is used to replace an element at a specified index in a list. In the code snippet `index({'a', 'b', 'c'}, 1, 'x')`, the function is called to replace the element at index 1 of the list `{'a', 'b', 'c'}` with the string `'x'`. Since Appian lists are 1-indexed, this means that the first element `'a'` will be replaced by `'x'`. However, since the function is used to modify a list and not to return a value, if this function is used as is without being part of a larger expression that outputs the list, it would not return any of the listed values. Reference: Appian Expression Language Documentation - Functions

Question 9

Question Type: MultipleChoice

You configured a Custom Data Type (CDT), `ACME_video`. You need to set up a node in your process model to write a new video as a row in the CDT's corresponding database table.

Which set of inputs must you configure on your Write to Data Store smart service node?

Options:

A- The data store and an input corresponding to each field of the `ACME_video` CDT.

B- The data store entity and an input of type ACME_video.

C- The data store and an input of type ACME_video.

Answer:

B

Explanation:

When configuring a Write to Data Store Entity smart service node to write a new row to a CDT's corresponding database table, you must specify the data store entity where the CDT is stored. Additionally, you need to provide an input of the type that matches the CDT, in this case, an input of type ACME_video, which will contain the data to be written to the database. Reference: Appian Documentation - Write to Data Store Entity Smart Service

To Get Premium Files for ACD101 Visit

<https://www.p2pexams.com/products/acd101>

For More Free Questions Visit

<https://www.p2pexams.com/appian/pdf/acd101>

