



CERTSWARRIOR

# dbt-Labs

## dbt-Analytics-Engineering

### dbt Analytics Engineering Certification Exam

**Questions&AnswersPDF**

**ForMoreInformation:**

**<https://www.certswarrior.com/>**

## **Features:**

- 90DaysFreeUpdates
- 30DaysMoneyBackGuarantee
- InstantDownloadOncePurchased
- 24/7OnlineChat Support
- ItsLatestVersion

**Visit us at: <https://www.certswarrior.com/exam/dbt-analytics-engineering>**

# Latest Version: 6.0

## Question: 1

A team member proposes directly joining multiple raw source tables into a single large model for convenience.

Which of the following is a potential long-term consequence of this approach?

- A. The dbt test command might run slower due to the complex join logic.
- B. Maintenance complexity increases as business logic or source tables change.
- C. Your DAG might inadvertently develop circular dependencies.
- D. Incremental materialization strategies would no longer be applicable.

**Answer: B**

Explanation:

Directly joining raw sources sacrifices modularity and flexibility. While option A is partially true, B highlights a more significant maintainability risk.

## Question: 2

(Multiple Select)

- A. The level of detail needed in downstream reporting.
- B. The desired frequency for updating the data
- C. The number of users currently active on the data warehouse.
- D. Existing model patterns within the project for consistency.

**Answer: A,B**

Explanation:

Reporting needs and data freshness requirements directly shape model design. Consistency (D) is important, but secondary to functional requirements- The number of active users is more relevant to database sizing

## Question: 3

A critical model depends on a third-party data source with periodic update delays. How could you structure your DAG to mitigate the impact of these delays on downstream reporting?

- A. Employ a snapshot of the third-party source to create a historical record.
- B. Utilize the 'defer' option to allow upstream models to run while awaiting the source update.
- C. Configure a 'view' that gracefully handles missing data from the third-party source.

D. Set up an alert to notify users when the upstream data source is not refreshed on time.

**Answer: B**

Explanation:

The 'defer' option lets your DAG continue partially while awaiting the delayed data. Others offer solutions but don't address the core DAG flow issue.

### Question: 4

"Isn't debugging harder with many smaller models in a DAG?" How would you address this concern expressed by a colleague?

- A. "Debugging is slightly harder, but the benefits outweigh the added effort."
- B. "Smaller models are easier to test individually, leading to faster issue identification."
- C. "We'll rely extensively on snapshots to trace errors back to their origin."
- D. "Our DAG will be simple, so debugging model dependencies won't be a concern."

**Answer: B**

Explanation:

Emphasize how modularity isolates problems and enables targeted testing. While the other options have some truth, option B presents the most convincing argument.

### Question: 5

You're setting up a new dbt project and want to customize the naming of compiled objects in the data warehouse. Which section of the dbt\_project.yml file is the most relevant?

- A. seeds:
- B. models:
- C. snapshots:
- D. vars:

**Answer: B**

Explanation:

The models: section, particularly the nested configurations within it, is where you control the naming of materialized models in your database.

### Question: 6

(Multiple Select)

- A. model-paths

- B. test-paths
- C. materialized: view
- D. database

**Answer: A,B**

Explanation:

model-paths and test-paths specify project directory structures- Options C would be within a model configuration, and D could be in a profile.

### Question: 7

You're working with a dbt project in both development and production environments. Which configuration technique is best suited to handle differences in database connection details between these environments?

- A. Define multiple profile sections within the dbt\_project.yml
- B. Utilize environment variables in conjunction with the profiles.yml file.
- C. Use the target flag when running dbt commands.
- D. Create separate branches for each environment and maintain different dbt\_project.yml files in each branch.

**Answer: B**

Explanation:

Environment variables and a corresponding profiles.yml provide the most flexible and secure way to manage environment-specific settings.

### Question: 8

You notice a pattern where warnings are generated for several models due to missing descriptions. How could you configure dbt\_project.yml to fail these models instead when descriptions are absent?

- A. Add on-run-start and on-run-end hooks to perform custom checks
- B. Set a project-level severity setting under the tests: section.
- C. Introduce a custom macro to validate model descriptions\_
- D. Globally change the default warning behavior using the warn-if: configuration

**Answer: D**

Explanation:

The warn-if: configuration lets you escalate specific warnings to errors at the project level.

### Question: 9

A downstream process relies on a critical model to be fully refreshed daily. How can you enforce this behavior using dbt\_project.yml configuration?

- A. Set the model's materialized property to incremental \_ refresh-
- B. Configure a schedule property within the models: section.
- C. Apply a post-hook that triggers a full refresh of the model.
- D. Update the model configuration to materialized: table and add a persist\_docs: true property.

**Answer: C**

Explanation:

Post-hooks enable you to run operations after a model executes. The other options don't provide the required control over the refresh behavior.

### Question: 10

A colleague expresses confusion about how dbt uses the dbt\_project.yml file. Which of the following is the MOST accurate high-level explanation?

- A. The dbt\_project.yml file acts as a blueprint for how dbt should generate your data models, tests, and documentation.
- B. The dbt\_project.yml file contains the actual SQL code that defines your data transformations.
- C. The dbt\_project.yml file is primarily used for version control and collaboration in dbt projects
- D. The dbt\_project.yml file serves as a temporary cache for in-development models.

**Answer: A**

Explanation:

Emphasize the configuration aspect of dbt\_project.yml; it guides dbt's behavior but doesn't contain the SQL logic itself.



# CERTSWARRIOR

## *FULL PRODUCT INCLUDES:*

Money Back Guarantee



Instant Download after Purchase



90 Days Free Updates



PDF Format Digital Download



24/7 Live Chat Support



Latest Syllabus Updates



**For More Information – Visit link below:**

**<https://www.certswarrior.com>**

**16 USD Discount Coupon Code: U89DY2AQ**