ATMIC NETWORKS
YOUR GUIDE TO GLOBAL CERTIFIATION

"Please note that these files may not be up to date. However, the questions will help you understand the exam format and typical question patterns."

www.atmicnetworks .com

Warning: Keep connected with our support team for latest updates

Question: 1

You're developing a functionality to manage transactions for an insurance company. Each transaction can have multiple items where respective item type and item amount can be entered. The amount entered can be either negative or positive, and is a decimal data type with a precision of 8 and a scale of 2.

A maximum of 10 items can be entered for a transaction. Once all the items are entered, you need to calculate and show the sum of all items' total amounts at the transaction level.

What is the best way to calculate and display the sum of all item amounts?

A. text(sum(ri!items.amount),"00.00")

B. fixed(sum(ri!items.amount),2,false())

C. round(sum(ri!items.amount),2)

Answer: B

Explanation:

fixed(sum(ri!items.amount),2,false()) is the best option because it returns the sum as a text value formatted to exactly two decimal places, which is ideal for currency or financial amounts. This ensures the total displays correctly as a string, maintaining precision and formatting for both positive and negative values.

Question: 2

You're establishing a Health Check schedule for your department.

Which two statements should you consider when implementing Health Checks? (Choose two.)

A. Run a Health Check at least once a month in production as part of server and application monitoring.

B. Run Health Checks during regular business hours in production to identify issues during heavy usage.

C. Configure a weekly Health Check cadence for the development environment.

D. Run Health Checks during regular non-business hours in production to avoid slowing down the environment.

Answer: A, D

Explanation:

Running a Health Check at least once a month in production is best practice for ongoing monitoring and early issue detection.

Running Health Checks during non-business hours in production helps avoid performance impact on end users.

Question: 3

You're developing a record view for a case management system that enables users to upload files to a specific location within an external records application.

Your requirements are:

In this record view, there should be a record action that allows the user to upload a file to the interface and configure the metadata (name, author, file size) before submitting.

If the file is larger than 250MB, the user will be notified that the file will be uploaded overnight.

Once the file has been successfully uploaded, the file will appear on the record view and the user can interact with the file.

Which two pieces of user-facing information must be captured from this business process so that you can develop this record view? (Choose two.)

A. The location of where the file will be uploaded to in the external records application.

B. The email of the system administrator so that an email can be sent to them if the integration fails.

C. The time of file upload if the file is less than 250MB.

D. How the user wishes to interact with these files in the record view - as a grid or a series of card layouts.

Answer: A, D

Explanation:

You need to know the target location in the external records application to properly upload the file.

Understanding how users want to interact with files (grid or card layout) is essential for designing the record view interface.

Question: 4

You're designing an expression rule that needs to retrieve employees from a database for a given department and display their full name concatenated with their role in a list for a dropdown field.

Which design approach should you recommend?

A. Query the data to return all employees. Using a!forEach(), for each item in the list, if the employee belongs to the given department, return the employee name and role, otherwise, return a null value.

B. Query the data to return all employees. Use wherecontains() to return the employees for the given department and use a!forEach() to return the employee name and role for each item in the list.

C. Query the data with a filter applied to only return employees for the given department. Using a!forEach(), for each item in the list, return the employee name and role.

Answer: C

Explanation:

Querying the data with a filter to return only employees for the given department is the most efficient approach. Then, using a!forEach() to concatenate and return the employee name and role meets the dropdown display requirement without unnecessary processing.

Question: 5

What are three ways to optimize the memory usage of a process model? (Choose three.)

A. Pass data directly to activity nodes and smart services.

B. Minimize the use of subprocesses in the model.

C. Decrease the number of process variables.

D. Configure process variables as parameters.

E. Divide large nodes into multiple smaller nodes.

F. Avoid storing large blocks of text in process variables.

Answer: A, D, F

Explanation:

Passing data directly to activity nodes and smart services reduces unnecessary use of process variables, optimizing memory.

Configuring process variables as parameters limits their lifespan and persistence, saving memory.

Avoiding large blocks of text in process variables minimizes memory consumption within the process model.

Question: 6

You're designing an integration object with JSON (application/json) as the content type.

Which two data types can be automatically converted to JSON in Appian? (Choose two.)

A. Appian Custom data type

B. Base64 documents inline with JSON

C. Primitive data types

D. HTML file type

Answer: A, C

Explanation:

Appian Custom data types (CDTs) can be automatically converted to JSON for integration.

Primitive data types such as text, number, and boolean are also automatically converted to JSON.

Question: 7

You're inspecting the items in a deployment package for issues you might have missed during package preparation.

Which two objects can be in your list of missing precedents for an application? (Choose two.)

A. Connected System

B. Document

C. Group Type Custom

D. System Knowledge Center

Answer: A, C

Explanation:

Connected System and Group Type Custom can both be missing precedents in an Appian deployment package, meaning dependent objects required by the application may not have been included.

Question: 8

You need to configure the security for the synced record type Case.

Your requirements are:

Only users in the "Archive Management" group can access cases in the "Archived" status.

Cases in the status "Deleted" are accessible to no one.

Users in the "All Users" group can already access the record.

What should you do?

A. Create a new user filter to allow access if the user is in the "Archive Management" group while the case status is "Archived. "

Configure the record-level security to exclude all cases that are in the "Deleted" status

B. Create a new user filter to allow access if the user is in the "Archive Management" group while the case status is "Archived. "

Configure a source filter to exclude all cases that are in the "Deleted" status.

C. Create a new record-level security rule to allow access if the user is in the "Archive Management" group while the case status is "Archived."

Configure a source filter to exclude all cases that are in the "Deleted" status.

Answer: C

Explanation:

Creating a record-level security rule for "Archived" cases restricts access based on user group, and a source filter ensures "Deleted" cases are excluded from being accessed by any user. This approach meets both security requirements efficiently.

Question: 9

You need to create a web API so that an external system can start a process in your app.

Which two statements are true about using a service account to authenticate the system? (Choose two.)

A. The service account must be granted access to that web API via group management.

B. The service account must be configured from within the web API object.

C. The service account must be a user in the Service Accounts group.

D. The service account is created as any other object in Appian.

Answer: A, C

Explanation:

The service account must be granted access to the web API, typically via group management, to allow authenticated calls.

The service account must be a user in the Service Accounts group to be used for authentication in integrations such as web APIs.

Question: 10

You're in the process of deploying a package to the client's TEST environment at the end of a sprint. This package is only relating to changes to a single application.

Which two statements best describes the practices you should follow for automated testing for expression rules either before or after deploying the package? (Choose two.)

A. Before using Compare and Deploy, individually evaluate the test cases in the expression rules you intend to deploy. For any failed test case, adjust the expression rule so that it passes.

B. After deploying the package, you should perform regression testing by running the Start Rule Tests (Applications) smart service to automatically run the test cases. This indicates whether there are any impacts caused by changes to the application.

C. During Compare and Deploy, the Inspect Deployment tab automatically runs the test cases in each of the expression rules. Review these findings, and either adjust the expression rule or delete outdated test cases.

D. After deploying the package, you should perform regression testing by running the Start Rule Tests (All) smart service to automatically run the test cases. This indicates whether there are any impacts caused by the changes across multiple applications.

Answer: A, B

Explanation:

Individually evaluating and fixing expression rule test cases before deployment ensures the new package does not introduce errors.

Running regression tests after deployment using the Start Rule Tests (Applications) smart service validates that the deployed changes did not negatively impact the application.

Question: 11

You're developing a user input form for a government services application that includes a file upload component.

If the uploaded file is a multimedia file (for example: .MP4, .MOV or .WAV), the customer wants these uploaded to Microsoft SharePoint instead.

What is the best way to achieve this requirement?

A. Use the out-of-the-box Appian Connected System, but since it does not have any pre-built connected systems, you will need to use either a HTTP or OpenAI Connected System to connect with SharePoint.

B. Develop a connected system plug-in given SharePoint has an extensive API that cannot easily be resolved using the Appian out-of-the-box Connected System.

C. Use the out-of-the-box Appian Connected System as it has a pre-built connected system to SharePoint to provide a guided experience for designers to send and receive data.

Answer: C

Explanation:

Appian provides an out-of-the-box pre-built connected system for SharePoint, allowing you to integrate file uploads to SharePoint with a guided experience for configuration—making this the best way to fulfill the requirement.

Question: 12

Your client reported that a form in the application is very slow to load.

You investigate and find a query entity which is nor performing well.

Which action should you perform to improve query performance for the query entity?

A. Change the data source of the query entity to a view with a WHERE clause.

B. Apply indices on all the fields referred in the query.

C. Apply the appropriate filters and indices in both Appian and the database.

Answer: C

Explanation:

Applying appropriate filters and indices in both Appian and the database ensures that only relevant data is retrieved and the database query runs efficiently, improving overall query performance.

Question: 13

You need to write an expression to retrieve a list of all account managers who don't currently have an active customer account.

Which code snippet should you use?

A)

```
a!localvariables(
    local IdomesticCustomers: rule!getAllDomesticCustomersl),
    local!internationalCustomers: rule!getAllInternationalCustomers(), local I customers: a!flatten(

        localidomesticCustomers, local! internationalCustoniers
    }

    locallcustomerManagers: index(local!customers, "manager", null), localiaccountManagers:
    rule!getAccountManagersByRegionAndVertical(
```

```
    regions: rule!getAllRegions(),

    verticals: rule!getAllVerticals()

  ocaliAMswithoutAccounts: reject(
    fnlisnull,
    a!forEach(
      items: localIaccountManagers,
      expression: if(

          contains{local!customerManagers, fv!item), null, fvlitem

  ),
  a!forEach(

    items: localJAHsWithoutAccounts, expression: rule!displayUserName(fv1 item)
```

B)

```
a!localVariables(
  local'customers: {
    rule!getAl 1OomesticCustoners(),
    rule lgetAlIInternat ianaKustarers()
  localiaccountManagers:  rule!getAccountHanagersByRegionAndVertical(  regions:
    rule'getAllRegions(), verticals: ruleigetAllVerticalsf)

  localIAMsWithoutAccounts: difference(localiaccountManagers, local!< a!forEach(
    items: localSAMsWithoutAccounts,
    expression: rule ldisplayUserName(fvlitem)
```

C)

```
a!localVariables(
    local!domesticCustomers: rule!getAliDomesticCustomers(),
    local!internationalCustomers: rule! getAl UntemationalCustomi
    local!customers: a!flatten({local!domesticCustomers,local!in'
    localJcustomerManagers: index(local!customers,"manager",null
    local!accountManagers: rule!getAccountManagersByRegionAndVer
      regions: rule!getAllRegions(),
      verticals: rule!getAllVerticals()
```

```
local!AMsWithoutAccounts: difference(local!accountManagers, a!forEach(
    items: localSAMsWithoutAccounts,
    expression: rule!displayUserName(fv!item)
```

A. Option A

B. Option B

C. Option C

Answer: B

Explanation:

Chosen snippet uses the difference() function to efficiently get all account managers who are not currently managers of any customer account. It directly compares local!accountManagers to local!customers.manager, resulting in a concise and optimal solution for retrieving account managers without an active customer account.

Question: 14

You're creating a new record type with data sync enabled. Users in the "ACME Employees" group must be able to access the record list and start the New Case action.

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

A. Add the “ACME Employees" group as a Viewer on the record type.

B. Add the "ACME Employees" group as a Viewer to the data store mapped to the record.

C. Grant Viewer permissions to the underlying interface object for the record list.

D. Grant Initiator permissions to the underlying process model for the New Case action.

Answer: A, D

Explanation:

Adding the “ACME Employees” group as a Viewer on the record type allows them to access the record list.

Granting Initiator permissions to the process model for the New Case action enables group members to start that action.

Question: 15

You have designed a three-step 'Wizard' form interaction using user input tasks in the process model. The second step of the Wizard must utilize another process model which contains a user input task inside.

Which two methods should be implemented for a seamless Wizard-like interaction for the user? (Choose two.)

A. Enable activity-chaining between all nodes on the main process model.

B. Enable activity chaining between all nodes in the second process model, and activity chain between all user input tasks.

C. Use a subprocess to call the second process model.

D. Use the Start Process smart service to call the second process model.

Answer: A, C

Explanation:

Enabling activity-chaining between all nodes on the main process model ensures the wizard steps flow seamlessly for the user.

Using a subprocess to call the second process model allows the main process to incorporate its user

input task as part of the wizard flow with activity-chaining support.

Question: 16

You need to implement a field-level audit functionality on the application data, and then display a log of the changes made over time to the users.

What should you do?

A. Create Audit table(s) and create stored procedures on each transaction table to compare and save the modified data.

B. Create utility process model(s) to insert new version of the modified data for the same record into the transaction tables on modification.

C. Create Audit table(s) and create triggers on each transaction table to record modifications into the Audit table.

Answer: C

Explanation:

Creating audit tables and using triggers on each transaction table ensures that all field-level changes are automatically recorded at the database level, providing a reliable and consistent audit log for display to users.

Question: 17

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

A. Relationships can be established between any record types that have sync enabled, regardless of the source.

B. Many-to-one relationships can be established using record relationships.

C. Record type relationships enforce referential integrity.

D. Up to 20 relationships can be added on a record type with data sync enabled.

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.

Question: 18

You're creating a support case record type while working on a support case management system. You want to track support case status changes.

You want to use record events to track this information.

Which record data source type is the most appropriate for you to use?

A. Web service without data sync enabled

B. Database source with data sync enabled

C. Process Model source with data sync enabled

Answer: B

Explanation:

A database source with data sync enabled is required to use record events for tracking support case status changes, as Appian record events are only supported for synced record types with a database as the data source.

Question: 19

A local variable is reevaluated every time a form loads and you notice subsequent slow performance of the form.

Which two steps could you perform to optimize this? (Choose two.)

A. Create a local variable inside with () for faster reevaluation.

B. Ensure the local variable is refreshed after a specified interval, and nor refreshed when referenced variable are changed.

C. Avoid executing queries every time a form reevaluates.

D. Ensure the local variable only updates when a referenced variable is modified or when it's being updated in the saveInto parameter of a component.

Answer: C, D

Explanation:

Avoiding queries on every form reevaluation prevents unnecessary data retrieval and improves form performance.

Ensuring the local variable only updates when necessary (e.g., on relevant variable changes or via saveInto) reduces redundant recalculations and optimizes responsiveness.

Question: 20

A process was recently pushed to the Prod environment. However, users are reporting that a task on this process is experiencing slow response times when they try to input data.

What is the most likely reason for this issue?

A. The process uses too many hidden variables.

B. There are too many SAIL components on the task.

C. Exception timers on tasks are set to values that are too low.

D. The task is assigned to too many users.

Answer: B

Explanation:

Having too many SAIL components on the task can cause slow response times for users, as the

interface may take longer to render and process inputs, impacting performance.

Question: 21

Which step should you perform to identify expression rules that have been causing or have caused performance issues in the past 30 days?

A. Monitor the runtime of all test cases.

B. Navigate to the Admin Console > Rule Performance tab.

C. Examine the application server log

Answer: B

Explanation:

The Admin Console > Rule Performance tab provides metrics and insights on expression rules that have caused performance issues in the past 30 days, allowing you to identify and address slow- running rules.

Question: 22

You need to connect to an external system using OAuth 2.0: SAML Bearer Assertion Flow authentication type, which requests access to an API on behalf of a signed in user.

This standard has the several steps involved with the SAML Bearer Assertion Flow.

Which two steps should you perform? (Choose two.)

A. Add required users and groups to OAuth 2.0 SAML Bearer Assertion Users system group.

B. Enable the checkbox labeled OAuth 2.0: SAML Bearer Assertion Flow in the Admin Console.

C. Create a Bearer Assertion process flow to authenticate the user.

D. Upload Client Certificate to Connected System.

Answer: A, D

Explanation:

Adding required users and groups to the OAuth 2.0 SAML Bearer Assertion Users system group ensures only authorized users can initiate the authentication flow.

Uploading a client certificate to the Connected System is needed for the assertion and secure communication required by the SAML Bearer Assertion Flow.

Question: 23

Which two items are configured in the Admin Console when you create a web API? (Choose two.)

A. API Key

B. Access Control Policies

C. Service Account

D. LDAP Authentication

Answer: A, C

Explanation:

API Keys are configured in the Admin Console to control and secure access to web APIs.

Service Accounts are managed in the Admin Console to allow external systems to authenticate and call the web APIs.

Question: 24

What are three locations to call an Integration that changes the source data? (Choose three.)

A. Rule

B. Web API (GET)

C. Process model (Call Integration Smart Service]

D. Web API (POST, PUT, DELETE)

E. Interface component savelnto parameter

Answer: A, C, D

Explanation:

Integrations can be called from rules, allowing for source data changes when used in the proper context.

The Call Integration Smart Service in a process model can execute integrations that modify source data.

Web APIs using POST, PUT, or DELETE methods are typically used to change source data via integrations.

Question: 25

You need to update an existing Appian application.

The application needs to be responsive and compatible with a variety of screen widths. For example: phone/tablet browser, Appian mobile app, desktop monitor.

Which two functions/parameters should you consider when updating the interfaces? (Choose two.)

A. stackWhen in Columns Layout and Side By Side Layout

B. a!isNativeMobile()

C. Width of a page in the site object

D. a!isPageWidth()

Answer: A, B

Explanation:

The stackWhen parameter in layout components helps interfaces adapt to different screen sizes by controlling when columns or side-by-side layouts stack vertically.

The a!isNativeMobile() function detects if the interface is being viewed on a native mobile app, allowing you to tailor the interface for mobile responsiveness.

Question: 26

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 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.

B. 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.

C. Use a component plug-in to achieve interaction with the embedded site, because this use case requires input-output or event parameters.

Answer: C

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.

Question: 27

You're refining a story for a centralized distribution center that extends the current application by allowing inventory managers to review orders.

These orders are created through a record action that various workshops access from a shared Appian site.

Which consideration best suits this use case?

A. How will you handle errors resulting from a failed integration as a result of the workshops using an external system to place orders?

B. What are the personas that will be utilizing this application so that an Appian group structure can be developed and maintained?

C. What components have already been built relating to this piece of functionality so that you can reuse them and speed up development?

Answer: C

Explanation:

Identifying the personas (such as inventory managers and workshop users) ensures you can design and maintain an appropriate Appian group structure for managing access and permissions, which is critical for a centralized application accessed by multiple roles.

Question: 28

A recent Health Check report of your application indicates an increased risk of large context size in the interfaces.

What are two possible contributors to this risk? (Choose two.)

A. Data transformation is performed inside the interface rather than delegated to the source system.

B. Multiple local variables hold duplicate data.

C. The load variable has a web service response as JSON.

D. Local variables are set to refresh every five minutes.

Answer: A, B

Explanation:

Performing data transformation inside the interface can increase the amount and size of context data stored, raising context size risk.

Multiple local variables holding duplicate data unnecessarily inflate the interface's memory footprint, contributing to large context size.

Question: 29

The synced record Task has a self-referential relationship defined in the column parentTaskId. There is a many-to-one record relationship between the id and parentTaskId called parentTask.

For a given task ID, you need to return the task name and the parent task name.

What should you do?

A. Create a sync-time custom record field on the Task record called parentName. Specify this field to return in the query field selection.

B. Use a!queryRecordType() with a filter on the task id, with fields specified to return recordType!Task.name and recordType!Task.parentTask.name.

C. Use a!queryRecordType() filtered on the task id once to return the task name and parent task ID.
Query the record again to return the parent task name.

Answer: B

Explanation:

Using a!queryRecordType() with a filter on the task ID and specifying both recordType!Task.name and recordType!Task.parentTask.name in the fields will efficiently return both the task name and its parent task name in a single query.

Question: 30

You're creating a synced record type which will store read-only data from an external web service. Data in the external system is updated nightly, and the latest information must be available to users in the morning. Currently there are approximately 150,000 entries in the external system, and this number is not expected to change significantly.

There is an existing integration object and associated expression rule, which when called returns the complete dataset from the external system.

What are two configurations that you must apply to ensure that the latest and complete record set is available? (Choose two.)

A. Within the record type, schedule a full sync to occur nightly after the data is refreshed in the external system.

B. Modify the integration object and associated expression rule to return batches of 1000 based on a batch number passed in from the record type.

C. Within the Source Filters settings on the record type, disable the row limit option so that all entries can be retrieved from the external system.

D. Set all record fields to evaluate in real-time so that the latest information is always retrieved from the web service.

Answer: A, C

Explanation:

Scheduling a nightly full sync ensures the record type is updated with the latest data after the external system refreshes.

Disabling the row limit in Source Filters allows the complete dataset to be retrieved and synced, ensuring all entries are available.

Question: 31

You need to display information for pending tasks for each individual in an application. The tasks may be for a single specific application.

Which context type should you select for the Task Report Type?

A. Tasks attributed to user

B Tasks by process

C. Tasks by process model

Answer: A

Explanation:

"Tasks attributed to user" is the correct context type, as it allows you to display pending tasks assigned to each individual in the application.

Question: 32

You're creating a point-of-sale (POS) system for a customer which can associate the items in a transaction to a customer based on a membership number. The item stock table and the customer sales record are hosted elsewhere and need to be written to via POST APIs from an Appian process model. This all occurs within a transaction process model that you're creating.

Which two steps should you perform to design your process model to improve process memory? (Choose two.)

A. Ensure that the entire process is activity-chained so that the person entering the transaction knows when the process is completed.

B. Create the two POST APIs as asynchronous subprocesses.

C. Create the two POST APIs as synchronous subprocesses.

D. Add a terminate process node to the end of the process.

Answer: B, D

Explanation:

Creating the POST API calls as asynchronous subprocesses allows them to run independently, reducing memory usage in the main process.

Adding a terminate process node at the end ensures the process completes and releases memory as soon as its work is finished.

Question: 33

You need to create a read-only grid that displays support case information. Data for the support cases is mapped to the "Case" record type with data sync enabled.

Within the record, there are fields named "createdDateTime" and "resolvedDateTime". These two fields store the timestamps for when a case is created and resolved, respectively.

The grid includes a column indicating the number of hours between when the case was created and when it was resolved. This column must be sortable within the grid.

How should you calculate the hour value that will display in the column?

A. Create a custom record field that evaluates in real-time, and calculates the number of hours between when the case was created and when it was resolved.

B. Within the grid column, utilize an Appian function to calculate the number of hours between when the case was created and when it was resolved.

C. Create a custom record field that evaluates at sync-time, and calculates the number of hours between when the case was created and the current date and time.

Answer: A

Explanation:

A custom record field that evaluates in real-time and calculates the hours between "createdDateTime" and "resolvedDateTime" ensures the calculated value is available as a sortable field within the read-only grid.

Question: 34

You need to add test cases to an existing expression rule in the system. Expression rule gives a complex data type value as an output result.

The test case should pass only when a field in the resultant data matches with a specific value mentioned.

How should you configure the test case assertion?

A. Assertion expression evaluates to True.

B. Test output matches the asserted output.

C. Test completes without errors.

Answer: A

Explanation:

Configuring the test case assertion so that the assertion expression evaluates to True allows you to specifically check whether the desired field in the complex data type result matches the required value.

Question: 35

DRAG DROP

You're creating a process to update specific fields in a list of records and you save the updated records to the database.

You're given the following performance considerations:

- Execution time
- Process instance memory usage
- Process execution engine load balancing

Based on these performance metrics, rank the options from best to worst.

Note: To answer, move all steps from the Options list to the Answer List area and arrange them in the correct ranking order (best at the top, and worst at the bottom).

% ⁰ ⁰ 0

Configure a subprocess node which calls a process model to update an

- Individual record. Configure the MNI sellings to run one instance lor each □ record in the list and to start at the same time.
- □ □

0

Configure a start process node which calls a process model to update an indivicual record. Configure the MNI settings to tun one instance for each record In the list and to start at the same time

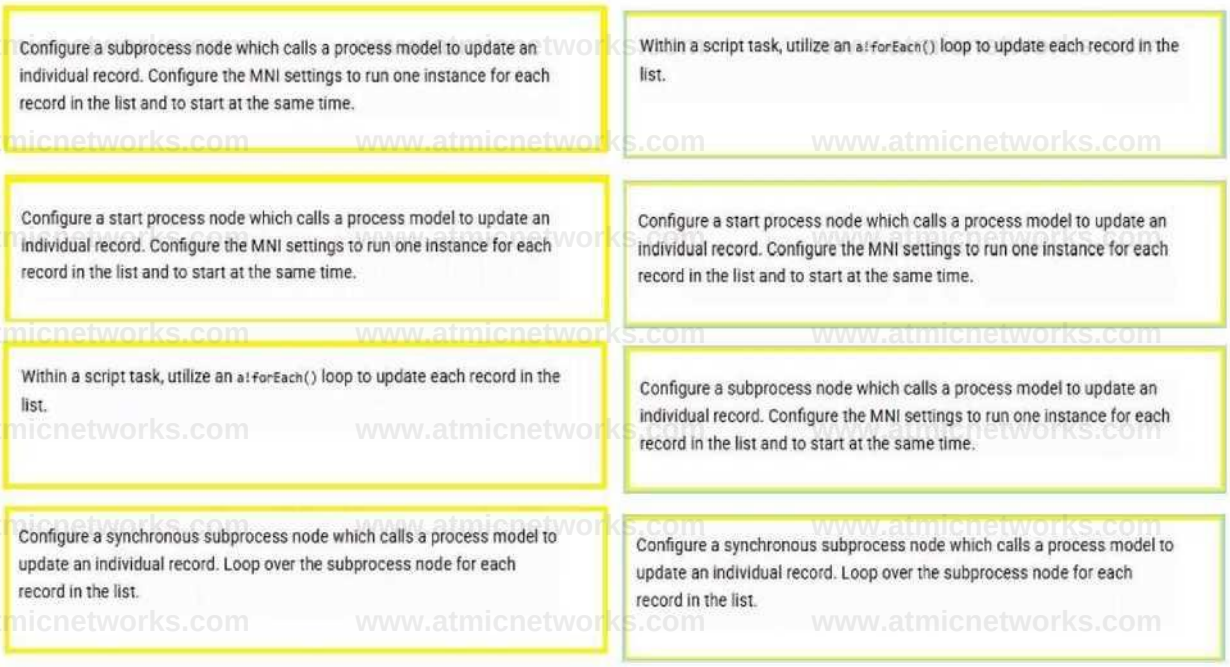
Within a script task, utilize an »ifereKn<) loop to update each record in the list.

o o

Configure a synchronous subprocess node which calls a process model to update an Individual record Loop over rhe subprocess node for each record in the list.

Answer:

Explanation:



Question: 36

You need to configure Appian user authentication against an external directory server. The requirement is to use the same corporate logon information without single sign-on (SSO).

Which authentication type should you use?

A. PIEE user authentication

B. OpenID Connect authentication

C. LDAP authentication

Answer: C

Explanation:

LDAP authentication allows users to log in with their corporate credentials by authenticating against an external directory server, without requiring single sign-on (SSO).

Question: 37

You're building an application with a review workflow.

Each submission must be approved by three users who can each approve or reject their request, and

leave a comment. This action changes the status of the submission to 'Approved' or 'Rejected' correspondingly, and pushes the review to the previous or next assignee.

Which data model captures the workflow requirements appropriately?

A. Two tables:

Review with columns id, statusId, comment, user

Ref Status with columns id, value

B. Two tables:

Review with columns id, statusId, comment, user1, user2, user3

Ref Status with columns id, value

C. Three tables:

Review to store id, statusId, comment

Ref Status with columns id, value

Review User with columns id, reviewId, user

Answer: C

Explanation:

Using three tables - Review, Ref Status, and Review User - enables you to track each user's approval or rejection, comments, and status changes for each review, supporting a flexible, normalized workflow for multiple reviewers.

Question: 38

You notice that a process consumes a significant amount of memory. The process involves ingesting long strings from a web service and saving them to the database using a looping approach. The string output is saved to a process variable with data type Text.

Which two options will reduce the memory usage? (Choose two.)

A. Save the string output directly to the data structure to be written to the database.

B. Configure the process to delete previously completed or cancelled instances.

C. Ensure that all process variables have the option "Hidden" deselected.

D. Evaluate each step in a separate script task and save the output to a process variable.

Answer: A, C

Explanation:

Saving the string output directly to the data structure for database writing avoids holding large text in process variables, reducing memory usage.

Deleting completed or cancelled process instances helps free up memory consumed by historical process data.

Question: 39

The synced record type Customer has a one-to-many relationship with the Case record type.

You need to calculate the number of open cases for each customer.

What are two valid methods to accomplish this? (Choose two.)

A. Within the Case record type, edit the relationship to the Customer record and select the "Open" status as the grouping filter.

B. Within the Customer record type, create a custom field that evaluates in real-time to aggregate related record fields and count the open cases.

C. Create a query expression for the Customer record type. Utilize the a!measure() function within the aggregation fields to count the related cases in the "Open" status.

D. Create a sync-time custom record field within the Customer record. Utilize the a!relatedRecordData() function to filter open cases and count the number of entries.

Answer: B, D

Explanation:

A real-time custom field in the Customer record type can aggregate and count related open cases.

A sync-time custom record field using a!relatedRecordData() can filter and count open cases, making the count available as a

synced field.

Question: 40

What should you use to create a predefined query in a database for ease of use?

A. Stored Procedure

B. Expression Rule

C. View

Answer: C

Explanation:

A database view is used to create a predefined query, making it easy to access, reuse, and join data for applications like Appian.

Question: 41

DRAG DROP

You need to create a new database schema using a connected data source system.

What should you do?

Note: To answer, move all steps from the Options list to the Answer List area and arrange them in the correct order.

Explanation:

Copy the credentials and connection URL.

Copy the credentials and connection URL

Create separate schemas for each application or suite of applications that you want to secure using the AppiinPrwesS .CreateNewSehemaO stored procedure.

Create separate schemas for each application or suite of applications that you want to secure using the ADoianProcess.createwewSdwaO stored procedure.

Create a MariaDB Data Source connected system for each schema using the credentials.

Create a MariaDB Data Source connected system for each schema using the credentials.

Put users in the Database Editors group to allow them to make changes in the new database schema through phpMyAdmin. Put users in the Database Viewers group to give them read-only permissions in phpMyAdmin.

Put users in the Database Editors group to allow them to make changes in the new database schema through phpMyAdmin. Put users in the Database Viewers group to give them read-only permissions in phpMyAdmin.

Question: 42

You're the administrator of your insurance company's case management system. You need to limit insurance agents to only those clients to which they have been assigned.

Which two statements must be true for one way to configure Record-Level security for the Customer record? (Choose two.)

A. Record-level security means agents must have permission to view the record type and permission to edit the records.

B. A new record-level security rule should be created using the Users found in fields option to specify that if a user is found in the "assignedAgent" field (which is a type User), then they can see their cases.

C. Data sync should be enabled.

D. A condition should be added to the record-level security rule to specify which cases specifically named agents can access.

Answer: B, D

Explanation:

Creating a record-level security rule using the "Users found in fields" option ensures agents only see records assigned to them through the "assignedAgent" field.

Adding a condition to the rule further refines which records (cases) each agent can access, implementing precise record-level security.

Question: 43

You're designing a report to show the total number of cases per month grouped by region. A synced record exists for the data source.

Which design pattern is most appropriate in this case?

A. Configure a!columnChartFieid() using a!columnChartConfig() with the case creation date as the primaryGrouping and the region as the secondaryGrouping.

B. Use a!queryRecordType() with a!aggregationFieids() to group the case creation date by region into a local variable. Pass the local variable value into the data input of a!columnChartField()

C. Configure a!pieChartFieid() using a!pieChartConfig() with the case creation date as the primaryGrouping and the region as the secondaryGrouping.

Answer: B

Explanation:

Using a!queryRecordType() with a!aggregationFields() to group by case creation date and region, storing the result in a local variable and passing it to the data input of a!columnChartField() ensures efficient aggregation and flexible chart display for your report.

Question: 44

You need to configure a process model that runs every day at 10:00 AM. The process should start only at the

specified time even when a newer version of the process model is published.

A Boolean type constant needs to be configured in the process model. The intended flow of the process should be executed only when the constant is True.

Which three steps should you perform? (Choose three.)

A. Add an expression-based condition in the start node timer event to check for the constant to be True.

B. Add a timer event in the start node.

C. Add a timer event just after the start node to check for the specific time.

D. Add a XOR Gateway just after the start node to check for the constant to be True.

E. Add an expression-based condition in the start node timer event to check for the specific time.

Answer: B, D, E

Explanation:

Add a timer event in the start node to schedule the process to start at 10:00 AM every day.

Add a XOR Gateway after the start node to check the Boolean constant and control the flow based on its value.

Add an expression-based condition in the start node timer event to ensure the process starts only at the specified time.

Question: 45

You're designing a process used to notify multiple users of marketing campaigns. These campaigns occur infrequently, but are sent out to the hundreds of customers that are stored in your app.

How should you best design your process model to send the emails whilst using a common email process model (CMN_SendEmail), to be most performant?

The outputs of this node are not required for the main process.

A. Set “CMN_SendEmail” as a “Start Process” process node and select “Automatically run multiple instances of this node” for all the users that are on the mailing list.

B. Set “CMN_SendEmair” as an asynchronous “ Subprocess" node and select "Automatically run multiple instances of this node" for all the users that are on the mailing list.

C. Set “CMN-SendEmail” as a synchronous "Subprocess" node and select “Automatically run multiple instances of this node” for all the users that are on the mailing list.

Answer: A

Explanation:

Using "Start Process" as the node type and enabling "Automatically run multiple instances of this node" for all users on the mailing list launches each email as a separate, lightweight process, maximizing performance and scalability, especially since the main process does not require the outputs.

Question: 46

HOTSPOT

Consider this POST request:

https://www.example.com/suite/webapi/customer

```
id:"7",
name:“John Smith", DOB:"1980-02-12“,
firstPurchaseDate:*,2023-03-12"
}]
```

Match the values to the correct fields in the resulting http!request value.

Note: Each value will be used once. To change your responses, you may deselect your response by clicking the blank space at the top of the selection list.

Body

Select a match:

formData

Select a match:

url

Select a match:

query Parameters

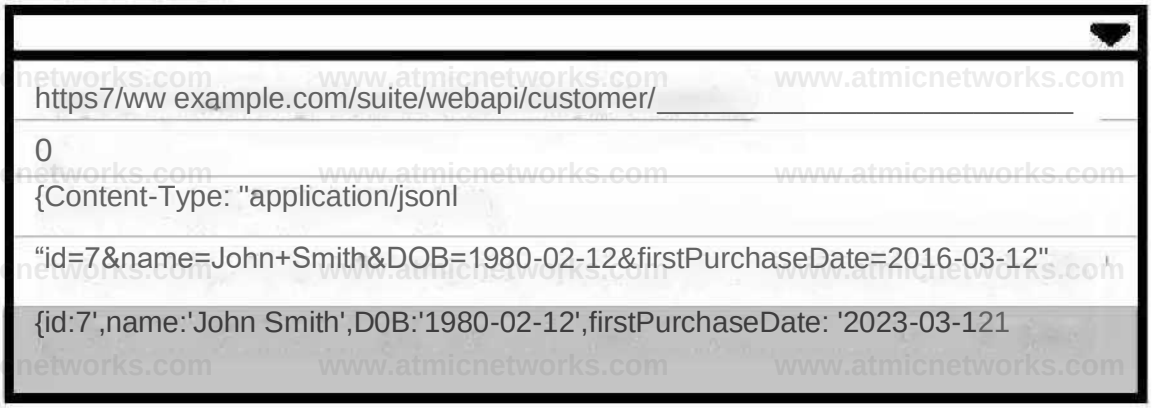
Select a match:

Answer:

Explanation:

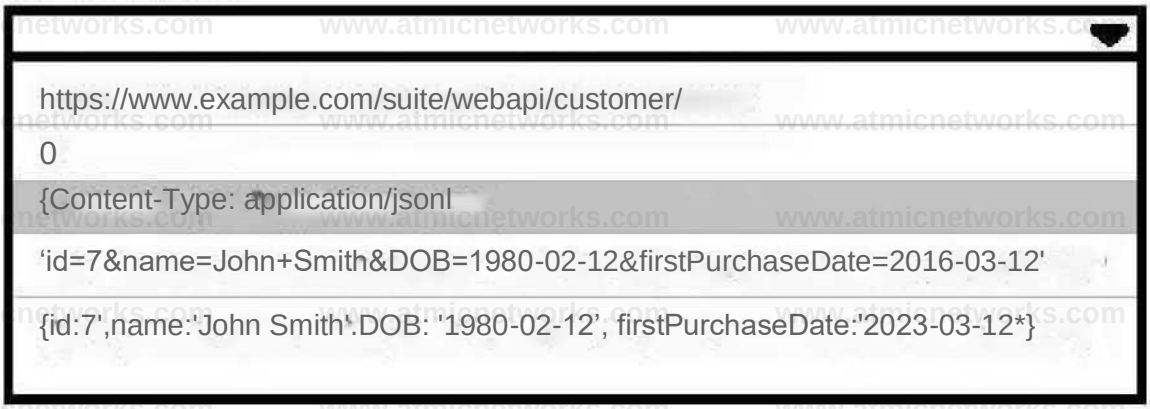
Body

Select a match:



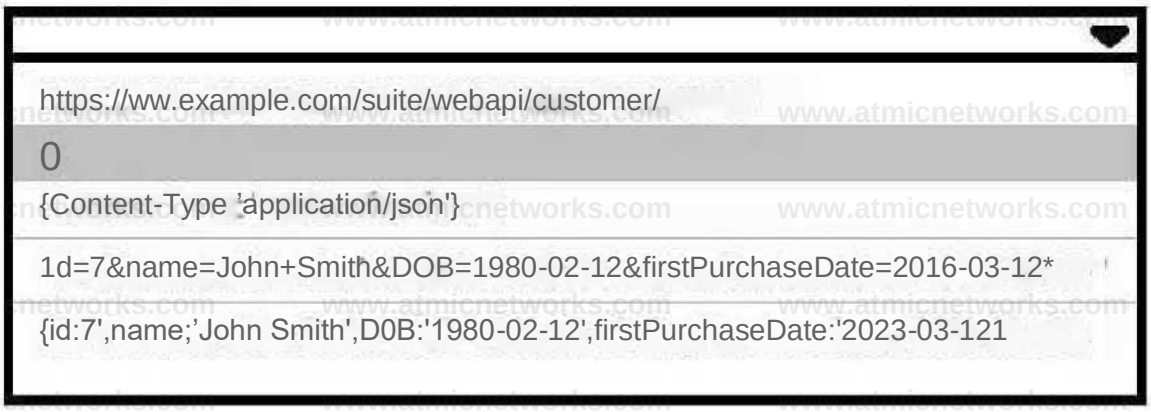
Headers

Select a match:



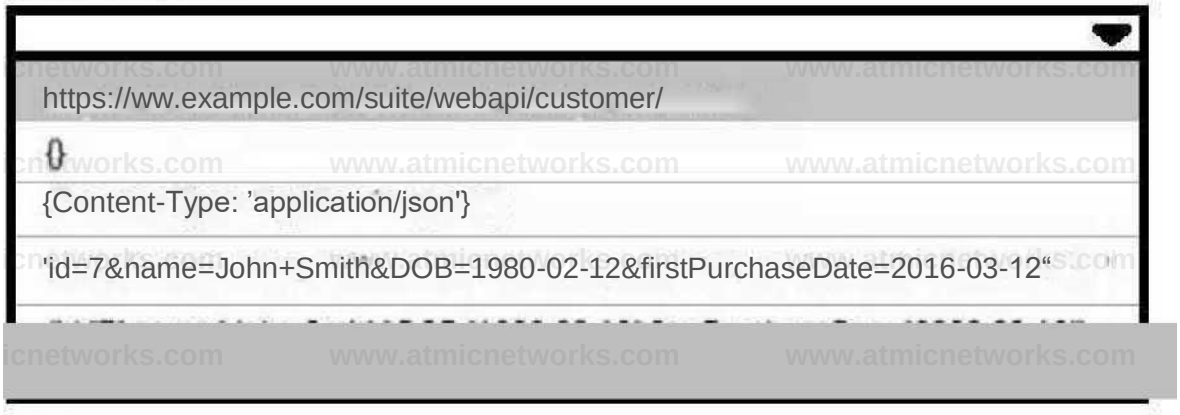
forniData

Select a match:



url

Select a match:



Question: 47

A seamless user experience (UX) is important for an application so that a user does not feel like they are moving through different applications. Often when many developers are working on the same project all together, this can be a concern.

How should you ensure that a seamless UX is maintained throughout the app?

A. Create and add reusable interface components to the Design Library so that developers can quickly and easily use reusable interfaces.

B. Allow each developer to copy client provided mock-ups individually.

C. Create reusable expression rules to ensure that the same rules are being used throughout.

Answer: A

Explanation:

Creating and adding reusable interface components to the Design Library ensures consistency in look and feel across the application, allowing all developers to easily implement a seamless user experience.

Question: 48

You need to implement a Gender dropdown, where the choice labels may need to be altered in the future. The transactional database tables rely on this data for reporting, so the data must be accurate.

What is the recommended action?

A. Create these entries in a table and reference them through a query entity rule as per the business requirement.

B. Create a constant and modify it as needed per the business requirement.

C. Create an expression rule and modify it as per the business requirement.

Answer: A

Explanation:

Storing the dropdown entries in a database table and referencing them through a query entity rule ensures data integrity, flexibility for future changes, and accuracy for reporting purposes.

Question: 49

You're redesigning user interfaces to achieve more efficient space usage.

Which layout component is the best choice for displaying an interface with a vertical meaningful order of components?

A. a!sideBySideLayout()

B. a!cardLayout()

C. a!columnsLayout()

Answer: C

Explanation:

a!columnsLayout() is best for organizing and displaying components in a vertical order with efficient space usage, as it supports arranging items in structured, responsive columns.

Question: 50

You need to configure your process model to store the result of an expression rule for every item in a list. Your design must have the smallest memory footprint.

What should you do?

A. Invoke a sub-process for each item in the list.

B. Configure a script task to use a!forEach() to iterate over each item in the list.

C. Create a loop of smart service nodes in the process model and execute it for each item in the list.

D. Execute MNI over a script task for each item in the list.

Answer: B

Explanation:

Configuring a script task to use a!forEach() to iterate over each item in the list allows you to process all items within a single task, minimizing the memory footprint compared to multiple subprocesses or smart service loops.

Question: 51

An application consists of a record type with sync enabled. This record type is used as a source to create a read-only grid.

A field "Number of days since created" is configured in the interface and a logic is implemented to calculate the number of days since a record is created in the application.

You need to implement sorting on this field.

What should you do?

A. Sorting is not possible on custom record type fields. Instead, create a view with a custom field and use the field from the view for sorting.

B. Create a custom record type field and use it in the sorting configuration.

C. Add sortField: "Number of days since created" configuration for the gridColumn that needs to be sorted.

Answer: B

Explanation:

Creating a custom record type field to calculate "Number of days since created" allows you to use this field in the sorting configuration of your read-only grid.

Question: 52

Users are experiencing slow load times when retrieving customer records with many transactions.

After using performance monitoring tools, you discover that the following interface definition is

responsible for the majority of page load time:

INTERFACE DEFINITION

```
1  a!localVariables(
2    local!pagingInfo: a!pagingInfo(
3      startIndex: 1,
4      batchSize: 25
5    ),
6    local!tableData: rule!APP_GetTransactionWithFilters(
7      pagingInfo: local!pagingInfo,
8      customerId: ri!customerId
9    ),
10   a!gridField(
11     data: local!tableData.data,
12     pagingSaveInto: local!pagingInfo,
13     columns: {
14       a!gridColumn(
15         /* Show the transaction Id */
16         value: fv!row.transactionId
17       ),
18       a!gridColumn(
19         /* Display the transaction date */
20         value: fv!row.transactionDate
21       ),
22       a!gridColumn(
23         /* Get the transaction type from the database */
24         value: rule!APP_GetTransactionTypeWithFilters(
25           transactionType: fv!row.transactionTypeId,
26           pagingInfo: a!pagingInfo(
27             startIndex: 1,
28             batchSize: 1
29           )
30         ).data.transactionTypeName
31       )
32     }
33   )
34 )
```

Additionally, both queries (rule!APP_GetTransactionsWithFilters and rule!APP_GetTransactionTypeWithFilters) take approximately 25 milliseconds each to execute when you test them using the expression editor.

Which action would result in the greatest reduction in the load time for this interface component?

A. Use a synced record for Transactions to improve the query response time for the query performed on line 6.

B. Prefetch transaction types and use the displayvalue() function to display the Transaction Type for each transaction.

C. Avoid fetching the total count when getting transactions.

Answer: B

Explanation:

Prefetching all transaction types once (instead of running a separate query for each row in the grid) and then using the displayvalue() function to display the corresponding transaction type name for each transaction will significantly reduce the number of queries executed. This eliminates the N+1 query problem and will greatly improve the interface load time.

Question: 53

DRAG DROP

You're executing a performance test of the client's application. You notice that server-side resources are at high-to-critical operating levels.

What should you do to troubleshoot this issue?

Note: To answer, move all steps from the Options list to the Answer List area and arrange them in the correct order.

Answer:

Explanation:

Question: 54

You need to create a process workflow for employee background investigations. This workflow takes an average of six months from start to completion. There are multiple phases to the background investigation which must be completed by people in different departments. Each department is responsible for inputting a large amount of data as part of the workflow.

There is an emphasis on process memory management since thousands of these investigations will happen concurrently.

What are two best practices that you should employ when designing this process workflow? (Choose

two.)

A. Utilize a record-backed design throughout the workflow where data entered by users is saved to and queried from the database.

B. Combine all stages of the workflow into a single process model, so that data can be efficiently shared between nodes using process variables.

C. Divide the workflow into multiple process models so that each process can be short-lived and consume less memory.

D. Ensure that the data management policy is set to archive the process instances after a minimum of six months, so that data is available throughout the workflow.

Answer: A, C

Explanation:

Using a record-backed design ensures data is persisted in the database, not held in memory, optimizing process memory management.

Dividing the workflow into multiple process models allows each to be short-lived, further reducing memory usage and improving scalability.

Question: 55

You're conducting a code review and notice that some accessibility features are missing on the interfaces.

Which two UX best practices should be followed to align with accessibility requirements? (Choose two.)

A. Use a single rich text icon with "POSITIVE" or "NEGATIVE" styling to indicate statuses.

B. Apply a label value for all fields when the label position is "Hidden".

C. Indicate the hierarchy of sections with a heading tag in addition to label sizes (for example: "H1").

D. Use rich text items to act as headers for structuring interfaces.

Answer: B, C

Explanation:

Applying a label value for all fields, even when the label position is hidden, ensures screen readers and assistive

technologies can interpret the field correctly.

Indicating the hierarchy of sections with heading tags improves navigation for users with assistive technologies, aligning with accessibility standards.

Question: 56

You're building a new application to track job applications for a local health club.

The owner wants to have one group of individuals perform the initial application review for completeness, a second group to manage the interview process, and a third group to make the actual hiring decisions.

Which design element should be implemented to best support a memory efficient application?

A. Create a synchronous process that accurately tracks a candidate's application from start to finish to ensure the hiring group has access to all the information necessary to make a decision.

B. Break up the application into subprocesses representing each of the different stages of the process to reduce memory use.

C. Create an entity-backed record for each of the open positions.

Answer: B

Explanation:

Breaking up the application into subprocesses for each stage allows each part to be short-lived and releases memory as each subprocess completes, resulting in a more memory-efficient application.

Question: 57

You're reviewing the process model built by your team member.

Which two design decisions might cause performance issues? (Choose two.)

A. Asynchronous subprocess

B. No target process for the Send Message event

C. Data types passed by reference

D. Too many nodes

Answer: B, D

Explanation:

Having no target process for the Send Message event can cause unnecessary processing and delays.

Too many nodes in a process model increase complexity and processing overhead, leading to potential performance issues.

Question: 58

An energy company wants to allow the public to be able to report power outages directly and provide supporting documentation in the form of photos and/or videos. The company also requires the reporter's contact details.

How can a developer design and implement this requirement in Appian?

A. A portal that interacts with records that stores the name, address, contact information, with a defined folder to store uploaded documents.

B. A custom version of the Appian mobile app that the user can use to capture details, attached photos, and upload documents.

C. A site where the user can trigger a process that captures the information including any uploaded files, and uploads everything to an internal Microsoft SharePoint site.

Answer: A

Explanation:

A portal interacting with records can capture public outage reports, contact details, and upload supporting documents to a defined folder, making the process accessible to the public without requiring authentication.

Question: 59

You need to create an expression rule that will be reused throughout your environment.

What are two reasons why you should include meaningful test cases when creating a new expression rule? (Choose two.)

A. To improve the performance of the environment.

B. To accelerate various types of testing, including: unit, regression, exploratory.

C. To facilitate Test-Driven Development.

D. To enhance the appearance of the code.

Answer: B, C

Explanation:

Meaningful test cases accelerate different testing types by providing clear validation scenarios for the rule.

They support Test-Driven Development by enabling you to define expected outcomes before or during rule creation.

Question: 60

You need to allow external systems to invoke Appian web APIs.

Which user role should be assigned?

A. System Administrator

B. Service Account

C. Application User

Answer: B

Explanation:

A Service Account user role is specifically designed for integrations, allowing external systems to securely authenticate and invoke Appian web APIs.

Question: 61

You run a report on different employee transactions by using a View.

You encounter the following error: "a!queryEntity: An error occurred while retrieving the data."

What is the most likely root cause of this error?

A. The view doesn't have a column mapped as a primary key in its corresponding CDT.

B. The view contains a large number of rows, requiring more time to fetch the data.

C. The rule contains a missing syntax.

Answer: A

Explanation:

The most likely root cause is that the view doesn't have a column mapped as a primary key in its corresponding CDT. Appian requires a primary key mapping for querying views to ensure data integrity and identification.

Question: 62

You receive a warning about approaching the 2,000,000 row limit with synced records in your Production environment.

What are two ways to efficiently address this warning to prevent reaching the limit? (Choose two.)

A. Design the process to write data in batches of 1,000.

B. Disable data sync immediately to prevent user facing-front end issues.

C. Keep only necessary data and plan to move the archived data to an audit table in a longer run.

D. Apply one or more source filters to limit the number of rows synced in Agpian.

Answer: C, D

Explanation:

Keeping only necessary data and moving archived data to an audit table reduces the number of synced rows, addressing the limit efficiently.

Applying source filters limits the data that is synced, preventing the total row count from exceeding the limit.

Question: 63

You're designing a custom component rule!pagingComponent() to page through a list of items. The component must display the current page numbers being shown, as well as the total number of items. Users should be able to click arrows to increase or decrease the pagination.

Review the following:

```
al localvariables(
local!query: rule!getData(),

rule!interface(), rule!pagingcomponent()
```

Which two patterns should be used when calling the component in this scenario? (Choose two.)

A. Passing into the rule input totalCount of rule!pagingComponent() the value: local!query.totalCount where local!query is returning a dataSubset and returnTotalCount = "True".

B. A rule input ri!pagingInfo on rule!pagingComponent() with a local variable of the same name in the main interface with an initial value. When the arrows are clicked, this updates the value of the local variable in the main interface.

C. A local variable local!totalCount calculated using count(local!query), which is passed into rule!pagingComponent() via a rule input. The data type of local!query is a list of record or Custom Data Type (CDT).

D. A local variable local!pagingInfo inside rule!pagingComponent() with an initial value. When the arrows are clicked, this updates the value of the local variable.

Answer: A, B

Explanation:

Passing local!query.totalCount (from a dataSubset with returnTotalCount: true) into a rule input allows rule!pagingComponent() to accurately display total items for pagination.

Managing paging state (such as ri!pagingInfo) as a local variable in the main interface and updating it when arrows are clicked ensures that pagination is controlled centrally, maintaining state across the UI and allowing the component to

function as intended.

Question: 64

You're creating a process model that will serve as the case creation workflow.

Your requirements:

This process captures input from the user before executing a series of unattended subprocess nodes.

Each subprocess performs extended data processing which is dependent on output from each of the previous nodes.

An activity chained task node is displayed asking the user if they'd like to create another case.

What should you do?

A. Combine the separate subprocesses into one synchronous subprocess and place it in between each user input task.

B. Configure the subprocess nodes to run asynchronously and set the inputs to pass as reference to enable data sharing.

C. Add an AND gate after the first user input task and place the synchronous subprocess nodes on a parallel path.

Answer: A

Explanation:

Combining the separate subprocesses into one synchronous subprocess and placing it between each user input task ensures each subprocess's output is available for the next step, maintaining data dependencies and supporting the required user experience with activity chaining.

Question: 65

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. calworkdays()

B. a!subtractDateTime()

C. a!customFieldDateDiff()

Answer: C

Explanation:

a!customFieldDateDiff() is specifically designed for calculating date differences as custom record fields, enabling sorting and real-time calculation in read-only grids.

Question: 66

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. Add the "ACME Supervisors" group as a member of the "Database Editors" group and grant DELETE privileges.

C. Update the record action security for "Delete Case" and include the "ACME Supervisors" group within the visibility settings.

D. Add the "ACME Supervisors" group as a Viewer to the data store mapped to the record.

Answer: B, C

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.

Question: 67

You need to assign a user input task to three different groups of users: Group A, Group B, and Group C.

Each task form shares some common components with the following key differences:

Group B will receive a task that includes an additional section.

Group C will only view and enter data for one section.

What should you do?

A. Within the process model, utilize an XOR gateway to conditionally select between the three different user input tasks.

B. Configure the version setting on the user input task node to conditionally display the correct form.

C. Utilize conditional logic on the form along with the showWhen parameter for interface components.

Answer: C

Explanation:

Using conditional logic with the showWhen parameter on the form allows you to display or hide specific sections for each group within a single user input task, making the solution flexible and maintainable.

Question: 68

Which section of the Interface Performance View explains the contribution towards the overall evaluation time of the current function, rule, or parameter?

A. Parameters and Direct Children

B. Precedent Functions and Rules

C. Descendant Rules

Answer: A

Explanation:

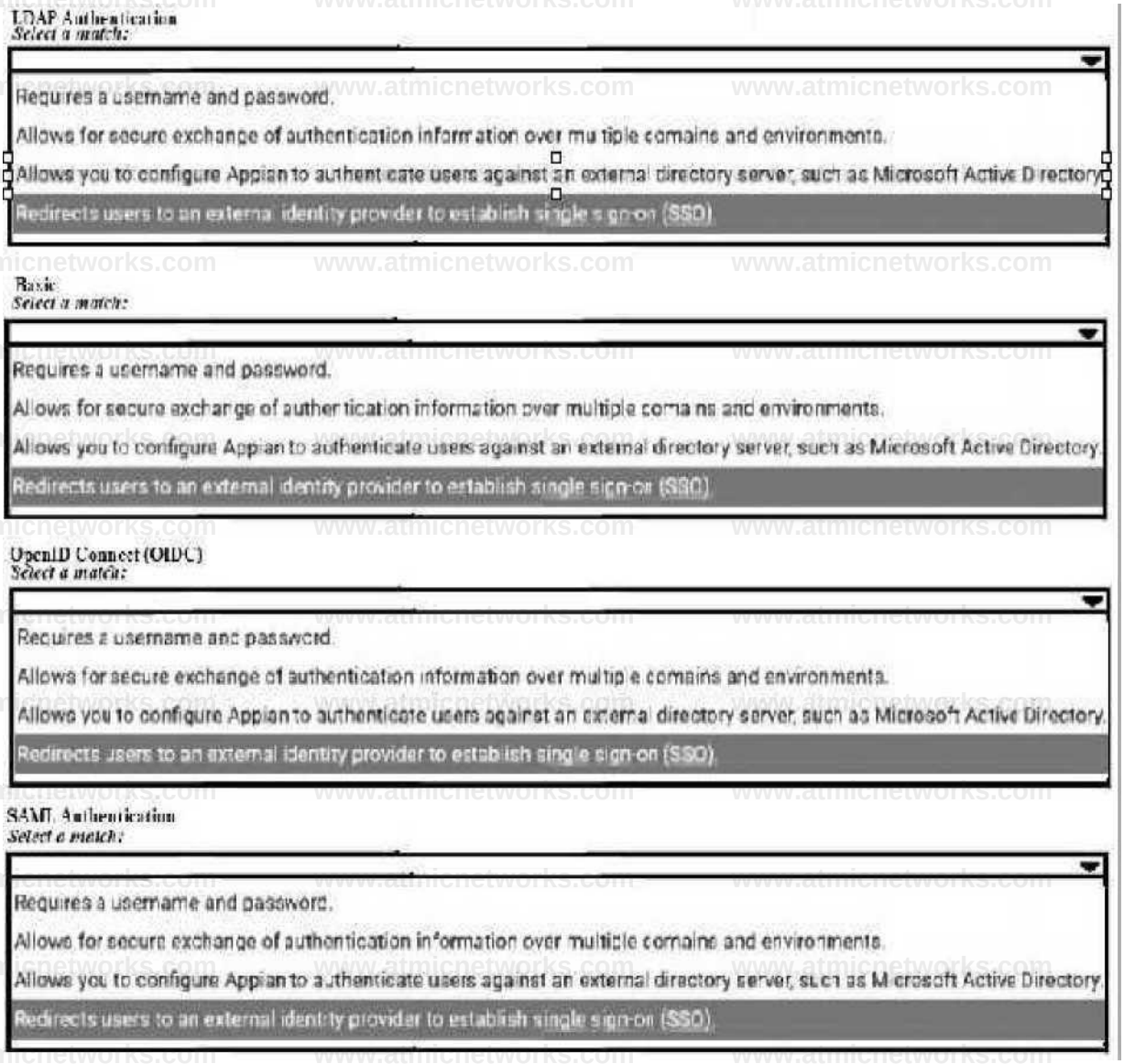
The "Parameters and Direct Children" section of the Interface Performance View explains the contribution towards the overall evaluation time of the current function, rule, or parameter.

Question: 69

HOTSPOT

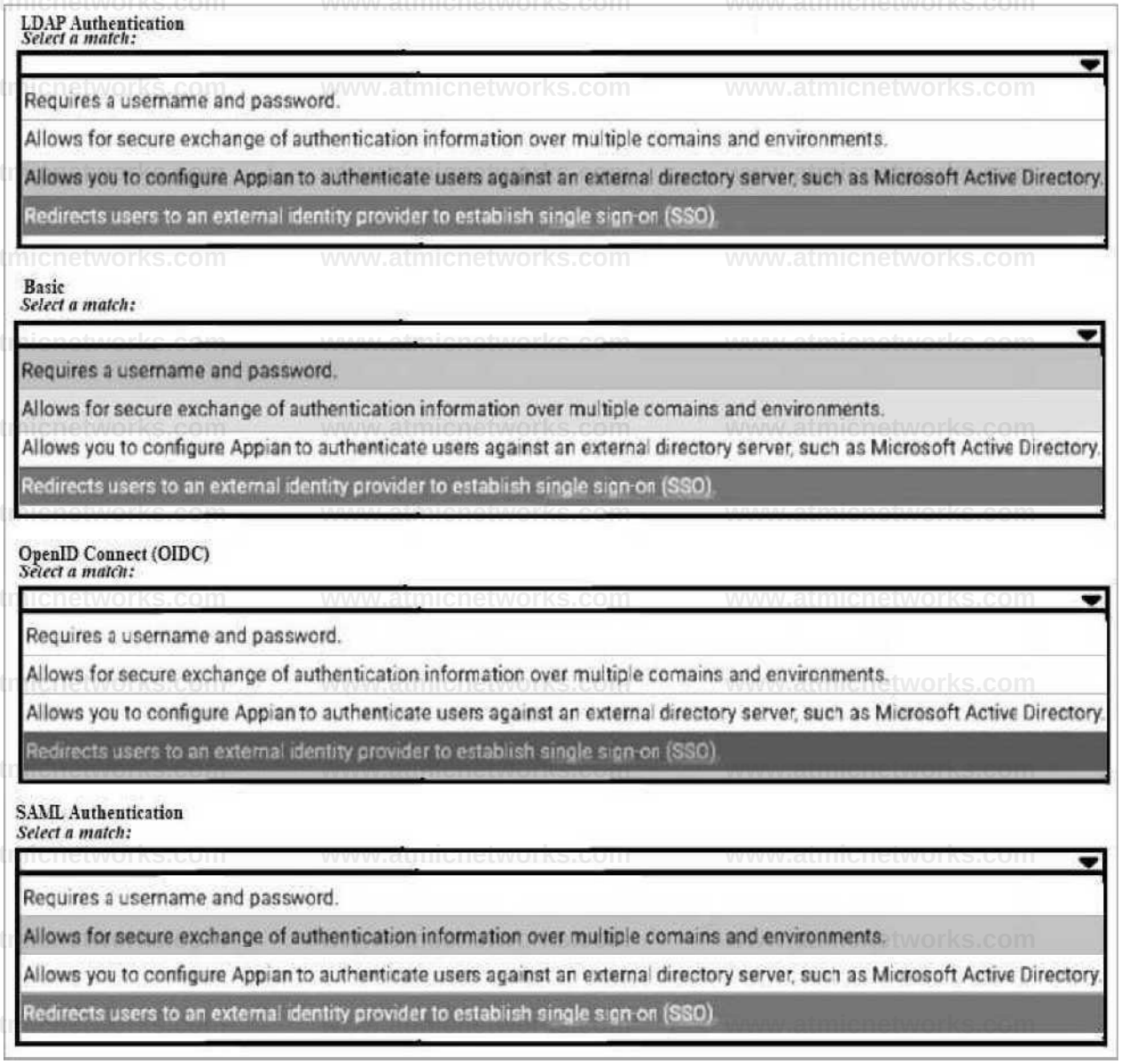
Match each authentication type to the correct authentication characteristic description.

Note: Each description will be used once. To change your responses, you may deselect your response by clicking the blank space at the top of the selection list.



Answer:

Explanation:



Question: 70

A view is created with the primary table PERSON and is joined using a left join with the APPOINTMENT table on the column "PERSON_ID". There is a one-to-many relationship between the PERSON and APPOINTMENT tables.

A Custom Data Type (CDT) and a data store entity have been created based on this view, with the primary key being "PERSON_ID".

Review the data presented in the PERSON and APPOINTMENT tables:

PERSON_ID	PERSON_NAME
1101	John Doe
1102	Jane Doe
1103	Richard Doe

APPOINTMENTJD	PERSONJD	STATUS
428	1101	Completed
429	1102	Completed
430	1101	New
431	1103	Completed
432	1101	New
433	1102	New
434	1101	New

Which two statements are true when the following query entity is executed? (Choose two.)

```
a!queryEntity(
   entity: cons!ABC_PFRSON_APPOINTMFNT_VIEW_DSF, query: a l query(
      logicalExpression: a 1 queryLogicalExpression( operator: “AND", filters: {
            a!queryFilter( field: "STATUS", operator: “=", value: "New"

            alqueryFilter(
               field: "PERSON ID", operator: value: "1101"

                                                    }
                                                ),

      paginginfo: a!pagingInfo(
         startindex: 1,
         batchsize: 1, sort: alsortInfo(field: "PERSON ID")

   fetchTotalCount: true()
```

A. Appian makes one more query to the database(other than the actual query to fetch the data) to get the totalCount based on the applied filters.

B. result totalCount is 3

C. result totalCount is 1

D. Appian makes one more query to the database(other than the actual query to fetch the data) to get the totalCount based on the applied filters and paging info.

Answer: A, C

Explanation:

Appian makes an additional query to the database (other than the main query for data) to retrieve the totalCount when fetchTotalCount: true() is used.

The result totalCount is 1, as there is only one person (PERSON_ID 1101) being filtered, even though there are multiple "New" appointments for that person; with a CDT based on PERSON_ID as the primary key, only one row is counted per person.

Question: 71

You're reviewing the data store performance logs. You notice several items were logged in the generated slow query log file: perf_monitor_rdbms_slow.csv.

What is the default threshold value for a slow query to be included in this log file?

A. 5 sec

B. 7 sec

C. 3 sec

Answer: C

Explanation:

The default threshold value for a slow query to be included in the perf_monitor_rdbms_slow.csv log file is 3 seconds.

Question: 72

You're creating an interface object that displays a report of open cases. This interface will be used as the content of a site page.

Users in the "ACME Employees" group must be able to view the report.

Which two steps should you perform? (Choose two.)

A. Ensure that the site page visibility permits the "ACME Employees" group to view the page.

B. Add the "ACME Employees" group as a viewer on the site object.

C. Include the "ACME Employees" group as a member of the "Report Viewers" system group.

D. Add the "ACME Employees" group as a viewer on the folder containing the interface object.

Answer: A, D

Explanation:

The site page visibility must permit the "ACME Employees" group to ensure they can access the site page with the report.

The "ACME Employees" group must be added as a viewer on the folder containing the interface object to grant them access to the report interface.

Question: 73

What are two advantages of database normalization? (Choose two.)

A. Enforcing data integrity.

B. Simplifying query statements to read data.

C. Minimizing the number of tables that need to be created.

D. Minimizing data redundancy.

Answer: A, D

Explanation:

Database normalization enforces data integrity by structuring data and relationships to prevent anomalies.

It minimizes data redundancy by ensuring each piece of information is stored only once, reducing duplication.

Question: 74

You're refining a story regarding a highly regulated form, where information needs to be sent to a regulatory body at the end of each month to ensure that the business complies with laws and legislation. The story only concerns the creation of the form.

You need to create a form that captures information to comply with the regulatory requirements.

Which two questions should you ask the business? (Choose two.)

A. What information does the regulatory body require?

B. How often is the information sent to the regulatory body?

C. Is the information sent to the regulatory body via an integration?

D. What is the maximum length of each field?

Answer: A, D

Explanation:

Knowing what information the regulatory body requires ensures the form captures all necessary data for compliance.

Understanding the maximum length of each field guarantees that the data collected will meet validation and regulatory constraints.

Question: 75

You're creating an interface to display all the employees of a company on a staff list.

You decide to use a!gridField() to create a user grid to display information from the company's AAA_Employee record, which has 42 fields relating to various employee information. To avoid cluttering the grid, you want to show six (6) fields on the grid.

How should you optimize the interface?

A. Query the AAA_Employee record using a!queryRecordType(), defining the six (6) fields you require to show in the "fields" parameterto be used in the data parameter of a!gridFieid().

B. Using a local variable, query the AAA_Employee record using a!queryRecordType(), without defining the six (6) fields you require to show in the "fields" parameter to be used in the data parameter of a!gridFieid().

C. Call the function a!forEach() in a local variable and query the AAA_Employee for each employee separately by using a!queryRecordType() to loop through all employees as defined by a constant to be used in the data parameter of a!gridFieid().

Answer: A

Explanation:

Querying the AAA_Employee record with a!queryRecordType(), specifying only the six fields needed in the fields parameter, optimizes data retrieval and improves grid performance by minimizing unnecessary data loading.

Question: 76

You're conducting a design review.

You identify slow-performing expression rules querying a specific data store, and need to understand the "number of operations against data stores."

Which metric from the data_store_details.csv file is helpful to you?

A. Execute Count

B. Query Count

C. TCCCotal Count

Answer: B

Explanation:

The "Query Count" metric in the data_store_details.csv file indicates the number of operations (queries) performed against data stores, helping you analyze data store usage.

Question: 77

You have a Case record type with data sync enabled to retrieve submitted support cases.

What is the best approach to limit who can see which records in the record type?

A. Security rules

B. Default filters

C. Record-level security

Answer: C

Explanation:

Record-level security is the best approach for limiting which users can see specific records in a record type with data sync enabled, as it allows you to define access at the record level.

Question: 78

You need to test a related action that updates record data.

Appropriate users must be able to access the form. You also need to ensure that the data is successfully updated.

Using Appian, which is the best testing method you should use?

A. User Accepting Testing

B. Unit Testing

C. Performance Testing

Answer: A

Explanation:

User Acceptance Testing (UAT) is the best method, as it ensures appropriate users can access the form and verifies that the data update process works as intended in a real-world scenario.

Question: 79

Your client wants to configure user authentication using single sign-on (SSO) in their cloud environment.

Which two authentication types should you use to support this requirement? (Choose two.)

A. PIEE user authentication

B. LDAP authentication

C. OpenID Connect authentication

D. Appian authentication

Answer: C, D

Explanation:

OpenID Connect authentication supports single sign-on (SSO) by integrating with external identity providers.

Appian authentication can work in conjunction with SSO configurations as part of a hybrid setup or fallback mechanism.

Question: 80

You need to implement a requirement where a third-party system starts a process in Appian. The third-party system can invoke a service only through Web Services Description Language (WSDL).

What should you do to start the process in Appian?

A. Create a default WSDL URL using process model UUID.

B. Create a custom plug-in.

C. Expose process model as a web service.

Answer: C

Explanation:

To allow a third-party system to start a process in Appian using WSDL, you should expose the process model as a web service. This generates a WSDL endpoint that the external system can invoke.

Question: 81

You're configuring Health Check settings under the Admin Console.

Which section would you find the setting to indicate that the Health Check will run on a Production Environment?

A. General

B. Scheduling

C. Automatic Upload

Answer: C

Explanation:

The Automatic Upload section in the Health Check settings of the Admin Console includes the option to indicate that the Health Check is running on a Production Environment, which helps Appian appropriately analyze and benchmark the data.

Question: 82

An insurance company's product details are currently distributed across numerous database tables, encompassing over 60 to 70 fields.

You need to implement a product comparison tool for business users to compare up to three insurance products simultaneously, with all the product details in a tabular format.

Refer to the following sample format:

Attribute	PRODUCT 1	PRODUCT2	PRODUCTS
Name			
Type			
Premium			
Average			
Claim time			

Which object should be used to transform the data into the required format?

A. Stored procedure

B. Record type with relationships

C. VCCCiews

Answer: C

Explanation:

Database views are ideal for transforming complex, distributed product data into a single, flattened structure tailored for reporting or comparison. In this case, a view can pivot the 60–70 fields and organize them by attribute (e.g., Name, Type, Premium) across multiple product columns, enabling a tabular comparison layout like the one shown. Views also optimize performance and reduce transformation overhead in the interface layer.

Question: 83

You're designing a file upload process that integrates with an external document management system.

Part of this process is to develop a start form with a file upload component, so that the user can upload multiple files.

The business has requested that the maximum file size that can be uploaded through this process is 1GB. Additionally, you're aware that the integration's API can accept files with a maximum file size of 3GB, but recommends no larger than 1GB.

Can the process be more performant without straining the application or integration?

A. Yes. Request the business allows for these files to be uploaded as an overnight process by storing large files in a staging table, so that adequate resources are available.

B. No. The file upload component is capable of accepting 1 GB files and the integration can handle this.

C. Yes. Request the business reconsiders its requirements and find manual workarounds as it is not recommended to upload files of this size in both Appian and the integration.

Answer: A

Explanation:

While Appian and the integration technically support 1 GB file uploads, handling such large files during interactive sessions can negatively impact application performance, user experience, and resource utilization.

A better approach is to request that the business allow these files to be uploaded asynchronously (e.g., overnight batch process), by temporarily storing them (e.g., in a staging table or folder) and processing them when system load is lower. This makes the process more performant and reliable without straining Appian or the external system.

Question: 84

You need to select a plug-in to meet a specific requirement and several plug-ins can do the job.

What are three key factors to consider? (Choose three.)

A. What are the limitations of the plug-in?

B. What are the compatible Appian versions?

C. Who is the creator of the plug-in?

D. Is the plug-in approved for use in the Cloud?

E. Is the plug-in open source or proprietary?

F. What is the size of the plug-in?

Answer: A, B, D

Explanation:

Limitations of the plug-in help you understand constraints or potential issues that could impact your implementation.

Compatible Appian versions ensure the plug-in will work correctly in your current and future environments.

Cloud approval status is essential for Appian Cloud environments, as only approved plug-ins are permitted for deployment.

Question: 85

Your client wants their customers to be able to schedule appointments directly from their website without going to an Appian user environment.

You need to build an embedded interface to be added to your client's web page. The style of the embedded interface should be consistent with the host web page.

Which three custom styling options can be configured in the Themes section of embedded interfaces? (Choose three.)

A. Two or three column layout

B. Page background color

C. Asymmetrical layout

D. Font Family

E. Web font stylesheet URL

Answer: B, D, E

Explanation:

Page background color - You can customize the background color to match the host webpage's theme.

Font family - You can set the font family to ensure the embedded interface aligns with the client's branding.

Web font stylesheet URL - This allows the use of custom web fonts, further aligning the style with the host site.

Question: 86

You're making a POST request to the Appian web API. You need to include complex data structures, such as lists or objects, in the request body.

How should you design the API to accept those requests?

A. Use query parameters to represent the complex data structures.

B. Use XML format to represent the complex data structures in the request body.

C. Convert the complex data structures to JSON and include them in the request body as a string.

Answer: C

Explanation:

To include complex data structures like lists or objects in a POST request to an Appian Web API, you should convert the data to JSON and include it in the request body as a string. Appian supports application/json content type for parsing complex data structures in APIs.

Question: 87

You need to relate two entities, Employee and Skill, in the data structure. You want to adhere to Appian best practices.

Employees can have multiple skills, and a single skill can relate to multiple employees.

What type of relationship do these entities represent, and what is the minimum number of tables required to implement the design?

A. Many-to-many; 3 tables

B. Many-to-many; 2 tables

C. One-to-many; 2 tables

Answer: A

Explanation:

The relationship between Employee and Skill is many-to-many - an employee can have multiple skills, and each skill can belong to multiple employees.

To model this properly and according to Appian best practices, you need 3 tables:

- Employee table
- Skill table
- EmployeeSkill (junction/association) table - contains foreign keys referencing both Employee and Skill to manage the many-to-many relationship.

Question: 88

HOTSPOT

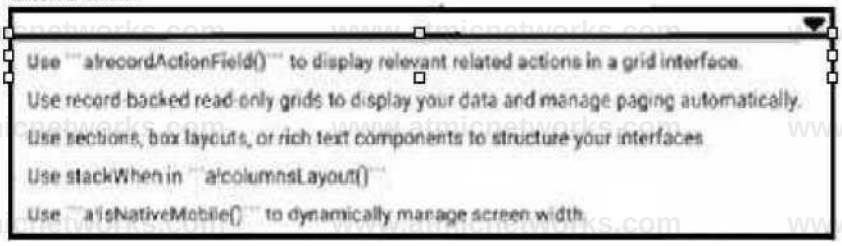
You're developing a case management application. Currently, users can view a list of all cases designed using a a!forEach() loop from a record query. Users can navigate to the case summary page which shows the case details in a two (2) column layout. Users are also able to update the case details.

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

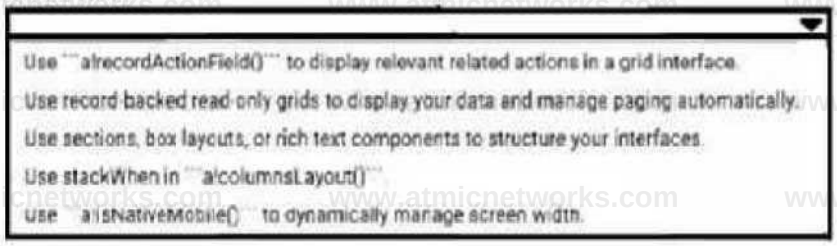
Match each feedback comment to the suggested UI/UX improvements.

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.

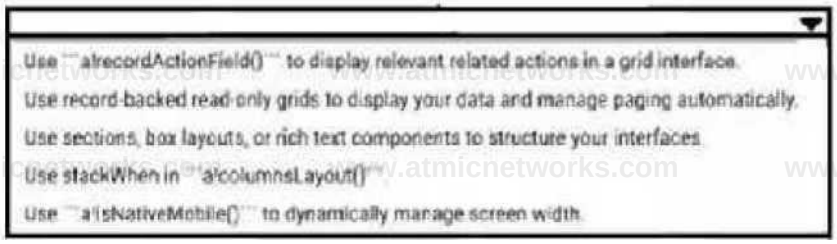
"Updating a case is time-consuming because I have to search the case list, select a case, go to 'Related Actions,' and find the correct action."
Select a match:



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



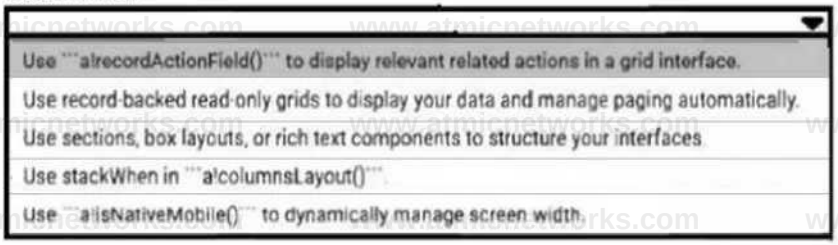
"When I access the application on my mobile device, the summary page is difficult to read as the columns become too narrow."
Select a match:



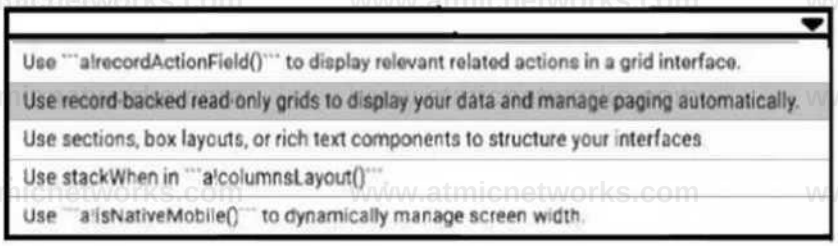
Answer:

Explanation:

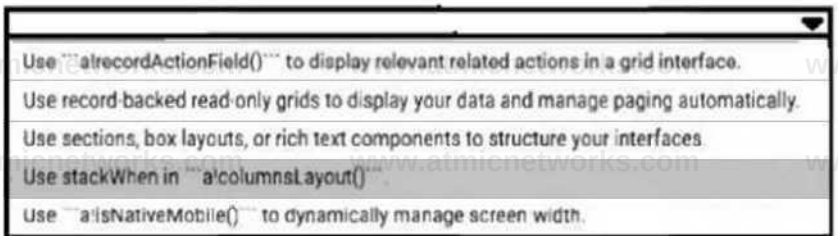
"Updating a case is time-consuming because I have to search the case list, select a case, go to 'Related Actions,' and find the correct action."
Select a match:



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



"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:



Question: 89

You're performing System Integration Testing on a client's application that integrates with an external system.

In this instance, you're testing the application's interactions with external integrations to ensure that data is being handled correctly between the two systems. However, this step has been done manually and is time-consuming.

What should you do to make this process more efficient?

A. Utilize the Postman API Platform to test the integrations with the external system and Appian's web APIs through a defined set of test cases.

B. Utilize Appian-Locust to develop and execute performance tests on the systems integrations by scripting a scenario where multiple users are interacting with the external systems simultaneously.

C. Utilize FitNesse for Appian to perform automated functional testing on the workflows that include

integrated systems through a defined set of test cases.

Answer: C

Explanation:

FitNesse for Appian is designed for automated functional testing of Appian workflows, including those that call external integrations. By building repeatable FitNesse test scripts that drive the end- to-end process, you can automatically verify that data is correctly sent to - and returned from - the external system each time, greatly reducing the manual effort of system-integration testing.

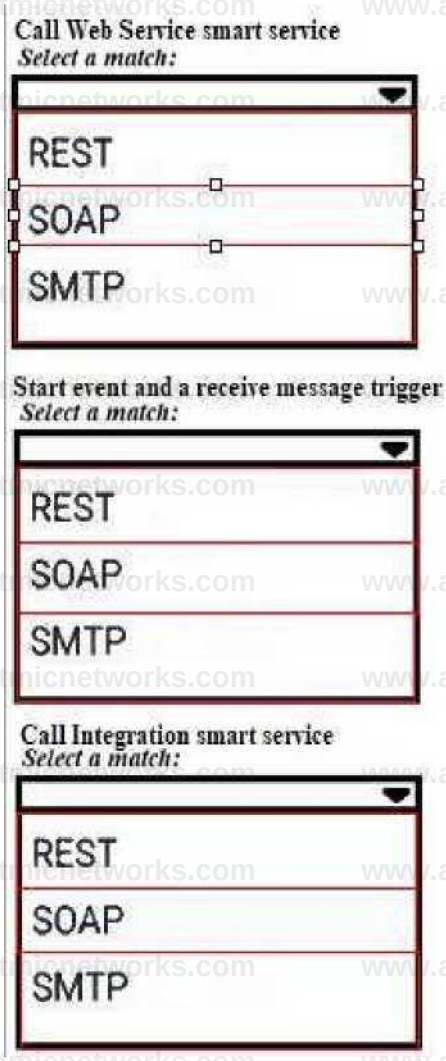
Question: 90

HOTSPOT

You're creating an application which has integrations with multiple systems using different protocols.

Match each process model component or node to the appropriate protocol.

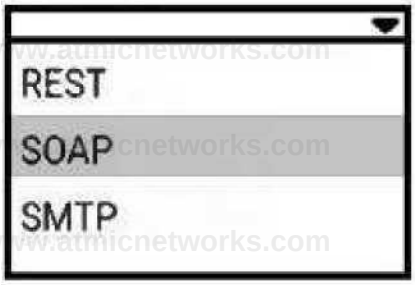
Note: Each protocol will be used once. To change your responses, you may deselect your response by clicking the blank space at the top of the selection list.



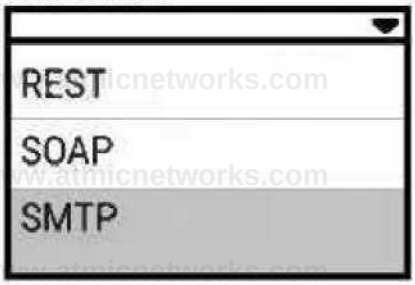
Answer:

Explanation:

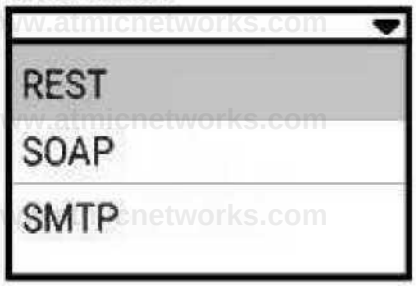
Call Web Service smart service
Select a match:



Start event and a receive message trigger
Select a match:



Call Integration smart service
Select a match:



Question: 91

You have set up SAML for single sign-on (SSO) in your cloud environment.

Which three options could you enable in the Attribute Mapping section of the Admin Console to sync with a user's profile in Appian? (Choose three.)

A. "Create new users upon sign in."

B. "Group Membership Synchronization."

C. "Remember Me on web and mobile.'

D. "Reactivate deactivated users upon sign in."

E. "Update user attributes upon sign in."

Answer: A, D, E

Explanation:

Create new users upon sign in - provisions new Appian users automatically when they first authenticate through SAML.

Reactivate deactivated users upon sign in - automatically re-enables an Appian account if the user returns after being deactivated.

Update user attributes upon sign in - refreshes profile details (e.g., name, email, custom attributes) each time the user signs in.

Question: 92

An insurance application has a dashboard in which all the cases with "Accident" case type and "Health Insurance" insurance type are displayed to the user, using the following query:

```
a! query(
    logical Expression: alqueryLogicalExpression( operator: "AND*,
      filters: { alqueryFilter(
          field: "casetype", operator: "=", value: "Accident"
        alqueryFilter(
          rield: "insuranceType", operator:
          value: "Health Insurance"
        ) }
    paginginfo: a!paginginfo( startindex: 1, batchsize: 10, sort: al sortinfo/
        field: "createoDate"
```

You need to update the query entity to display all the cases that are "Accident" case type and "Health Insurance" insurance type, or created in the last ten days.

What is the right configuration for the logical expression?

A)

```
a lqueryLogkilExpression(
        operator: AND“, filters: {
          a l query?ilter( field: "caserype", operator: "-",
             value: "Accident"

          a!queryFilter(
             field: "insuranceType",
          ).operator: value: "Health Insurance"

        logicalExpressions: alqueryLogicalExpression(
        operator: "OR", filters: alqueryFilter{ field:
        “createdDate", value: todayQ - 10, operator: ">■
        },
      )
    )
  )
```

B)

```
a!queryLogicalExpression(
      operator: "OR",
      filters: a!queryFilter(
        field: "createdDate",
        value: today() - 10,
        operator: ">="
      ),
      logicalExpressions: a!queryLogicalExpression(
        operator: "AND",
        filters: {
          a!queryFilter(
            field: "caseType",
            operator: "=",
            value: "Accident"
          ),
          a!queryFilter(
            field: "insuranceType",
            operator: "=",
            value: "Health Insurance"
          )
        }
      )
    )
```

C)

```
a! queryLo£kalExpression( operator: AND", filters: {
        alqueryFilter( field: "caseType", operator: "e", value: "Accident"

        alqueryFilter(
          field: "insuranceType", operator: "=".
          value: "Health Insurance

        a!queryFilter(
          field: "createdDate",
          value: todayQ 10, operator: ">=
        )
```

A. Option A

B. Option B

C. Option C

Answer: B

Explanation:

A filter for createdDate >= today() – 10 (records created in the last 10 days).

A nested AND logical expression that requires both caseType = "Accident" and insuranceType = "Health Insurance".

Question: 93

There are two record types, ABC and XYZ, with sync enabled. The XYZ record type is added as a relationship into the ABC record type.

A user has Viewer permission to the ABC record type but does not have access to the XYZ record type.

A site page is presented to the user where the data is sourced from the ABC record type and its related record type the XYZ reference.

What information does the user see on the site page?

A. Page is presented to the user and the data references to the XYZ record type appear as "null".

B. Page is presented to the user with the XYZ record type data and fields references redacted.

C. Page does not load and an error message is presented: "The record type [identifier=XYZ] does not exist, has been deleted, or you do not have sufficient privileges to access its data."

Answer: C

Explanation:

Because the interface references fields on the XYZ record type, the user must have at least Viewer

permission on XYZ as well as on ABC. Since the user lacks access to XYZ, the query cannot be executed and the page fails to load, displaying the error:

"The record type [identifier = XYZ] does not exist, has been deleted, or you do not have sufficient privileges to access its data."

Question: 94

You need to connect to an external system using OAuth 2.0: SAML Bearer Assertion Flow authentication type, which requests access to an API on of a signed in user.

This standard has several steps involved with the SAML Bearer Assertion Flow.

Which two steps should you perform to connect to the external system? (Choose two.)

A. Use an a!authorizationLink() on the relevant interface.

B. Confiqure SAML for sinqle siqn-on (SSO).

C. Generate a service account key in third-party systems.

D. Register the connection in the third-party system

Answer: B, D

Explanation:

Configure SAML for single sign-on (SSO).

The bearer-assertion flow relies on an existing SAML SSO configuration so Appian can issue a signed SAML assertion on behalf of the signed-in user.

Register the connection in the third-party system.

You must register (or "trust") Appian's client/certificate with the external system so it will accept the SAML assertion and issue an OAuth 2.0 access-token for the user.

Question: 95

The synced record Task has a self-referential relationship defined in the column parentTaskId. There is a many-to- one record relationship between the id and parentTaskId called parentTask.

For a given task ID, you need to return the task name and the parent task name.

What should you do?

A. Use a!queryRecordType() With a filter on the task id, with fields specified to return recordType!Task.name and recordType!Task.parentTask.name.

B. Create a sync-time custom record field on the Task record called parentName. Specify this field to return in the query field selection.

C. Use a!queryRecordType() filtered on the task id once to return the task name and parent task ID. Query the record again to return the parent task name.

Answer: A

Explanation:

A single a!queryRecordType() call filtered by the task's id and selecting both recordType!Task.name and recordType!Task.parentTask.name leverages the existing self-referential relationship, returning the task name and its parent's name in one efficient query without additional custom fields or extra queries.

Question: 96

After reviewing the latest Expression performance logs, you notice you have several expressions contributing to slow interface or process performance.

Which two conditions will lead to a higher risk associated with slow-performing expressions? (Choose two.)

A. When the expression is used in a local variable.

B. When the expression is used frequently or by many concurrent users.

C. When the expression is evaluated on a mobile device.

D. When the expression is a custom function plug-in.

Answer: B, D