ACD300$^{\text{Q\&As}}$

Appian Certified Lead Developer

Pass Appian ACD300 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.geekcert.com/acd300.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Appian Official Exam Center

QUESTION 1

You are required to configure a connection so that Jira can inform Appian when specific tickets change (using webhook).

Which three required steps will allow you to connect both systems?

A. Create a Web API object and set up the correct security.

B. Configure the connection In Jira specifying the URE and credentials

C. Create a new API Key and associate a service account

D. Give the service account system administrator privileges

E. Create an integration object from Applan to Jira to periodically check the ticket status

Correct Answer: ABC

The three required steps that will allow you to connect both systems are:

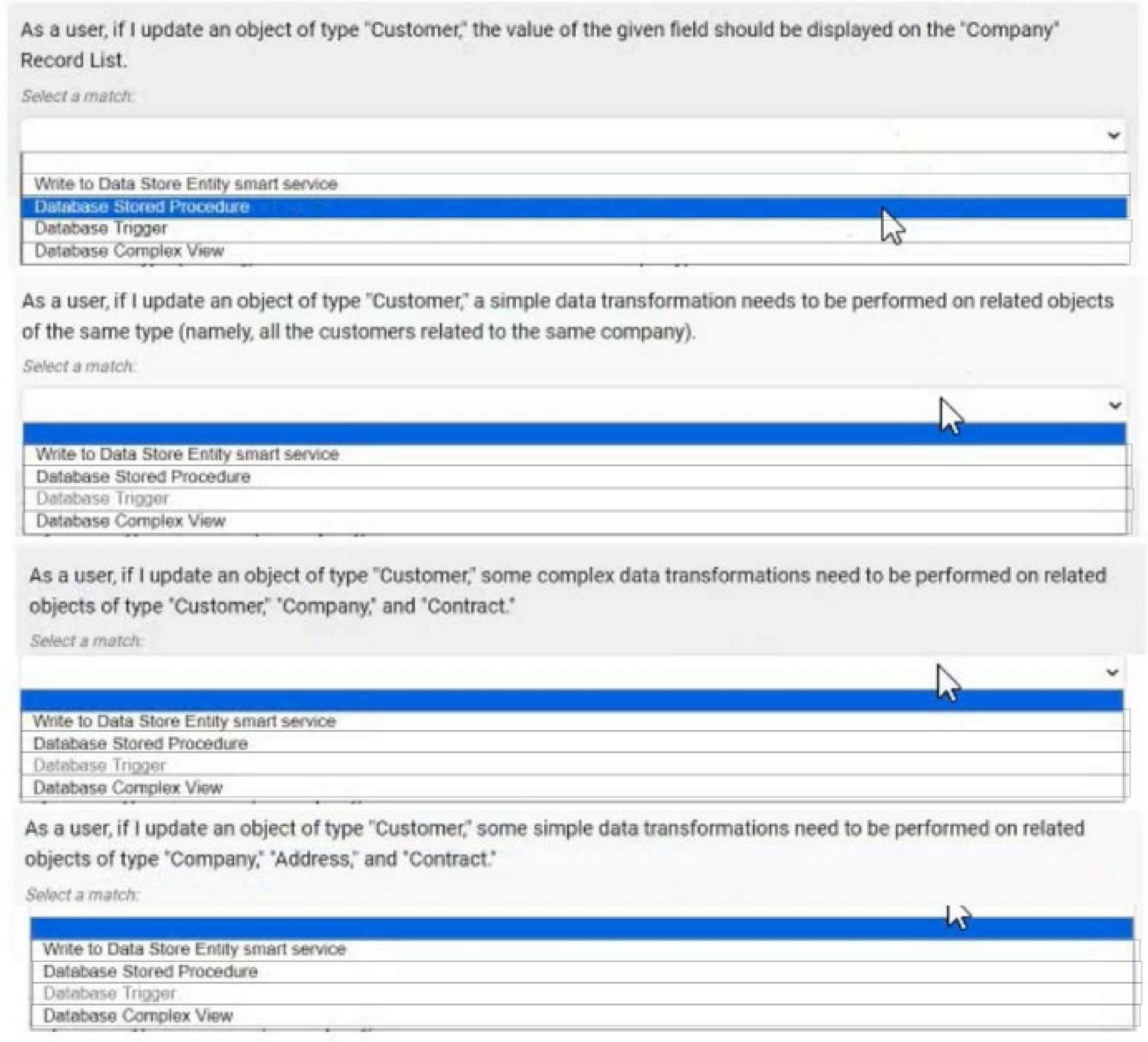
A. Create a Web API object and set up the correct security. This will allow you to define an endpoint in Appian that can receive requests from Jira via webhook. You will also need to configure the security settings for the Web API object, such as authentication method, allowed origins, and access control. 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 References: Appian Documentation, section "Web API" and "API Keys".

QUESTION 2

HOTSPOT For each scenario outlined, match the best tool to use to meet expectations. Each tool will be used once Note: To change your responses, you may deselected your response by clicking the blank space at the top of the selection list.

Hot Area:



As a user, if I update an object of type "Customer," the value of the given field should be displayed on the "Company" Record List.

Select a match:

- Write to Data Store Entity smart service
- Database Stored Procedure
- Database Trigger
- Database Complex View

As a user, if I update an object of type "Customer," a simple data transformation needs to be performed on related objects of the same type (namely, all the customers related to the same company).

Select a match:

- Write to Data Store Entity smart service
- Database Stored Procedure
- Database Trigger
- Database Complex View

As a user, if I update an object of type "Customer," some complex data transformations need to be performed on related objects of type "Customer," "Company," and "Contract."

Select a match:

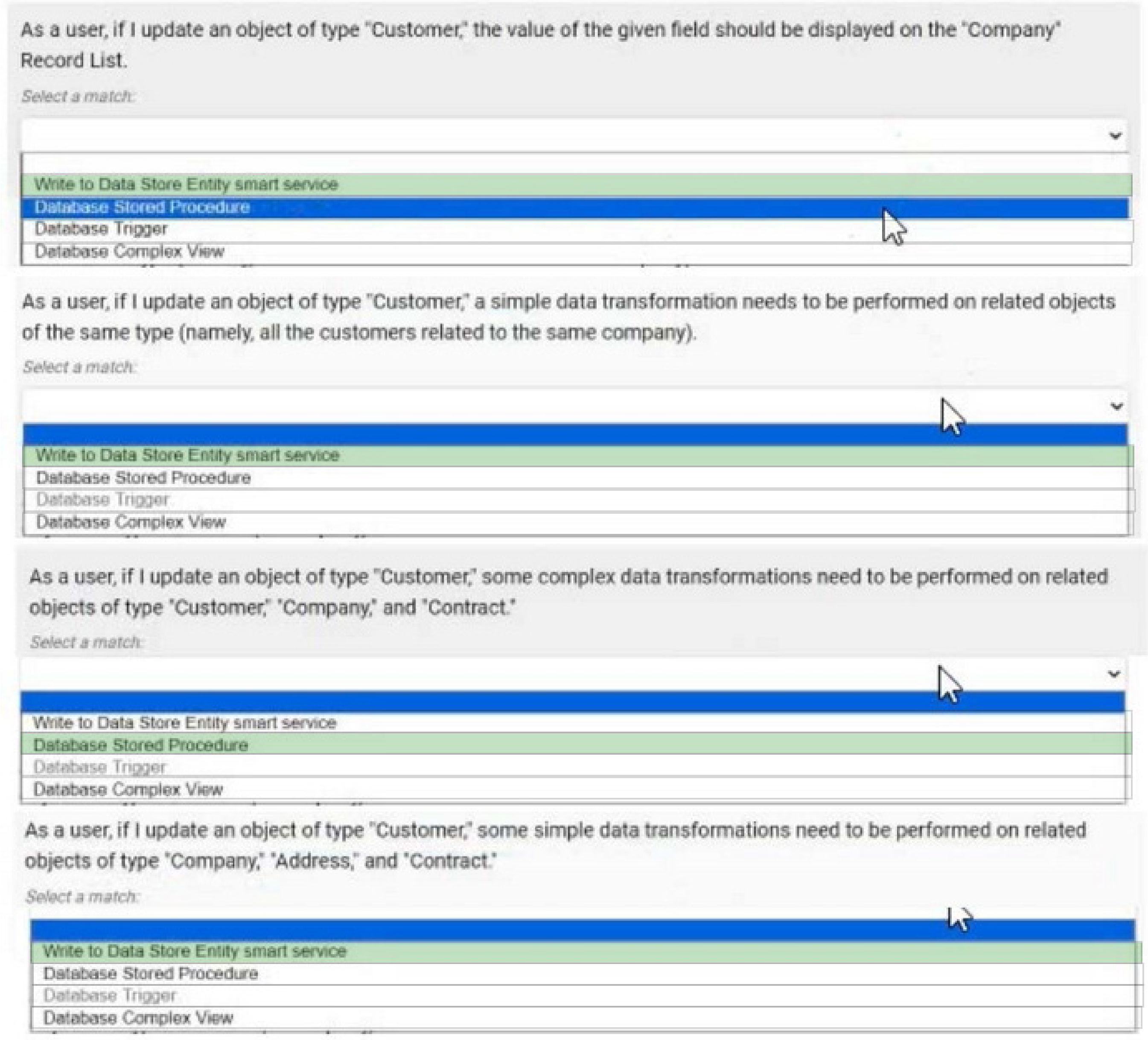
- Write to Data Store Entity smart service
- Database Stored Procedure
- Database Trigger
- Database Complex View

As a user, if I update an object of type "Customer," some simple data transformations need to be performed on related objects of type "Company," "Address," and "Contract."

Select a match:

- Write to Data Store Entity smart service
- Database Stored Procedure
- Database Trigger
- Database Complex View

Correct Answer:



As a user, if I update an object of type "Customer," the value of the given field should be displayed on the "Company" Record List.

Select a match:

Write to Data Store Entity smart service
Database Stored Procedure
Database Trigger
Database Complex View

As a user, if I update an object of type "Customer," a simple data transformation needs to be performed on related objects of the same type (namely, all the customers related to the same company).

Select a match:

Write to Data Store Entity smart service
Database Stored Procedure
Database Trigger
Database Complex View

As a user, if I update an object of type "Customer," some complex data transformations need to be performed on related objects of type "Customer," "Company," and "Contract."

Select a match:

Write to Data Store Entity smart service
Database Stored Procedure
Database Trigger
Database Complex View

As a user, if I update an object of type "Customer," some simple data transformations need to be performed on related objects of type "Company," "Address," and "Contract."

Select a match:

Write to Data Store Entity smart service
Database Stored Procedure
Database Trigger
Database Complex View

QUESTION 3

You add an index on the searched field of a MySQL table with many rows (>100k).

The field would benefit greatly from the Index in which three scenarios?

A. The field contains a textual shot Business code.

B. The field contains long unstructured text such as a hash

C. The field contains many datetimes, covering a large range

D. The Add contains Dig integers, above and below 0.

E. The field contains a structured JSON.

Correct Answer: ACD

The field would benefit greatly from the index in the following scenarios:

A. The field contains a textual short Business code. This is a scenario where an index can improve the performance of queries that search for exact matches or ranges of values in the field. A textual short Business code is likely to have high

cardinality, meaning that it has many distinct values and low duplication. This makes the index more selective and efficient, as it can quickly narrow down the results based on the search criteria.

C. The field contains many datetimes, covering a large range. This is a scenario where an index can improve the performance of queries that search for exact matches or ranges of values in the field. A datetime field is likely to have high

cardinality, meaning that it has many distinct values and low duplication. This makes the index more selective and efficient, as it can quickly narrow down the results based on the search criteria.

D. The field contains big integers, above and below 0. This is a scenario where an index can improve the performance of queries that search for exact matches or ranges of values in the field. A big integer field is likely to have high cardinality,

meaning that it has many distinct values and low duplication. This makes the index more selective and efficient, as it can quickly narrow down the results based on the search criteria.

The other options are incorrect for the following reasons:

B. The field contains long unstructured text such as a hash. This is a scenario where an index might not improve the performance of queries that search for exact matches or ranges of values in the field. A long unstructured text field is likely to

have low cardinality, meaning that it has few distinct values and high duplication. This makes the index less selective and efficient, as it cannot quickly narrow down the results based on the search criteria. Moreover, indexing a long

unstructured text field could increase thestorage space and maintenance cost for the database, which could affect the overall performance.

E. The field contains a structured JSON. This is a scenario where an index might not improve the performance of queries that search for exact matches or ranges of values in the field. A structured JSON field is not a native data type in

MySQL, and it requires special functions or operators to access or manipulate its elements. Indexing a structured JSON field could increase the complexity and overhead for the database, which could affect the overall performance. Verified

References: Appian Documentation, section "Query Optimization".

QUESTION 4

What are two advantages of having High Availability (HA) for Applan Cloud applications?

A. An Applan Cloud HA instance is composed of multiple active nodes running in different availability zones in differentregions.

B. Data andtransactions are continuouslyreplicated across the active nodes to achieve redundancy and avoid single points offailure.

C. A typical Appian Cloud HA instance is composed of two active nodes.

D. In the event of a system failure, your Appian instance will fie restored and available to your users in less than 15 minutes.having lost no more than the last 1minute worth of data.

Correct Answer: BD

The two advantages of having High Availability (HA) for Appian Cloud applications are:

B. Data and transactions are continuously replicated across the active nodes to achieve redundancy and avoid single points of failure. This is an advantage of having HA, as it ensures that there is always a backup copy of data and transactions in case one of the nodes fails or becomes unavailable. This also improves data integrity and consistency across the nodes, as any changes made to one node are automatically propagated to the other node. D. In the event of a system failure, your Appian instance will be restored and available to your users in less than 15 minutes, having lost no more than the last 1 minute worth of data. This is an advantage of having HA, as it guarantees a high level of service availability and reliability for your Appian instance. If one of the nodes fails or becomes unavailable, the other node will take over and continue to serve requests without any noticeable downtime or data loss for your users. The other options are incorrect for the following reasons:

A. An Appian Cloud HA instance is composed of multiple active nodes running in different availability zones in different regions. This is not an advantage of having HA, but rather a description of how HA works in Appian Cloud. An Appian Cloud HA instanceconsists of two active nodes running in different availability zones within the same region, not different regions.

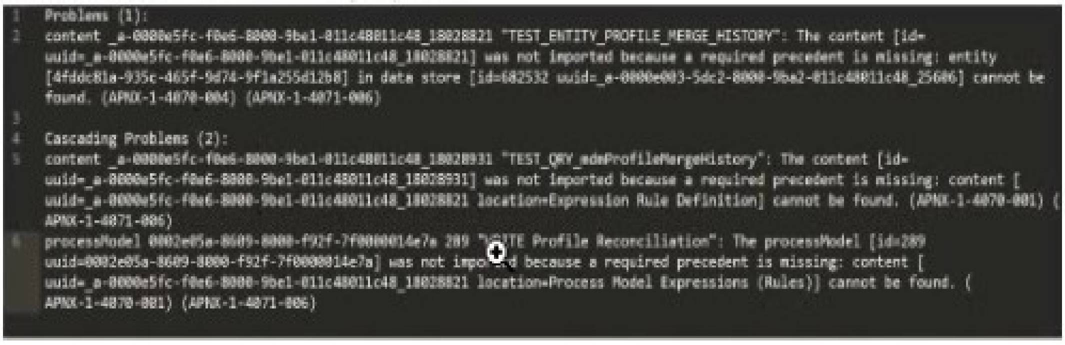
C. A typical Appian Cloud HA instance is composed of two active nodes. This is not an advantage of having HA, but rather a description of how HA works in Appian Cloud. A typical Appian Cloud HA instance consists of two active nodes running in different availability zones within the same region, but this does not necessarily provide any benefit over having one active node. Verified References: Appian Documentation, section "High Availability".

QUESTION 5

You are taking your package from the source environment and importing it into the target environment.

Review the errors encountered during inspection:

Whatis the first action you should take to Investigate the issue?



```
Problems (1):
content _a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028821 "TEST_ENTITY_PROFILE_MERGE_HISTORY": The content [id=
uuid=_a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028821] was not imported because a required precedent is missing: entity
[4fddc81a-935c-465f-9d74-9f1a255d12b8] in data store [id=682532 uuid=_a-0000e003-5dc2-8000-9ba2-011c48011c48_25606] cannot be
found. (APNX-1-4070-004) (APNX-1-4071-006)

Cascading Problems (2):
content _a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028931 "TEST_QRY_mdmProfileMergeHistory": The content [id=
uuid=_a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028931] was not imported because a required precedent is missing: content [
uuid=_a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028821 location=Expression Rule Definition] cannot be found. (APNX-1-4070-001) (
APNX-1-4071-006)
processModel 0002e05a-8609-8000-f92f-7f0000014e7a 289 "[illegible]TE Profile Reconciliation": The processModel [id=289
uuid=0002e05a-8609-8000-f92f-7f0000014e7a] was not imported because a required precedent is missing: content [
uuid=_a-0000e5fc-f0e6-8000-9be1-011c48011c48_18028821 location=Process Model Expressions (Rules)] cannot be found. (
APNX-1-4070-001) (APNX-1-4071-006)
```

A. Check whether the object(UUIDending in 18028821) is included in this package

B. Check whether the object(UUD ending in 7t00000i4e7a)is included in this package

C. Check whether the object (UUID ending in 25606) is included in this package

D. Check whether the object (UUID ending in 18028931) is included in this package

Correct Answer: B

The error message indicates that the object with UUID ending in 18028821 has a dependency on another object with UUID ending in 7t00000i4e7a, which is missing from the target environment. Therefore, the first action to investigate the issue is to check whether the object with UUID ending in 7t00000i4e7a is included in this package or not. If not, then it should be added to the package or imported separately before importing the current package. Verified References: Appian Certified Lead Developer study guide, page 17, section "Importing and Exporting Applications".