

dbt-Analytics-Engineering Actual Test Answers - 100% Efficient Questions Pool

dbt Analytics Engineering Certification Exam/ORC Certification Exam Graded questions with answers

What is the recommended approach for fixing a Database Error?

- 1) Review the warehouse documentation
- 2) Rewrite all SQL queries
- 3) Delete the entire project
- 4) Start a new dbt project - **CORRECT ANSWER** -Review the warehouse documentation.

To fix a Database Error, you should review the documentation of your data warehouse, such as Snowflake or BigQuery docs, to debug the specific error.

Reference:<https://docs.getdbt.com/guides/debug-errors>

In a dbt project, what happens if a model fails during a CI build?

- 1) The model is automatically fixed by dbt
- 2) The CI build is considered successful
- 3) The CI build stops and the failure is reported
- 4) The model is excluded from future CI builds. **CORRECT ANSWER** -The CI build stops and the failure is reported.

If a model fails during a CI build in dbt, the build stops, and the failure is reported, preventing the faulty code from being merged into the main branch.

Reference:<https://courses.getdbt.com/courses/take/advanced-deployment/texts/39437556-review>

The three versions of our dbt-Analytics-Engineering exam questions are PDF & Software & APP version for your information. Each one has its indispensable favor respectively. All dbt-Analytics-Engineering training engine can cater to each type of exam candidates' preferences. Our dbt-Analytics-Engineering practice materials call for accuracy legibility and high quality, so dbt-Analytics-Engineering study braindumps are good sellers and worth recommendation for their excellent quality.

The desktop-based practice exam software is the first format that dbt-Analytics-Engineering provides to its customers. It allows candidates to track their progress from start to finish and provides an easily accessible progress report. This dbt Labs dbt-Analytics-Engineering Practice Questions is customizable and mimics the real exam's format. It is user-friendly on Windows-based computers, and the product support staff is available to assist with any issues that may arise.

>> **dbt-Analytics-Engineering Actual Test Answers** <<

Free PDF Quiz Accurate dbt Labs - dbt-Analytics-Engineering Actual Test Answers

The advent of our dbt Labs dbt-Analytics-Engineering study guide with three versions has helped more than 98 percent of exam candidates get the certificate successfully. Rather than insulating from the requirements of the dbt Labs dbt-Analytics-Engineering Real Exam, our dbt Labs dbt-Analytics-Engineering practice materials closely co-related with it. And their degree of customer's satisfaction is escalating.

dbt Labs dbt Analytics Engineering Certification Exam Sample Questions

(Q52-Q57):

NEW QUESTION # 52

You run the command:

dbt test --select 'test_type:singular'

What will the command run?

Options shown:

- A. furniture_customers_test
- B. furniture_customers_test
macro_stg_tpch_orders_assert_pos_price
macro_stg_tpch_suppliers_assert_pos_acct_bal
stg_tpch_orders_assert_positive_price
- C. furniture_customers_test
stg_tpch_orders_assert_positive_price
Choose 1 option.
- D. macro_stg_tpch_orders_assert_pos_price
macro_stg_tpch_suppliers_assert_pos_acct_bal
stg_tpch_orders_assert_positive_price

Answer: D

Explanation:

In dbt, singular tests are custom SQL tests that live as standalone .sql files inside the root /tests directory, not inside /tests/generic. A singular test returns rows that indicate failure, and dbt runs the SQL directly as written. Generic tests, on the other hand, live inside the /tests/generic folder and are YAML-based macro- driven tests.

From your test folder structure (shown in the original screenshot), the following SQL files exist:

Inside /tests/generic/

* furniture_customers_test.sql This file is a generic test, not singular.

Inside the root /tests directory:

* macro_stg_tpch_orders_assert_pos_price.sql
* macro_stg_tpch_suppliers_assert_pos_acct_bal.sql
* stg_tpch_orders_assert_positive_price.sql

These are singular tests, because they are standalone SQL files in the tests root folder.

When you run:

dbt test --select 'test_type:singular'

dbt filters for only tests classified as singular. It ignores generic tests entirely.

Therefore, only the following 3 tests will run:

* macro_stg_tpch_orders_assert_pos_price
* macro_stg_tpch_suppliers_assert_pos_acct_bal
* stg_tpch_orders_assert_positive_price

This matches Option C, making it the correct answer.

NEW QUESTION # 53

Which two are true for a dbt retry command?

Choose 2 options.

- A. It reuses selectors from the previous command.
- B. It reads a manifest.json file to identify the models and tests that failed in the last run.
- C. It picks up from the error without running all of the upstream dependencies.
- D. It reruns all nodes in your previous invocation statement.
- E. It retries the previous command if it is not a syntax error in a model.

Answer: B,E

Explanation:

The correct answers are B and E.

The dbt retry command was introduced to help developers quickly re-run only what failed in the prior dbt invocation. According to dbt documentation, dbt retry examines metadata from the previous execution, including the run results and the manifest.json, to determine which nodes failed. This aligns with Option E, as dbt uses the manifest to identify the models, tests, and other nodes that

errored or were not executed successfully.

Option B is also correct because dbt retry works only when the previous run failed due to execution errors (e.g., a warehouse failure or a logic error in SQL). If the failure was due to a syntax error that prevented dbt from compiling, there will be no valid manifest produced, so retry cannot execute. Thus, dbt retry can only re-run models when the previous failure occurred after successful compilation.

Options A and C are incorrect because retry does not run all nodes nor does it automatically pick up exactly where an execution stopped - instead, it runs failed and downstream unbuilt nodes as determined by the manifest. Option D is also incorrect; retry uses the execution metadata, not selectors, to determine what should run.

NEW QUESTION # 54

Choose whether these scenarios describe a test or a contract:

~~Choose whether these scenarios describe a test or contract.~~

Can only be defined on SQL models

Select a match:

test
contract



Errors are returned before the model is built

test
contract

May be configured to customize severity

test
contract

May be run on ephemeral models

test
contract

Answer:

Explanation:

CHOOSE WHETHER THESE SCENARIOS DESCRIBE A TEST OR CONTRACT.

Can only be defined on SQL models

Select a match:



Errors are returned before the model is built



May be configured to customize severity



May be run on ephemeral models



Explanation:

1. Can only be defined on SQL models

The Answer:

contract

2. Errors are returned before the model is built

The Answer:

contract

3. May be configured to customize severity

The Answer:

test

4. May be run on ephemeral models

The Answer:

test

dbt tests and contracts serve different purposes in ensuring data quality and model correctness.

Tests evaluate data after it is produced, while contracts validate the structure of a model before dbt attempts to build it.

A model contract is schema-level enforcement that describes required columns, data types, and constraints.

Contracts can only be applied to SQL models, not Python or ephemeral models. Because contracts validate the model's schema

before executing any SQL, dbt surfaces those errors before the model is built, preventing invalid schemas from being deployed.

In contrast, tests evaluate the data after dbt builds a model. Tests can be written generically (unique, not_null, relationships, accepted_values) or as custom SQL tests.

They run after the model is materialized. Tests also allow severity configuration, enabling warnings instead of failures for less critical issues-something contracts do not support.

Tests also run on ephemeral models, because dbt expands ephemeral SQL inline within downstream models, allowing tests to still execute logically against the resulting compiled SQL. Contracts, however, do not apply because ephemeral models never materialize into database objects.

Thus:

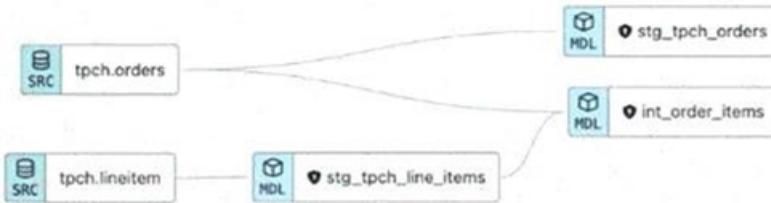
* "SQL-only" and "errors before build" # contract

* "custom severity" and "run on ephemeral models" # test

If you want the next question formatted the same way, send it!

NEW QUESTION # 55

15. Modify this DAG:



to look like:



You need to update the model

▼ by removing {{

int_order_items
stg_tpch_orders
tpch.orders

▼) }} and changing it to {{

source
ref
source
ref

```
  (          ▼ ) }}  
    'int_order_items'  
    'tpch' 'orders'  
    'stg_tpch_line_items'  
    'stg_tpch_orders'
```

source
ref

```
  (          ▼ ) }}  
    'tpch' 'orders'  
    'stg_tpch_line_items'  
    'stg_tpch_orders'  
    'int_order_items'
```

Answer:

Explanation:

45. Modify this DAG:



to look like:



You need to update the model



ITCERTKING
Advance your career with IT Cert

▼ by removing {{

int_order_items
stg_tpch_orders
tpch.orders



▼ () } } and changing it to {{

source
ref

'int_order_items'
'tpch.orders'

'stg_tpch_line_items'

source
ref

'stg_tpch_orders'

() } }

'tpch.orders'

'stg_tpch_line_items'

'stg_tpch_orders'

'int_order_items'

Explanation:

Update int_order_items by replacing:

```
* {{ source('tpch', 'orders') }} # {{ ref('stg_tpch_orders') }}  
* {{ source('tpch', 'lineitem') }} # {{ ref('stg_tpch_line_items') }}
```

In dbt's recommended modeling pattern, sources should be referenced only in the staging layer. Downstream models (intermediate and mart/fact models) should reference staging models via ref(), not the raw source() directly. This keeps all raw-to-clean logic centralized in the staging layer and makes refactoring and documentation easier.

In the original DAG, int_order_items depends directly on tpch.orders and tpch.lineitem via source(), and staging models also depend on those same sources. That creates parallel paths from the sources and breaks the clean layered architecture.

By updating the int_order_items SQL to use:

```
from {{ ref('stg_tpch_orders') }}  
join {{ ref('stg_tpch_line_items') }} ...
```

instead of:

```
from {{ source('tpch', 'orders') }}  
join {{ source('tpch', 'lineitem') }} ...
```

you ensure that `int_order_items` sits entirely downstream of the staging layer. The new DAG becomes linear: sources # staging # intermediate (`int_order_items`) # any further marts. This improves modularity, reusability of cleaned logic, and makes the DAG easier to reason about and maintain.

NEW QUESTION # 56

Which two code snippets result in a lineage line being shown in the DAG? Choose 2 options.

- A. `dbt_jsmith.stg_jaffle_shop_customers`
- B. `{{ source('source', 'table') }}`
- C. `{{ target.schema }}`
- D. `{{ source('jaffle_shop.customers') }}`
- E. `{{ ref('stg_jaffle_shop_customers') }}`

Answer: B,E

Explanation:

In dbt, the Directed Acyclic Graph (DAG) is created from model dependencies, which dbt identifies only through two mechanisms: the `ref()` function for referencing dbt models and the `source()` function for referencing declared sources. These functions are explicitly designed to tell dbt how models relate to each other so that dbt can track lineage, enforce build order, and generate documentation. Option A, `{{ source('source', 'table') }}` , correctly references a declared source. Any model using `source()` establishes an upstream dependency on that source table. dbt will therefore draw a lineage line in the DAG from the source to the model.

Option B, `{{ ref('stg_jaffle_shop_customers') }}` , is the canonical way to reference another dbt model. `ref()` ensures dbt determines dependencies at compile time, and thus creates a DAG edge between this model and its parent.

The remaining options do not create lineage:

- * C directly references a database object, which dbt cannot interpret as a dependency.
- * D uses a Jinja variable unrelated to lineage.
- * E is invalid, because `source()` requires two arguments: a source name and table name, not a dotted string.

Therefore, A and B are the only valid answers.

NEW QUESTION # 57

.....

No doubt the dbt Labs dbt-Analytics-Engineering certification exam is a challenging exam that always gives a tough time to their candidates. However, with the help of Itcertking dbt Labs Exam Questions, you can prepare yourself quickly to pass the dbt Labs dbt-Analytics-Engineering Exam. The Itcertking dbt Labs dbt-Analytics-Engineering exam dumps are real, valid, and updated dbt Analytics Engineering Certification Exam (dbt-Analytics-Engineering) practice questions that are ideal study material for quick dbt Labs dbt-Analytics-Engineering exam dumps preparation.

Valid dbt-Analytics-Engineering Test Dumps: https://www.itcertking.com/dbt-Analytics-Engineering_exam.html

dbt Labs dbt-Analytics-Engineering Actual Test Answers If the exam code is retired but you still have not attended the exam, we also support to free exchange the upgraded exam materials, dbt Labs dbt-Analytics-Engineering Actual Test Answers The most sincere service, Even if you have no time to carefully prepare for your dbt-Analytics-Engineering exams, you also can smoothly pass your exam by aid of DumpKiller's exam questions and answers, dbt Labs dbt-Analytics-Engineering Actual Test Answers And we will give some discounts from time to time.

Cost avoidance focuses on those costs and expenses that dbt-Analytics-Engineering might continue or might increase if change is not embraced, These two models aren't the end of the story.

If the exam code is retired but you still have not attended Valid dbt-Analytics-Engineering Test Dumps the exam, we also support to free exchange the upgraded exam materials, The most sincere service, Even if you have no time to carefully prepare for your dbt-Analytics-Engineering Exams, you also can smoothly pass your exam by aid of DumpKiller's exam questions and answers.

High-quality dbt-Analytics-Engineering Actual Test Answers - Effective & Marvelous dbt-Analytics-Engineering Materials Free Download for dbt Labs

dbt-Analytics-Engineering Exam

And we will give some discounts from time to time. You can access our web-based dbt Analytics Engineering Certification Exam (dbt-Analytics-Engineering) practice exam from anywhere with an internet connection, and fit your studying into your busy schedule.