

Higher Quality  
Better Service!

**EXAM SELL**

Certified IT practice exam authority

Accurate study guides, High passing rate!

Exam Sell provides update free of charge in  
one year!



**Exam** : **ACD101**

**Title** : **Appian Associate Developer**

**Version** : **DEMO**

1. A customer wants to display a small toolbar with three icons - a "sad face," a "face with neutral expression," and a "happy face" - on the bottom of every page in their application.

Users will be instructed to use the icon that best expresses their current experience using the application. This will allow the customer to collect valuable data about users.

Which object type should be called from each page to implement this feature?

- A. An interface, because the component must render user interface elements.
- B. An expression rule, because the component captures expressions of user sentiment for analysis.
- C. A decision, because the component captures a choice that users select from an array of custom selection components, rather than a standard dropdown or radio button.

**Answer:** A

**Explanation:**

In Appian, interfaces are used to design and render user interface elements, allowing for the creation of custom layouts and components that can interact with users. In this scenario, the toolbar with emotion icons is a UI element that needs to be consistently presented across various pages. Using an interface ensures that these icons can be interactively used by users to express their sentiments, and the data collected can be analyzed for user experience insights. Interfaces are capable of embedding such UI elements and can be invoked or included across multiple pages within an application, making them the ideal choice for this requirement.

Reference: Appian Documentation on Interfaces: Provides detailed guidance on how to design and use interfaces to create user-centric UI components.

2. You select the "Generate groups and folders to secure and organize objects" option while creating a new application, Acme, with the prefix ACM.

By default, which two groups are generated by Appian? (Choose two.)

- A. ACM Administrators
- B. ACM Designers
- C. ACM Viewers
- D. ACM Users

**Answer:** A, B

**Explanation:**

When creating a new application in Appian and opting to generate groups and folders for organization and security, Appian automatically creates specific groups to facilitate application development and management. The default groups include "Administrators" and "Designers" with the application prefix, in this case, ACM. The ACM Administrators group is intended for users who will have full control over the application, including configuration and deployment aspects. The ACM Designers group is designated for users primarily involved in the design and development of the application, granting them necessary permissions to create and modify application components without full administrative privileges.

Reference: Appian Documentation on Application Design: Offers insights into best practices for structuring and securing Appian applications, including the use of groups for effective application management.

3. You need to remove an unused field from an existing record type Product, which has data sync enabled and is backed by a database table.

What should you do?

- A. Delete the field from the record type and optionally delete the column from the database table.
- B. Delete the field from the product Custom Data Type (CDT) and perform a full resync of the record type.
- C. Delete the column from the database table and perform a full resync of the record type.

**Answer: B**

**Explanation:**

In Appian, when dealing with a record type that has data sync enabled and is backed by a database table, changes to the structure of the underlying data model, such as removing a field, should be carefully managed. The correct approach involves deleting the unused field from the Custom Data Type (CDT) that defines the structure of the data for the record type. Following this change, a full resynchronization of the record type is necessary to ensure that the changes in the CDT are reflected in the record type and its associated data in Appian. This process ensures data integrity and consistency across the application and the database.

Reference: Appian Documentation on Data Management: Provides guidelines on managing data structures, including CDTs and record types, within Appian applications.

4.You have a record action that should only be visible to certain users under conditions specified by an expression.

How should you configure this?

- A. Set security on the process model.
- B. Set permissions directly on the user object.
- C. Set security on the record action.

**Answer: C**

**Explanation:**

To control the visibility of a record action based on certain user conditions or expressions, the security settings of the record action itself must be configured. This involves defining conditions or expressions within the record action's security settings that evaluate whether a user meets the criteria to see and interact with the action. This method allows for dynamic control over the accessibility of actions based on user roles, specific attributes, or other contextual factors, ensuring that only authorized users can see and execute the action under specified conditions.

Reference: Appian Documentation on Record Actions: Details how to configure and secure record actions, including visibility and permissions, to tailor the user experience within record views.

5.Which Appian feature is used to automate repetitive manual tasks such as extracting data from a system for which there is no API?

- A. RPA
- B. Process Mining
- C. Connected Systems

**Answer: A**

**Explanation:**

Robotic Process Automation (RPA) is the Appian feature designed to automate repetitive manual tasks, especially in scenarios where interacting with systems without APIs is necessary. RPA bots can mimic human actions, such as navigating through system interfaces, extracting data, and entering information, effectively bridging the gap between digital processes and systems that lack API

integrations. This capability is particularly useful for integrating legacy systems into modern workflows, streamlining operations that would otherwise require manual intervention.

Reference: Appian Documentation on RPA: Explains how RPA can be leveraged within Appian to automate tasks, providing examples and best practices for implementing RPA bots in business processes.