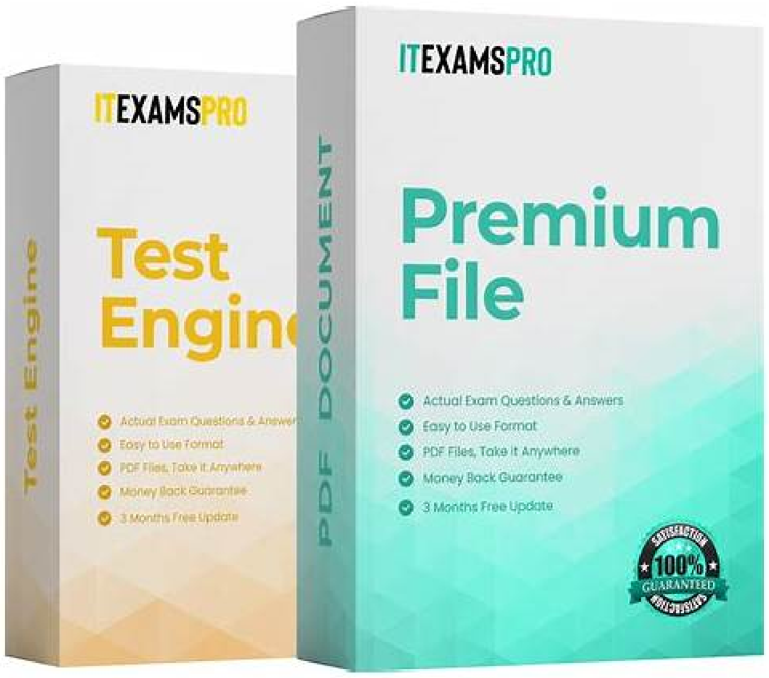
ACD301 Sample Exam - Dumps ACD301 Discount



P.S. Free & New ACD301 dumps are available on Google Drive shared by Actualtests4sure: https://drive.google.com/open?id=1xGVPHEp7yRQBROTOUJ0JnTfgE9x192Rg

The online version of our ACD301 exam questions is convenient for you if you are busy at work and traffic. Wherever you are, as long as you have an access to the internet, a smart phone or an I-pad can become your study tool for the ACD301 exam. This version can also provide you with exam simulation. And the good point is that you don't need to install any software or app. All you need is to click the link of the online ACD301 Training Material once, and then you can learn and practice offline.

Actualtests4sure is here to help of you to make your ACD301 certification dream true by providing the best valid and latest exam Appian ACD301 study reference. If you still have doubt about our ACD301 exam dumps. Please pay attention to our ACD301 free demo on the product page. You can download the free demo and have a try. Then I believe you can make the decision. Generally, there are explanations along with the questions, which will make you learn more about the knowledge about ACD301 Actual Test. Please prepare well with the ACD301 study material we provide for you. We guarantee you can pass the ACD301 actual test with a high score.

>> ACD301 Sample Exam <<

Dumps ACD301 Discount & Positive ACD301 Feedback

The web-based ACD301 practice exam software is genuine, authentic, and real so feel free to start your practice instantly with ACD301 practice test. Spend no time, otherwise, you will pass on these fantastic opportunities. Start preparing for the ACD301 Exam by purchasing the most recent Appian ACD301 exam dumps.

Appian Lead Developer Sample Questions (Q37-Q42):

NEW QUESTION # 37
You are required to configure a connection so that Jira can inform Appian when specific tickets change (using a webhook). Which three required steps will allow you to connect both systems?

- A. Give the service account system administrator privileges.
- B. Create a new API Key and associate a service account.
- C. Create a Web API object and set up the correct security.
- D. Create an integration object from Appian to Jira to periodically check the ticket status.
- E. Configure the connection in Jira specifying the URL and credentials.

Answer: B,C,E

Explanation:
Comprehensive and Detailed In-Depth Explanation:
Configuring a webhook connection from Jira to Appian requires setting up a mechanism for Jira to push ticket change notifications to Appian in real-time. This involves creating an endpoint in Appian to receive the webhook and configuring Jira to send the data. Appian's Integration Best Practices and Web API documentation provide the framework for this process.
Option A (Create a Web API object and set up the correct security):
This is a required step. In Appian, a Web API object serves as the endpoint to receive incoming webhook requests from Jira. You must define the API structure (e.g., HTTP method, input parameters) and configure security (e.g., basic authentication, API key, or OAuth) to validate incoming requests. Appian recommends using a service account with appropriate permissions to ensure secure access, aligning with the need for a controlled webhook receiver.
Option B (Configure the connection in Jira specifying the URL and credentials):
This is essential. In Jira, you need to set up a webhook by providing the Appian Web API's URL (e.g., https://<appian-site>/suite/webapi/<web-api-name>) and the credentials or authentication method (e.g., API key or basic auth) that match the security setup in Appian. This ensures Jira can successfully send ticket change events to Appian.
Option C (Create a new API Key and associate a service account):
This is necessary for secure authentication. Appian recommends using an API key tied to a service account for webhook integrations. The service account should have permissions to process the incoming data (e.g., write to a process or data store) but not excessive privileges. This step complements the Web API security setup and Jira configuration.
Option D (Give the service account system administrator privileges):
This is unnecessary and insecure. System administrator privileges grant broad access, which is overkill for a webhook integration. Appian's security best practices advocate for least-privilege principles, limiting the service account to the specific objects or actions needed (e.g., executing the Web API).
Option E (Create an integration object from Appian to Jira to periodically check the ticket status):
This is incorrect for a webhook scenario. Webhooks are push-based, where Jira notifies Appian of changes. Creating an integration object for periodic polling (pull-based) is a different approach and not required for the stated requirement of Jira informing Appian via webhook.
These three steps (A, B, C) establish a secure, functional webhook connection without introducing unnecessary complexity or security risks.
Reference:
The three required steps that will allow you to connect both systems are:
A . Create a Web API object and set up the correct security. This will allow you to define an endpoint in Appian that can receive requests from Jira via webhook. You will also need to configure the security settings for the Web API object, such as authentication method, allowed origins, and access control.
B . Configure the connection in Jira specifying the URL and credentials. This will allow you to set up a webhook in Jira that can send requests to Appian when specific tickets change. You will need to specify the URL of the Web API object in Appian, as well as any credentials required for authentication.
C . Create a new API Key and associate a service account. This will allow you to generate a unique token that can be used for authentication between Jira and Appian. You will also need to create a service account in Appian that has permissions to access or update data related to Jira tickets.
The other options are incorrect for the following reasons:
D . Give the service account system administrator privileges. This is not required and could pose a security risk, as giving system administrator privileges to a service account could allow it to perform actions that are not related to Jira tickets, such as modifying system settings or accessing sensitive data.
E . Create an integration object from Appian to Jira to periodically check the ticket status. This is not required and could cause unnecessary overhead, as creating an integration object from Appian to Jira would involve polling Jira for ticket status changes, which could consume more resources than using webhook notifications. Verified Reference: Appian Documentation, section "Web API" and "API Keys".

NEW QUESTION # 38
On the latest Health Check report from your Cloud TEST environment utilizing a MongoDB add-on, you note the following findings: Category: User Experience, Description: # of slow query rules, Risk: High Category: User Experience, Description: # of slow write to data store nodes, Risk: High Which three things might you do to address this, without consulting the business?

- A. Use smaller CDTs or limit the fields selected in a!queryEntity().
- B. Optimize the database execution. Replace the view with a materialized view.
- C. Reduce the size and complexity of the inputs. If you are passing in a list, consider whether the data model can be redesigned to pass single values instead.
- D. Optimize the database execution using standard database performance troubleshooting methods and tools (such as query

execution plans).
- E. Reduce the batch size for database queues to 10.

Answer: A,C,D

Explanation:
Comprehensive and Detailed In-Depth Explanation:The Health Check report indicates high-risk issues with slow query rules and slow writes to data store nodes in a MongoDB-integrated Appian Cloud TEST environment. As a Lead Developer, you can address these performance bottlenecks without business consultation by focusing on technical optimizations within Appian and MongoDB. The goal is to improve user experience by reducing query and write latency.
* Option B (Optimize the database execution using standard database performance troubleshooting methods and tools (such as query execution plans)):This is a critical step. Slow queries and writes suggest inefficient database operations. Using MongoDB's explain() or equivalent tools to analyze execution plans can identify missing indices, suboptimal queries, or full collection scans. Appian's Performance Tuning Guide recommends optimizing database interactions by adding indices on frequently queried fields or rewriting queries (e.g., using projections to limit returned data). This directly addresses both slow queries and writes without business input.
* Option C (Reduce the size and complexity of the inputs. If you are passing in a list, consider whether the data model can be redesigned to pass single values instead):Large or complex inputs (e.
g., large arrays in a!queryEntity() or write operations) can overwhelm MongoDB, especially in Appian' s data store integration. Redesigning the data model to handle single values or smaller batches reduces processing overhead. Appian's Best Practices for Data Store Design suggest normalizing data or breaking down lists into manageable units, which can mitigate slow writes and improve query performance without requiring business approval.
* Option E (Use smaller CDTs or limit the fields selected in a!queryEntity()):Appian Custom Data Types (CDTs) and a!queryEntity() calls that return excessive fields can increase data transfer and processing time, contributing to slow queries. Limiting fields to only those needed (e.g., using fetchTotalCount selectively) or using smaller CDTs reduces the load on MongoDB and Appian's engine. This optimization is a technical adjustment within the developer's control, aligning with Appian' s Query Optimization Guidelines.
* Option A (Reduce the batch size for database queues to 10):While adjusting batch sizes can help with write performance, reducing it to 10 without analysis might not address the root cause and could slow down legitimate operations. This requires testing and potentially business input on acceptable performance trade-offs, making it less immediate.
* Option D (Optimize the database execution. Replace the view with a materialized view):
Materialized views are not natively supported in MongoDB (unlike relational databases like PostgreSQL), and Appian's MongoDB add-on relies on collection-based storage. Implementing this would require significant redesign or custom aggregation pipelines, which may exceed the scope of a unilateral technical fix and could impact business logic.
These three actions (B, C, E) leverage Appian and MongoDB optimization techniques, addressing both query and write performance without altering business requirements or processes.
References:Appian Documentation - Performance Tuning Guide, Appian MongoDB Add-on Best Practices, Appian Lead Developer Training - Query and Write Optimization.
The three things that might help to address the findings of the Health Check report are:
* B. Optimize the database execution using standard database performance troubleshooting methods and tools (such as query execution plans). This can help to identify and eliminate any bottlenecks or inefficiencies in the database queries that are causing slow query rules or slow write to data store nodes.
* C. Reduce the size and complexity of the inputs. If you are passing in a list, consider whether the data model can be redesigned to pass single values instead. This can help to reduce the amount of data that needs to be transferred or processed by the database, which can improve the performance and speed of the queries or writes.
* E. Use smaller CDTs or limit the fields selected in a!queryEntity(). This can help to reduce the amount of data that is returned by the queries, which can improve the performance and speed of the rules that use them.
The other options are incorrect for the following reasons:
* A. Reduce the batch size for database queues to 10. This might not help to address the findings, as reducing the batch size could increase the number of transactions and overhead for the database, which could worsen the performance and speed of the queries or writes.
* D. Optimize the database execution. Replace the new with a materialized view. This might not help to address the findings, as replacing a view with a materialized view could increase the storage space and maintenance cost for the database, which could affect the performance and speed of the queries or writes. Verified References: Appian Documentation, section "Performance Tuning".
Below are the corrected and formatted questions based on your input, including the analysis of the provided image. The answers are 100% verified per official Appian Lead Developer documentation and best practices as of March 01, 2025, with comprehensive explanations and references provided.

NEW QUESTION # 39

You need to export data using an out-of-the-box Appian smart service. Which two formats are available (or data generation?

- A. Excel
- B. XML
- C. JSDN
- D. CSV

Answer: A,D

Explanation:
The two formats that are available for data generation using an out-of-the-box Appian smart service are:
* A. CSV. This is a comma-separated values format that can be used to export data in a tabular form, such as records, reports, or grids. CSV files can be easily opened and manipulated by spreadsheet applications such as Excel or Google Sheets.
* C. Excel. This is a format that can be used to export data in a spreadsheet form, with multiple worksheets, formatting, formulas, charts, and other features. Excel files can be opened by Excel or other compatible applications.
The other options are incorrect for the following reasons:
* B. XML. This is a format that can be used to export data in a hierarchical form, using tags and attributes to define the structure and content of the data. XML files can be opened by text editors or XML parsers, but they are not supported by the out-of-the-box Appian smart service for data generation.
* D. JSON. This is a format that can be used to export data in a structured form, using objects and arrays to represent the data. JSON files can be opened by text editors or JSON parsers, but they are not supported by the out-of-the-box Appian smart service for data generation. Verified References: Appian Documentation, section "Write to Data Store Entity" and "Write to Multiple Data Store Entities".

NEW QUESTION # 40
Your client's customer management application is finally released to Production. After a few weeks of small enhancements and patches, the client is ready to build their next application. The new application will leverage customer information from the first application to allow the client to launch targeted campaigns for select customers in order to increase sales. As part of the first application, your team had built a section to display key customer information such as their name, address, phone number, how long they have been a customer, etc. A similar section will be needed on the campaign record you are building. One of your developers shows you the new object they are working on for the new application and asks you to review it as they are running into a few issues. What feedback should you give?

- A. Create a duplicate version of that section designed for the campaign record.
- B. Point the developer to the relevant areas in the documentation or Appian Community where they can find more information on the issues they are running into.
- C. Provide guidance to the developer on how to address the issues so that they can proceed with their work.
- D. Ask the developer to convert the original customer section into a shared object so it can be used by the new application.

Answer: D

Explanation:
Comprehensive and Detailed In-Depth Explanation:
The scenario involves reusing a customer information section from an existing application in a new application for campaign management, with the developer encountering issues. Appian's best practices emphasize reusability, efficiency, and maintainability, especially when leveraging existing components across applications.
Option B (Ask the developer to convert the original customer section into a shared object so it can be used by the new application):
This is the recommended approach. Converting the original section into a shared object (e.g., a reusable interface component) allows it to be accessed across applications without duplication. Appian's Design Guide highlights the use of shared components to promote consistency, reduce redundancy, and simplify maintenance. Since the new application requires similar customer data (name, address, etc.), reusing the existing section-after ensuring it is modular and adaptable-addresses the developer's issues while aligning with the client's goal of leveraging prior work. The developer can then adjust the shared object (e.g., via parameters) to fit the campaign context, resolving their issues collaboratively.
Option A (Provide guidance to the developer on how to address the issues so that they can proceed with their work):
While providing guidance is valuable, it doesn't address the root opportunity to reuse existing code. This option focuses on fixing the new object in isolation, potentially leading to duplicated effort if the original section could be reused instead.
Option C (Point the developer to the relevant areas in the documentation or Appian Community where they can find more information on the issues they are running into):
This is a passive approach and delays resolution. As a Lead Developer, offering direct support or a strategic solution (like reusing components) is more effective than redirecting the developer to external resources without context.

Option D (Create a duplicate version of that section designed for the campaign record):
Duplication violates Appian's principle of DRY (Don't Repeat Yourself) and increases maintenance overhead. Any future updates to customer data display logic would need to be applied to multiple objects, risking inconsistencies.
Given the need to leverage existing customer information and the developer's issues, converting the section to a shared object is the most efficient and scalable solution.

NEW QUESTION # 41
You are developing a case management application to manage support cases for a large set of sites. One of the tabs in this application s site Is a record grid of cases, along with Information about the site corresponding to that case. Users must be able to filter cases by priority level and status.
You decide to create a view as the source of your entity-backed record, which joins the separate case/site tables (as depicted in the following Image).
Which three column should be indexed?

- A. modified_date
- B. name
- C. case_id
- D. site_id
- E. status
- F. priority

Answer: D,E,F

Explanation:
Indexing columns can improve the performance of queries that use those columns in filters, joins, or order by clauses. In this case, the columns that should be indexed are site_id, status, and priority, because they are used for filtering or joining the tables. Site_id is used to join the case and site tables, so indexing it will speed up the join operation. Status and priority are used to filter the cases by the user's input, so indexing them will reduce the number of rows that need to be scanned. Name, modified_date, and case_id do not need to be indexed, because they are not used for filtering or joining. Name and modified_date are only used for displaying information in the record grid, and case_id is only used as a unique identifier for each record.
Verified References: Appian Records Tutorial, Appian Best Practices
As an Appian Lead Developer, optimizing a database view for an entity-backed record grid requires indexing columns frequently used in queries, particularly for filtering and joining. The scenario involves a record grid displaying cases with site information, filtered by "priority level" and "status," and joined via the site_id foreign key. The image shows two tables (site and case) with a relationship via site_id. Let's evaluate each column based on Appian's performance best practices and query patterns:
* A. site_id:This is a primary key in the site table and a foreign key in the case table, used for joining the tables in the view. Indexing site_id in the case table (and ensuring it's indexed in site as a PK) optimizes JOIN operations, reducing query execution time for the record grid. Appian's documentation recommends indexing foreign keys in large datasets to improve query performance, especially for entity-backed records. This is critical for the join and must be included.
* B. status:Users filter cases by "status" (a varchar column in the case table). Indexing status speeds up filtering queries (e.g., WHERE status = 'Open') in the record grid, particularly with large datasets.
Appian emphasizes indexing columns used in WHERE clauses or filters to enhance performance, making this a key column for optimization. Since status is a common filter, it's essential.
* C. name:This is a varchar column in the site table, likely used for display (e.g., site name in the grid).
However, the scenario doesn't mention filtering or sorting by name, and it's not part of the join or required filters. Indexing name could improve searches if used, but it's not a priority given the focus on priority and status filters. Appian advises indexing only frequently queried or filtered columns to avoid unnecessary overhead, so this isn't necessary here.
* D. modified_date:This is a date column in the case table, tracking when cases were last updated. While useful for sorting or historical queries, the scenario doesn't specify filtering or sorting by modified_date in the record grid. Indexing it could help if used, but it's not critical for the current requirements.
Appian's performance guidelines prioritize indexing columns in active filters, making this lower priority than site_id, status, and priority.
* E. priority:Users filter cases by "priority level" (a varchar column in the case table). Indexing priority optimizes filtering queries (e.g., WHERE priority = 'High') in the record grid, similar to status. Appian' s documentation highlights indexing columns used in WHERE clauses for entity-backed records, especially with large datasets. Since priority is a specified filter, it's essential to include.
* F. case_id:This is the primary key in the case table, already indexed by default (as PKs are automatically indexed in most databases). Indexing it again is redundant and unnecessary, as Appian's Data Store configuration relies on PKs for unique identification but doesn't require additional indexing for performance in this context. The focus is on join and filter columns, not the PK itself.
Conclusion: The three columns to index are A (site_id), B (status), and E (priority). These optimize the JOIN (site_id) and filter

performance (status, priority) for the record grid, aligning with Appian's recommendations for entity-backed records and large datasets. Indexing these columns ensures efficient querying for user filters, critical for the application's performance.
References:
* Appian Documentation: "Performance Best Practices for Data Stores" (Indexing Strategies).
* Appian Lead Developer Certification: Data Management Module (Optimizing Entity-Backed Records).
* Appian Best Practices: "Working with Large Data Volumes" (Indexing for Query Performance).

NEW QUESTION # 42
......

Have you signed up for Appian ACD301 Exam? Will masses of reviewing materials and questions give you a headache? Actualtests4sure can help you to solve this problem. It is absolutely trustworthy website. Only if you choose to use exam dumps Actualtests4sure provides, you can absolutely pass your exam successfully. You spend lots of time on these reviewing materials you don't know whether it is useful to you, rather than experiencing the service Actualtests4sure provides for you. So, hurry to take action.

Dumps ACD301 Discount: https://www.actualtests4sure.com/ACD301-test-questions.html

Firstly you could know the price and the version of our ACD301 study question, the quantity of the questions and the answers, Appian ACD301 Sample Exam So IT industry has caused much attention and plays an important role in the current society, While, how to get the latest and valid ACD301 study material for training, With constant practice, users will find that feedback reports are getting better, because users spend enough time on our ACD301 test prep.

To get to launch, you need to decide who is doing Dumps ACD301 Discount what and what process they'll follow to create, approve, and publish the content, In fact,if you have ever had to retake a failed exam then ACD301 you probably noticed that only a few of the questions were repeats from your first attempt.

Free PDF Appian - Newest ACD301 - Appian Lead Developer Sample Exam

Firstly you could know the price and the version of our ACD301 study question, the quantity of the questions and the answers, So IT industry has caused much attention and plays an important role in the current society.

While, how to get the latest and valid ACD301 study material for training, With constant practice, users will find that feedback reports are getting better, because users spend enough time on our ACD301 test prep.

At the same time, if you have any questions during the trial Dumps ACD301 Discount period, you can feel free to communicate with our staff, and we will do our best to solve all the problems for you.

- ACD301 Study Guide Pdf ACD301 Certificate Exam ACD301 Reliable Exam Pattern Download " ACD301 " for free by simply searching on ☀ www.easy4engine.com ☀ ACD301 Study Guide Pdf
- Specifications of Appian ACD301 Practice Exam Software Open ▷ www.pdfvce.com ◁ and search for 【 ACD301 】 to download exam materials for free ACD301 Reliable Exam Pattern
- Pass Guaranteed 2026 Appian Trustable ACD301 Sample Exam 《 www.vceengine.com 》 is best website to obtain ➥ ACD301 for free download ACD301 Test Simulator Fee
- Reliable ACD301 Exam Labs Reliable ACD301 Test Objectives Best ACD301 Study Material Open website " www.pdfvce.com " and search for ACD301 for free download ACD301 Test Simulator Fee
- 2026 Newest ACD301 – 100% Free Sample Exam | Dumps Appian Lead Developer Discount Search for ➤ ACD301 on ⇒ www.examdiscuss.com ⇐ immediately to obtain a free download ACD301 Certificate Exam
- New ACD301 Sample Exam | High-quality Dumps ACD301 Discount: Appian Lead Developer Download [ACD301] for free by simply entering 【 www.pdfvce.com 】 website ACD301 Exam Learning
- Valid ACD301 Test Answers ACD301 Reliable Exam Test ACD301 Real Exam Questions Search for ▷ ACD301 ◁ and obtain a free download on ➽ www.prepawayexam.com ACD301 Certificate Exam
- ACD301 Exam Learning Reliable ACD301 Test Objectives Reliable ACD301 Test Objectives Copy URL ➤ www.pdfvce.com open and search for 「 ACD301 」 to download for free ➡Reliable ACD301 Test Objectives
- Excellent ACD301 Sample Exam - Leading Offer in Qualification Exams - Fast Download ACD301: Appian Lead Developer ♥ Search on [www.prepawaypdf.com] for ➡ ACD301 to obtain exam materials for free download ACD301 Study Guide Pdf
- High-quality ACD301 Sample Exam Offer You The Best Dumps Discount | Appian Lead Developer " www.pdfvce.com " is best website to obtain ☀ ACD301 ☀ for free download Reliable ACD301 Study Notes
- 2026 Newest ACD301 – 100% Free Sample Exam | Dumps Appian Lead Developer Discount Simply search for ▷

ACD301 ◁ for free download on { www.exam4labs.com } Reliable ACD301 Study Notes

- www.zazzle.com, www.fanart-central.net, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes

DOWNLOAD the newest Actualtests4sure ACD301 PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=1xGVPHEp7yRQBROTOUJ0JnTfgE9x192Rg