**Which feature of Jinja allows you to reuse code across multiple dbt models?**

Choose only ONE best answer.

**A**

Control structures

**B**

Environment variables

**C**

Macros

**D**

Pivoting data

**This answer is correct.**

Answer: C) Macros

Explanation: Jinja, the templating language used by dbt, allows you to create reusable code snippets called macros.

Macros can be defined once and used across multiple dbt models, analyses, tests, and hooks. This can help to reduce the amount of duplicated code in your dbt project and make it easier to maintain.

Control structures and pivoting data are other features of Jinja, but they do not directly address the problem of code reuse.

Environment variables are used to store configuration information that can be accessed from within your dbt project, but they are not directly related to code reuse.

Reference: <https://docs.getdbt.com/docs/build/jinja-macros>

**You are working on a large dbt project with multiple models. You have been asked to improve the readability of the database schema by using model aliasing. Which of the following statements about model aliasing is true?**

Choose only ONE best answer.

**A**

Model aliasing cannot be used in combination with custom schemas.

**B**

Model aliasing can be used to override the default relation identifier for a model.

**C**

Model aliasing is only useful for small projects with a few models.

**D**

Model aliasing cannot be used in a dbt project that uses package dependencies.

**This answer is incorrect. The correct answer is 'B'**

Answer: B

Explanation: Model aliasing is used to override the default relation identifier for a model. It is a useful feature to improve the readability of the database schema and to provide more meaningful names for the tables or views in the database.

Model aliasing can be used in combination with custom schemas to further enhance the clarity and direction of the data warehouse.

Model aliasing is useful for projects of any size and can be used in a dbt project that uses package dependencies.

Reference: <https://docs.getdbt.com/docs/build/custom-aliases>

**In dbt metrics, if you want to group your metrics by a custom dimension like "is\_weekend", what variable must be set in the "dbt\_project.yml" file?**

Choose only ONE best answer.

**A**

is\_weekend\_dimension\_list

**B**

customar\_dimension\_list

**C**

dimension\_group\_list

**D**

metric\_group\_list

**This answer is incorrect. The correct answer is 'B'**

Answer: B

Explanation: To group your metrics by a custom dimension like "is\_weekend" in dbt metrics, you need to set the variable "custom\_calendar\_dimension\_list" in the "dbt\_project.yml" file. This variable allows you to define custom calendar dimensions that can be used to group your metrics for reporting and analysis.



Reference: <https://docs.getdbt.com/docs/build/metrics>

**A data analyst wants to use snapshots in their dbt project to track changes in a dataset. What is the best practice for snapshotting the data in dbt?**

Choose only ONE best answer.

**A**

Snapshot the data after it has been cleaned and transformed by downstream models.

**B**

Snapshot the data in its raw form and use downstream models to clean up the data.

**C**

Snapshot the data after it has been cleaned and transformed, but before it has been joined with other tables.

**D**

C and A

**This answer is correct.**

Answer: b) Snapshot the data in its raw form and use downstream models to clean up the data.

Explanation:

This approach ensures the most accurate representation of the source data and allows for maximum flexibility in working with the data.

Reference: <https://docs.getdbt.com/docs/build/snapshots>

**What is a best practice in SQL for separating logic that cleans up data from logic that transforms data?**

Choose only ONE best answer.

**A**

Using common table expressions (CTEs)

**B**

Using the ref function to build models on top of other models

**C**

Keeping all logic in a single query

**D**

Using the dbt tool to separate logic out into separate models

**This answer is correct.**

Correct Answer: A

Explanation:

As a best practice in SQL, you should separate logic that cleans up your data from logic that transforms your data. You have already started doing this in the existing query by using Common Table Expressions (CTEs). In dbt, it is best practice to separate the logic out into separate models and using the ref function to build models on top of other models, this ensures modularity.  
  
Reference: <https://docs.getdbt.com/docs/get-started/getting-started-dbt-core>

**Alex is working on a dbt project and has defined several hooks for the models in both the dbt\_project.yml file and in the config block of a specific model. He wants to know the order in which the hooks will be executed. How does dbt execute hooks defined in both the dbt\_project.yml file and in the config block of a specific model?**

Choose only ONE best answer.

**A**

Only the hooks defined in the config block of a specific model are executed.

**B**

Only the hooks defined in the dbt\_project.yml file are executed.

**C**

Hooks are executed randomly, regardless of where they are defined.

**D**

Both sets of hooks are executed, and the SQL statements defined in both will be executed in the order they were defined.

**This answer is correct.**

Answer: d) Both sets of hooks are executed, and the SQL statements defined in both will be executed in the order they were defined.

Explanation: When hooks are defined in both the dbt\_project.yml file and in the config block of a specific model, both sets of hooks are executed when the model is built, and the SQL statements defined in both will be executed in the order they were defined. This means that if there are multiple hooks defined for a model, they will be executed one after the other.

Reference: <https://docs.getdbt.com/docs/build/hooks-operations>

**Alex is a data engineer working on a dbt project that uses the merge strategy with a defined unique\_key. He notices that rows matching the unique\_key are not being replaced with new data as expected. Which of the following reasons could explain this issue?**

Choose only ONE best answer.

**A**

The unique\_key is not set up correctly in the configuration.

**B**

The merge strategy is not suitable for handling unique keys.

**C**

The unique\_key must be combined with another strategy for it to work correctly.

**D**

The dbt\_project.yml file is missing or improperly configured.

**This answer is correct.**

Answer: A

Explanation: If rows matching the unique\_key are not being replaced with new data as expected, the most likely reason is that the unique\_key is not set up correctly in the configuration.

When using the merge strategy, dbt should update the target table by merging the new data with the existing data based on the unique\_key.

This strategy is designed to update existing records and insert new records efficiently. If the unique\_key is not properly defined, the merge strategy may not function as intended, leading to the observed issue.

To resolve this, ensure that the unique\_key is correctly defined and used in the merge strategy.

Reference: <https://docs.getdbt.com/docs/build/incremental-models>

**Do you need to create a target schema before running dbt?**

Choose only ONE best answer.

**A**

Yes, you need to create the target schema before running dbt.

**B**

No, you do not need to create the target schema before running dbt. dbt will check if the schema exists when it runs, and create it if it does not exist.

**C**

You only need to create the target schema if you are running dbt on-premises, but not if you are using dbt Cloud.

**D**

It depends on the version of dbt you are using.

**This answer is correct.**

Answer: B

Explanation: When running dbt, you do not need to manually create the target schema beforehand. dbt automatically checks for the existence of the target schema during execution and creates it if it does not exist. This feature helps streamline the development process and manage schema creation more efficiently.

Reference: <https://docs.getdbt.com/docs/build/sources>

**You want to capture the current state of a source table in your dbt project at a specific point in time. Which dbt config should you use?**

Choose only ONE best answer.

**A**

seed

**B**

snapshot

**C**

archive

**D**

model

**This answer is correct.**

Answer: B

Explanation: If you want to capture the current state of a source table in your dbt project at a specific point in time, you should use the snapshot configuration.

Snapshots in dbt allow you to capture and store the state of a table at specified intervals, making it possible to analyze and compare historical data.

Reference: <https://docs.getdbt.com/docs/build/snapshots>

**You are a Analytics engineer working on a dbt project. Which of the following is a benefit of defining tests in dbt?**

Choose only ONE best answer.

**A**

Speed up the development process.

**B**

Allow for more flexibility in the code.

**C**

Improve the user interface of the project.

**D**

Confirm the code's functionality and prevent code regressions.

**This answer is correct.**

Answer: D. Confirm the code's functionality and prevent code regressions

Explanation: Defining tests in dbt helps confirm the code's functionality and prevent code regressions.

Tests can be defined as singular tests written as .sql files in the test directory or as generic tests, which are parameterized queries defined in special test blocks that can be referenced throughout the .yml files.

Both types of tests have their own advantages and can be used depending on the situation.

However, the main benefit of defining tests in dbt is to confirm the code's functionality and prevent code regressions.

Reference: <https://docs.getdbt.com/docs/build/tests>

**In a dbt project, how can you change the default location of the profiles.yml file to a different directory?**

Choose only ONE best answer.

**A**

Modify the dbt\_project.yml file to include a reference to the new directory.

**B**

Use the DBT\_PROFILES\_DIR environment variable to change the default location of the profiles.yml file.

**C**

Include a --profiles-dir option in the dbt\_project.yml file.

**D**

Move the profiles.yml file to the new directory and update the file path in each model's configuration.

**This answer is incorrect. The correct answer is 'B'**

Answer: B

Explanation: There are multiple ways to direct dbt to a different location for your profiles.yml file, but the most appropriate method is to use the DBT\_PROFILES\_DIR environment variable to change the default location.

Specifying this environment variable overrides the directory that dbt looks for your profiles.yml file in. You can specify this by running:

export DBT\_PROFILES\_DIR=path/to/directory.

Reference: <https://docs.getdbt.com/docs/get-started/connection-profiles>

**What is the reason why dbt Python models have limited access to the context of the project?**

Choose only ONE best answer.

**A**

Python models do not support passing context as an argument to the model() function.

**B**

Python models use a different templating language that does not have access to the project's context.

**C**

Jinja, the templating language used by dbt SQL models, provides access to the project's context, which is not available to Python models.

**D**

The context is not relevant or necessary for Python models to perform their functions.

**This answer is correct.**

Answer: C

Explanation: The reason why dbt Python models have limited access to the context of the project is that they do not use Jinja, a templating language that is used by dbt SQL models to render compiled code and provides access to the project's context.

Instead, the context is made available from the dbt class and passed in as an argument to the model() function.

Reference: <https://docs.getdbt.com/docs/build/python-models>

**In dbt, which method is used to configure a Python model within the model's .py file?**

Choose only ONE best answer.

**A**

dbt.configure()

**B**

dbt.setup()

**C**

dbt.config()

**D**

dbt.define()

**This answer is correct.**Answer: C  
  
Explanation: The dbt.config() method is used to configure a dbt Python model within the model's .py file. Therefore, option C is correct, while the other options are incorrect.  
  
Reference: https://docs.getdbt.com/docs/build/python-models

**How can you specify a schema other than the target schema in your profiles.yml for building models in dbt, and where can you specify this information?**

Choose only ONE best answer.

**A**

You can specify the schema in the dbt\_project.yml file using the model-paths configuration.

**B**

You can specify the schema in the model file using the ref function.

**C**

You can specify the schema in the dbt\_project.yml file using the schema configuration block, or in the model file using a config block.

**D**

You can specify the schema in the model file using the schema keyword.

**This answer is incorrect. The correct answer is 'C'**

Answer: C

Explanation: To specify a schema other than the target schema for building models in dbt, you can use the schema configuration block in the dbt\_project.yml file, or a config block in the model file.

Reference: <https://docs.getdbt.com/docs/build/sources>

**Which of the following is a valid use case for using variables in dbt models?**

Choose only ONE best answer.

**A**

Generating reports and visualizations from SQL queries

**B**

Storing configuration data for use in dbt projects

**C**

Defining data transformations in a graphical user interface

**D**

Configuring timezones or avoiding hardcoded table names

**This answer is incorrect. The correct answer is 'D'**

Answer: D) Configuring timezones or avoiding hardcoded table names

Explanation: One of the primary use cases for using variables in dbt models is to configure timezones or avoid hardcoded table names. By using variables for these purposes, you can make your dbt models more flexible and adaptable to changing requirements.

Reference: <https://docs.getdbt.com/docs/build/project-variables>

**What will happen if you specify threads: 1 when running dbt?**

Choose only ONE best answer.

**A**

dbt will start building only one model, and finish it, before moving onto the next model.

**B**

dbt will start building all models at once, without finishing any of them.

**C**

dbt will start building all models in a sequential order

**D**

dbt will start building only one model, and finish it, before moving onto the next model, but the run time of the project will increase.

**This answer is correct.**

Explanation:

When you specify threads: 1 in your dbt project configuration, it means that you are restricting the number of threads that can be used to execute your dbt commands to 1. This implies that dbt will run only one operation at a time, i.e., build one model at a time, and wait for the completion of the current operation before proceeding to the next one.

Reference: <https://docs.getdbt.com/docs/core/connection-profiles#understanding-threads>

**Which of the following types of files are best suited for loading using dbt's seed functionality, according to dbt documentation?**

Choose only ONE best answer.

**A**

Large CSV exports from a production database

**B**

Raw data files in JSON or XML format

**C**

Files containing business-specific logic, such as a list of country codes or employee IDs

**D**

Files containing sensitive information, such as PII or passwords

**This answer is correct.**

Answer: C

Explanation: According to dbt documentation, seeds are version controlled and maintainable, and are best suited to files containing business-specific logic, such as a list of country codes or employee IDs. Seeds should not be used to load raw data, as this should be loaded using an ETL tool or SQL script, and large CSVs are not performant when loaded using dbt's seed functionality. Similarly, files containing sensitive information should not be stored in seeds, but instead should be secured and protected using appropriate security measures.

Reference: <https://docs.getdbt.com/docs/build/seeds>

**You are a data scientist working on a data pipeline and you want to write a model in dbt that doesn't directly build into the database. Which materialization option should you use for this purpose?**

Choose only ONE best answer.

**A**

Table

**B**

View

**C**

Incremental

**D**

Ephemeral

**This answer is correct.**

Answer: D. Ephemeral

Explanation: The ephemeral materialization in dbt allows you to write reusable logic without directly building the model into the database. Instead, the code from this model will be used as a common table expression in dependent models.

This materialization option is best suited for very lightweight transformations that don't need to be directly queried, and are used only in one or two downstream models. If you want to write a model in dbt that doesn't directly build into the database, you should use the ephemeral materialization.

Reference: <https://docs.getdbt.com/docs/build/materializations>

**As a data engineer, you are concerned about securely storing credentials for your dbt project. What method is considered more secure for storing credentials in dbt?**

Choose only ONE best answer.

**A**

Storing credentials directly in the profiles.yml file.

**B**

Using environment variables to load credentials.

**C**

Saving credentials in a separate plain-text file within the project directory.

**D**

Storing credentials in the dbt\_project.yml file.

**This answer is correct.**

Answer: B

Explanation: Using environment variables to load credentials is considered more secure than storing them directly in the profiles.yml file or other project files. By storing sensitive information in environment variables, you minimize the risk of accidentally exposing your credentials, as they are not saved within the project's files or version control system.

Reference:  <https://docs.getdbt.com/docs/core/connection-profiles>

**What is the purpose of the --vars command line option in dbt?**

Choose only ONE best answer.

**A**

To define variables in a dbt project

**B**

To override variables for a run of dbt

**C**

To configure timezones and avoid hardcoded table names

**D**

To generate reports and visualizations from SQL queries

**This answer is correct.**Answer: B) To override variables for a run of dbt  
  
Explanation: The --vars command line option in dbt is used to override variables for a specific run of dbt. By providing a YAML dictionary as a string, you can set or override variables that are defined in the dbt\_project.yml file or in package-specific files. This can be useful for setting variables that change frequently, such as environment variables or user-specific variables.  
  
Reference: https://docs.getdbt.com/docs/build/project-variables

**When implementing data integrity checks in dbt, which of the following methods should be avoided?**

Choose only ONE best answer.

**A**

Running data integrity tests against the data warehouse

**B**

Running data integrity tests against individual models

**C**

Automating data integrity checks

**D**

Performing manual checks on the data

**E**

None of the above

**This answer is incorrect. The correct answer is 'B'**Answer: B) Running Data Integrity Tests Against Individual Models.  
  
In Dbt When Using Sources It Is Best Adviced To Perform And Running Data Intergrity Tests Against The Sources Rather Than Against Individual Models  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Getting-Started/Building-Your-First-Project/Test-And-Document-Your-Project

**Which of the following statements are true about the ref function in dbt?**

Choose only ONE best answer.

**A**

The ref function allows you to build models on top of other models

**B**

The ref function allows you to modularize your models

**C**

The ref function allows you to make your models reuseable

**D**

All of the above

**This answer is incorrect. The correct answer is 'D'**Answer: D. All Of The Above.  
  
In Dbt We Modularize And Make Our Models Reuseable By Using The Ref Function To Build Models On Top Of Other Models.  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Getting-Started/Building-Your-First-Project/Build-Your-First-Models

**A company wants to use the check strategy to detect changes in their product inventory. Which configuration must be specified to use the check strategy in dbt?**

Choose only ONE best answer.

**A**

updated\_at

**B**

check\_cols

**C**

unique\_key

**D**

None of the above

**This answer is correct.**Answer: b. check\_cols, A list of columns to check for changes, or all to check all columns  
  
Reference: https://docs.getdbt.com/docs/build/snapshots

**Which of the following best describes how dbt optimizes your workflow?**

Choose only ONE best answer.

**A**

By creating a new database with each dbt run

**B**

By processing raw data in real-time

**C**

By building up reusable, modular data models

**D**

By storing data in a more efficient format like avro and parquet

**This answer is correct.**Answer: C) By Building Up Reusable, Modular Data Models.  
  
Dbt Optimizes Your Workflow By Building Up Reusable, Or Modular, Data Models That Can Be Referenced In Subsequent Work Instead Of Starting At The Raw Data With Every Analysis.  
  
Reference: Https://Docs.Getdbt.Com/Docs/Introduction

**A team is working on a dbt project where they are using snapshots to track changes to their product inventory. They want to prevent accidental deletion of their snapshots. What is the recommended approach for setting privileges on the snapshot tables?**

Choose only ONE best answer.

**A**

Run the snapshots as the same user or role as other models in the warehouse

**B**

Run the snapshots as a different user or role in the warehouse

**C**

Don't set any privileges on the snapshots

**D**

Give all users full access to the snapshots

**This answer is correct.**Answer: B. Run the snapshots as a different user or role in the warehouse.  
  
This is considered a best practice for snapshot configurations, as it helps to ensure that the snapshots remain intact and continue to provide an accurate record of the data over time. By running the snapshots as a different user or role, it makes it more difficult to accidentally delete a snapshot.  
  
Reference: https://docs.getdbt.com/docs/build/snapshots

**When using dbt to test a model, what does dbt do with the queries it constructs for each test?**

Choose only ONE best answer.

**A**

It saves them to the YAML file for future reference.

**B**

It discards them after the test is run.

**C**

It executes them and returns the number of records that failed the test.

**D**

It compiles them into a single executable query.

**This answer is correct.**Answer: C. It Executes Them And Returns The Number Of Records That Failed The Test  
  
When Building Models With Dbt We Generally Add Tests To Ensure Our Models Are Working As They Should. To Add Tests In Dbt We Add Them To Thw Yaml File For The Models We Want To Test. After Adding The Preferred Test To The Yaml File We Can Run "Dbt Test" To Confirm All Tests Are Passed. When We Run 'Dbt Test', Dbt Iterates Through The Yaml File, And Constructs A Query For Each Test. Each Query Will Return The Number Of Records That Failed The Test. If This Number Is 0, Then The Test Is Successful.  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Getting-Started/Building-Your-First-Project/Test-And-Document-Your-Project

**What command should you use to initiate a new dbt project?**

Choose only ONE best answer.

**A**

dbt new dbt\_project\_name

**B**

dbt init dbt\_project\_name

**C**

dbt create dbt\_project\_name

**D**

dbt start dbt\_project\_name

**This answer is correct.**

Correct Answer: B

Explanation: To initiate a new dbt project, you use the "dbt init dbt\_project\_name" command. This command creates a new directory with the specified project name and sets up the necessary files and directories for a dbt project.

Reference: <https://docs.getdbt.com/docs/get-started/getting-started-dbt-core>

**Sarah is a data analyst who works with a dbt Cloud project. She needs to define an environment variable to store the database password. Which prefix should she use for the variable name to define it as an environment secret variable in dbt Cloud?**

Choose only ONE best answer.

**A**

DBT

**B**

DBT\_ENV

**C**

DBT\_ENV\_SECRET

**D**

DBT\_SECRET

**This answer is correct.**Answer: C  
  
Explanation: In dbt Cloud, environment secret variables need to be prefixed with DBT\_ENV\_SECRET\_.   
  
  
Reference: https://docs.getdbt.com/docs/build/project-variables

**You are reviewing a dbt project that has a source schema defined in the schema.yml file. The source schema has a name in the database that looks like machine generated name which is not easily readable, you want to use a more sensible name in dbt. How can you implement this and still ensure dbt compiles using the actual source schema name?**

Choose only ONE best answer.

**A**

Use DDL and DML statements to change the schema nameto one that can be easily read and makes sense

**B**

Use the schema property to define the actual names as per the database, and use your name: property for the name that makes sense and is easily readable

**C**

Use the 'database' property to define the actual names as per the database, and use your name: property for the name that makes sense and is easily readable

**D**

Use the 'table' property to define the actual names as per the database, and use your name: property for the name that makes sense and is easily readable

**This answer is incorrect. The correct answer is 'B'**Answer: B  
  
Explanation: When you define a source in the schema.yml file, you can use the 'name' property to provide a more meaningful name for the source. To use the source in a downstream model, you can construct the source function using the 'name' property and it gets compiled using the actual name of the schema.

Reference: https://docs.getdbt.com/docs/build/sources

**As a data engineer working on a dbt project, you need to install a package from a tarball URL hosted internally within your organization. What information do you need to provide in order to install this package?**

Choose only ONE best answer.

**A**

The tarball URL and the package version.

**B**

The tarball URL and the subfolder name where the package source code is installed.

**C**

The tarball URL and the branch name of the package repository.

**D**

The tarball URL and the name of the data warehouse used in the project.

**This answer is correct.**

Answer: B

Explanation: To install packages using tarball URLs hosted internally, you need to specify the URL of the tarball and the subfolder name where the package source code is installed within. This information will allow dbt to fetch and install the package from the internal location, ensuring that the correct version and source code are used in your project.

Reference: https://docs.getdbt.com/docs/build/packages

**Maria is working on a dbt project and needs to create snapshots for a table that does not have a reliable updated\_at column. Which snapshot strategy should she use to compare a list of columns between their current and historical values?**

Choose only ONE best answer.

**A**

Timestamp strategy

**B**

Merge strategy

**C**

Check strategy

**D**

Incremental strategy

**This answer is incorrect. The correct answer is 'C'**

Answer: C

Explanation: Use the check strategy for tables that do not have a reliable updated\_at column. This strategy works by comparing a list of columns between their current and historical values. If any of these columns have changed, then dbt will invalidate the old record and record the new one. If the column values are identical, then dbt will not take any action.

Reference: <https://docs.getdbt.com/docs/build/snapshots>

**You are running dbt and want to check the connection to your warehouse. What should you do?**

Choose only ONE best answer.

**A**

Use the debug command within the dbt project

**B**

Use the test command within the dbt project

**C**

Use the run command within the dbt project

**D**

Use the check command within the dbt project

**This answer is incorrect. The correct answer is 'A'**Explanation:   
  
Use The Debug Command To Check Whether You Can Successfully Connect To Your Warehouse. Simply Run Dbt Debug From Within A Dbt Project To Test Your Connection.  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Connection-Profiless

**What is the purpose of the dbt analysis folder?**

Choose only ONE best answer.

**A**

To store SQL queries for use in dbt models.

**B**

To store SQL queries for use in dbt tests.

**C**

To organize analytical SQL queries in a dbt project.

**D**

To store SQL queries for use in dbt seeds.

**This answer is incorrect. The correct answer is 'C'**Answer: C Explanation: The purpose of the dbt analysis folder is to organize analytical SQL queries in a dbt project. These queries are not run by dbt when you initiate the dbt run command, but can be compiled using the dbt compile command. This feature provides a way to version control your non-model SQL queries and keep them organized alongside your other dbt project components. Option A is incorrect because SQL queries for use in dbt models are stored in the models folder. Option B is incorrect because SQL queries for use in dbt tests are stored in the tests folder. Option D is incorrect because SQL queries for use in dbt seeds are stored in the seeds folder. Reference: https://docs.getdbt.com/docs/build/projects

**A data analyst wants to create a snapshot configuration to track changes in a customer table that contains sensitive information. They want to ensure that the snapshot configuration provides the best handling of changes to the source data, including the addition and deletion of columns. Which strategy should they use when working with snapshot tables in dbt?**

Choose only ONE best answer.

**A**

The timestamp strategy

**B**

The check\_cols strategy

**C**

The unique key strategy

**D**

The hard delete strategy

**This answer is incorrect. The correct answer is 'A'**Answer: a) The timestamp strategy.  
  
The timestamp strategy is recommended for snapshot configurations when working with snapshot tables in dbt. It provides better handling of changes to the source data, including additions and deletions of columns, compared to the check\_cols strategy. Using the timestamp strategy is considered a best practice for snapshot configurations.  
  
Reference: https://docs.getdbt.com/docs/build/snapshots

**Which of the following statements is true about the Python model in dbt?**

Choose only ONE best answer.

**A**

The Python model is stored in a database table.

**B**

The Python model is stored in a .py file in the models/ folder.

**C**

The Python model is a class compiled by dbt Core.

**D**

The Python model defines a function named dbt().

**This answer is incorrect. The correct answer is 'B'**Answer: B  
  
Explanation: According to the given text, the Python model in dbt is stored in a .py file in the models/ folder. The other options are incorrect because the text explicitly states that the Python model is not stored in a database table, it is not a class compiled by dbt Core, and it defines a function named model() and not dbt().  
  
Reference: https://docs.getdbt.com/docs/build/sources

**How can using dbt sources improve data analysis workflows?**

Choose ALL answers that apply.

* **A**

It allows for easy dependency tracing

* **B**

It allows for source freshness reporting

* **C**

It allows for better data visualization

* **D**

It allows for easier data cleansing

**This answer is incorrect. The correct answer is 'A' 'B'**Answer: A, B  
  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Learning-More/Refactoring-Legacy-Sql

**A data team wants to reduce the number of columns that need to be checked in their database when using the check strategy in dbt. What is the recommended technique they can use to achieve this?**

Choose only ONE best answer.

**A**

Include all columns in the check\_cols configuration

**B**

Exclude columns from the check\_cols configuration

**C**

Use a surrogate key to condense many columns into a single column

**D**

None of the above

**This answer is incorrect. The correct answer is 'C'**Answer: c. Use a surrogate key to condense many columns into a single column

**You have created a metric that uses multiple dimensions, but when you run the metric, the resulting number is not providing useful information. What is the most likely reason for this issue?**

Choose only ONE best answer.

**A**

The metric is not defined correctly in the dbt project.

**B**

The dimensions being used in the metric calculation are not relevant to the data being analyzed.

**C**

The table being used to calculate the metric does not contain enough data to make the calculation meaningful.

**D**

The metric is being calculated using an incorrect formula.

**This answer is correct.**Answer: B  
  
Explanation: When a metric is created in dbt that uses multiple dimensions, it takes all the information in the table and simplifies it into a single number that gives you a quick understanding of the data.  
  
Reference: https://docs.getdbt.com/docs/build/metrics

**What is the command used to specify a target other than the default when using DBT?**

Choose only ONE best answer.

**A**

--target

**B**

--prod

**C**

--dev

**D**

--target\_schema

**This answer is incorrect. The correct answer is 'A'**Correct Answer: A. "--Target"  
  
Explanation:  "Users Can Also Use The "--Target" Option When Issuing A Dbt Command To Use A Target Other Than The Default."   
  
Dbt Supports Multiple Targets Within A Profile, Which Encourages The Use Of Separate Development And Production Environments. When Developing In Dbt Locally It Is Always Good Practice To Use The Dev Target Usually Set As The Default. A Separate "Prod" Target Can Also Be Created For Production Environments. Users Can Also Use The "--Target" Option When Issuing A Dbt Command To Use A Target Other Than The Default.  
  
Reference: Https://Docs.Getdbt.Com/Docs/Get-Started/Connection-Profiless

**In a dbt Python model, what is the concept of "lazy evaluation"?**

Choose only ONE best answer.

**A**

The ability to preview data using methods like .show() or .head() in development.

**B**

The ability to execute Python code remotely on a data platform.

**C**

The ability to create a series of meaningful transformations using CTEs.

**D**

The output of each dataframe operation not being immediately calculated, but only computed when explicitly asked for.

**This answer is incorrect. The correct answer is 'D'**Answer: D  
  
Explanation: In a dbt Python model, the concept of "lazy evaluation" refers to the output of each dataframe operation not being immediately calculated. Instead, the operations are only computed when you explicitly ask for the final result of the data. In development, you can preview the data using methods like .show() or .head(). When you run a Python model, the full result of the final DataFrame will be saved as a table in your data warehouse.  
  
Option A is incorrect because previewing data in development is not the same as "lazy evaluation".  
  
Option B is incorrect because remote execution is not necessarily related to "lazy evaluation".  
  
Option C is incorrect because CTEs are a separate concept from "lazy evaluation".  
  
Reference: https://docs.getdbt.com/docs/build/sources

**How can you specify column types in dbt?**

Choose only ONE best answer.

**A**

Use the type configuration block in the dbt\_project.yml file to specify the column types for each model.

**B**

Use the dtype function to specify column types in the SQL select statement.

**C**

Cast the column to the correct type in your model using SQL syntax.

**D**

Use the column\_type configuration block in the model file to specify the column types for each column.

**This answer is incorrect. The correct answer is 'C'**Answer: C  
  
Explanation: In dbt, you can specify column types by casting the column to the correct type in your model using SQL syntax. For example, if you want to cast a column called age to an integer type, you could write: select cast(age as integer) as age from my\_table This will ensure that the age column is treated as an integer by downstream tools and models.  
  
Option A is incorrect because the type configuration block in the dbt\_project.yml file is used to specify the types of sources and targets, not the types of columns in a model.  
  
Option B is incorrect because there is no dtype function in dbt.  
  
Option D is incorrect because there is no column\_type configuration block in dbt.  
  
Reference: https://docs.getdbt.com/docs/build/sources

**What happens when dbt encounters an error in the SQL code of a model during execution?**

Choose only ONE best answer.

**A**

dbt returns the error generated by the database or data warehouse, and downstream models are skipped.

**B**

dbt continues running the remaining models and ignores the model with the error.

**C**

dbt automatically fixes the error and reruns the model.

**D**

dbt stops execution immediately and does not provide any error message.

**This answer is correct.**

Answer: A

Explanation: When dbt encounters an error in the SQL code of a model during execution, it will return the error that the database or data warehouse returns, and any models downstream of this model will also be skipped. To debug any errors, the dbt documentation suggests using the error message and the compiled SQL.

Reference: <https://docs.getdbt.com/docs/get-started/getting-started-dbt-core>

**You are a data platform engineer working with a Python model that requires incremental updates. What materialization options do you have for Python models in dbt?**

Choose only ONE best answer.

**A**

View and incremental

**B**

Table and view

**C**

Table and incremental

**D**

Ephemeral and table

**This answer is correct.**Answer: C. Table and incremental  
  
  
Explanation: In dbt, Python models have two materialization options: table and incremental. Incremental models in Python support the same incremental strategies as SQL models, but the specific strategies depend on the database adapter being used. It is not possible to use view or ephemeral materialization for Python models, nor can Python be used for non-model resources like tests and snapshots. Therefore, for a Python model that requires incremental updates, you can use either the table or incremental materialization. The incoming data must be filtered to only include new rows. The insert\_overwrite strategy for incremental models is not yet supported for BigQuery/Dataproc, but the merge incremental strategy is supported.  
  
Reference: https://docs.getdbt.com/docs/build/materializations

**Imagine you are a data engineer working for a company that deals with massive amounts of data. You have been tasked with optimizing the data transformation process to reduce the time it takes to process the data.You have been using dbt as your primary tool for data transformation, but you have noticed that the process is taking too long when working with large data volumes. You have heard about a feature in dbt called "incremental\_predicates" that can help improve performance in such scenarios. How does this dbt feature does this?**

Choose only ONE best answer.

**A**

The "incremental\_predicates" feature in dbt optimizes the incremental build process for large data volumes by using Tree based algorithms to determine changed data and update only those records, resulting in improved performance and reduced processing time for data transformations.

**B**

The "incremental\_predicates" feature in dbt optimizes the incremental build process for large data volumes by using specified SQL expressions to determine changed data and update only those records, resulting in improved performance and reduced processing time for data transformations.

**C**

The "incremental\_predicates" feature in dbt optimizes the incremental build process for large data volumes by using specified Jinja control structures to determine changed data and update only those records, resulting in improved performance and reduced processing time for data transformations.

**D**

dbt has no incremental\_predicates feature, dbt uses pre-hook and post-hooks to determine changed data and update only those records, resulting in improved performance and reduced processing time for data transformations.

**This answer is incorrect. The correct answer is 'B'**Answer: B)   
Explanation: The "incremental\_predicates" feature in dbt is used to improve performance when working with large data volumes. It allows for a list of SQL expressions to be specified that will be used to optimize the incremental build process. By specifying these expressions, dbt can better determine which data has changed and only update those records instead of reprocessing the entire dataset. This can significantly improve performance and reduce the time it takes to run data transformations on large datasets.  
  
Reference: https://docs.getdbt.com/docs/build/incremental-models

**You have an incremental model in dbt with the default setting "on\_schema\_change: ignore", but when running the dbt command, the new column you added is not appearing in the target table. What could be the cause of this issue?**

Choose only ONE best answer.

**A**

The database does not support adding new columns in this way.

**B**

The on\_schema\_change parameter is not set correctly in the configuration.

**C**

A full-refresh of both the incremental model and any related models has not been executed.

**D**

There is an issue with the SQL used in the model.

**This answer is correct.**Correct answer: c) A full-refresh of both the incremental model and any related models has not been executed.  
  
Explanation: When using an incremental model in dbt with the "on\_schema\_change: ignore" setting, any changes made to the columns in the incremental model will not be reflected in the target table until a full-refresh of both the incremental model and any related models is executed. This means that if you add a column to the incremental model, it will not appear in the target table until a full-refresh is executed. Therefore, the most likely cause of this issue is that a full-refresh of both the incremental model and any related models has not been executed.  
  
Reference: https://docs.getdbt.com/docs/build/incremental-models

**How do exposures help data consumers understand and use the outputs of a dbt project?**

Choose only ONE best answer.

**A**

By providing a dedicated page in the auto-generated documentation site with context relevant to the outputs.

**B**

By creating a separate schema for each user to maintain separate development and production environments.

**C**

By enabling you to test and run resources that feed into the exposure.

**D**

By defining and describing the upstream use of the project.

**This answer is correct.**Answer: A  
  
Explanation: Exposures in dbt help data consumers understand and use the outputs of a project by providing a dedicated page in the auto-generated documentation site with context relevant to the outputs. This allows data consumers to easily access information about the outputs and how they can be used in downstream applications, dashboards, or data science pipelines.  
  
Option B is incorrect because exposures do not create a separate schema for each user.  
  
Option C is incorrect because testing and running resources that feed into an exposure is only one of the benefits of using exposures, and does not directly help data consumers understand and use the outputs.  
  
Option D is incorrect because exposures define and describe the downstream use of the project, not the upstream use.  
  
Reference: https://docs.getdbt.com/docs/build/exposures

**Select the best reason for using dbt sources rather than referencing raw database tables:**

Choose ALL answers that apply.

* **A**

It allows for easy dependency tracing

* **B**

It improves performance of SQL queries

* **C**

It allows for version control of data sources

* **D**

It improves data visualization capabilities

**This answer is incorrect. The correct answer is 'A' 'C'**

Answer: A,C

Implement Dbt Sources Rather Than Referencing Raw Database Tables: During Legacy To Dbt Migration Once We Can Confirm That Our Sql Code Has Worked And Materialized To The Data Warehouse Or Platform, Its Time To Convert All The Dataware House Sources To Dbt Sources.

The Recommended Practice For Querying The Data Warehouse From Dbt Is To Use Dbt Sources, This Allows Us To Call The Same Table In Multiple Places Using The {{ Src('My\_Source', 'My\_Table') }} Rather Than My\_Database.My\_Schema.My\_Table.

This Is A Best Practice Because;

Source Freshness Reporting: Using Sources Unlocks The Ability To Run Source Freshness Reporting To Make Sure Your Raw Data Isn'T Stale.

Easy Dependency Tracing: If You'Re Migrating Multiple Stored Procedures Into Dbt, With Sources You Can See Which Queries Depend On The Same Raw Tables. This Allows You To Consolidate Modeling Work On Those Base Tables, Rather Than Calling Them Separately In Multiple Places.

Build The Habit Of Analytics-As-Code: Sources Are An Easy Way To Get Your Feet Wet Using Config Files To Define Aspects Of Your Transformation Pipeline. With A Few Lines Of Code In A Schema .Yml File In Your Dbt Project'S /Models Subfolder, You Can Now Version Control How Your Data Sources (Snowplow, Shopify, Etc) Map To Actual Database Tables.

**What are the limitations of the dbt.config() method in terms of the types of arguments that can be passed to it?**

Choose only ONE best answer.

**A**

It can accept any type of data structure or function.

**B**

It can only accept strings and numbers.

**C**

It can only accept basic data types such as strings, booleans, and numbers.

**D**

It can accept any data type as long as it is defined in a YAML file.

**This answer is incorrect. The correct answer is 'C'**Answer: C  
  
Explanation: The dbt.config() method can only accept basic data types such as strings, booleans, and numbers as arguments. Therefore, option C is correct, while the other options are incorrect.  
  
Reference: https://docs.getdbt.com/docs/build/python-models

**A healthcare company wants to keep a historical record of changes to their patient data, such as changes to their medical history and medication information. which feature of dbt is the best for this scenario?**

Choose only ONE best answer.

**A**

dbt historical cannot be used for tracking changes in patient data as it violates HIPAA regulations.

**B**

dbt snapshots can be used to keep a historical record of changes to patient data, including medical history and medication information.

**C**

dbt data lineage can only be used for tracking changes to dimension tables in data warehousing and cannot be used in healthcare settings.

**D**

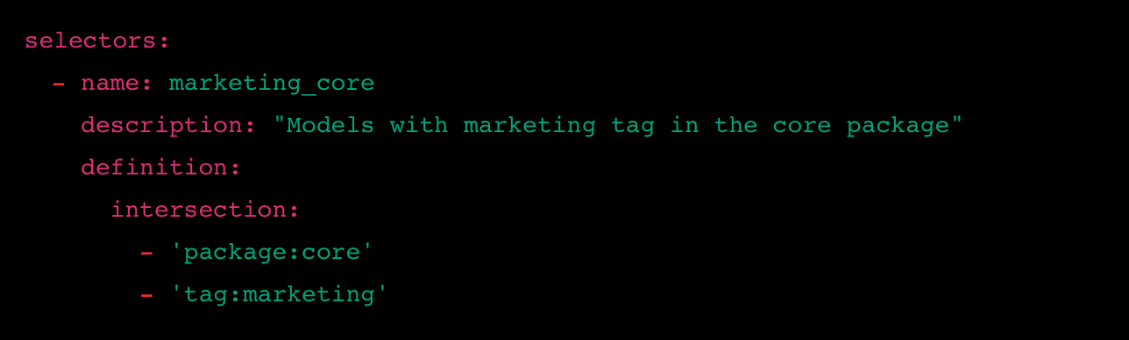
dbt incremental can be used for tracking changes to patient data but only if the patient has given their explicit consent.

**This answer is correct.**Answer: B.  
  
dbt snapshots can be used to keep a historical record of changes to patient data, including medical history and medication information. By using snapshots, the healthcare company can implement type-2 Slowly Changing Dimensions (SCDs) to track changes to rows in mutable source tables over time, making it easier to track the evolution of patient data over time.  
  
Reference: https://docs.getdbt.com/docs/build/snapshots

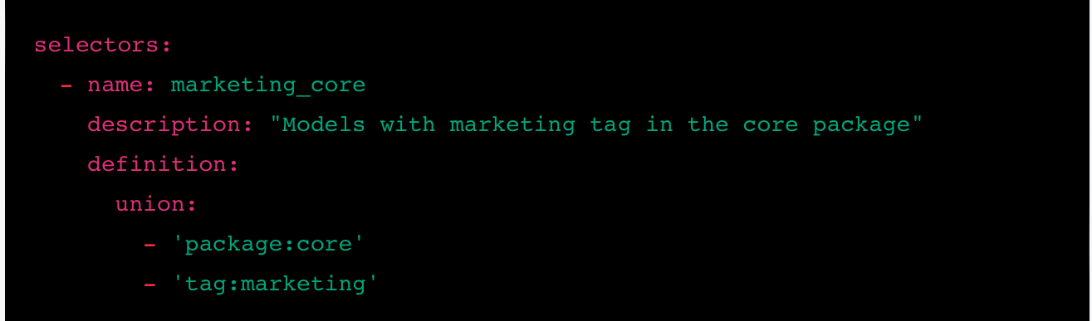
**You are working on a dbt project and want to select models that have a tag of "marketing" and are part of the "core" package. Which of the following YAML selectors will achieve this?**

Choose only ONE best answer.

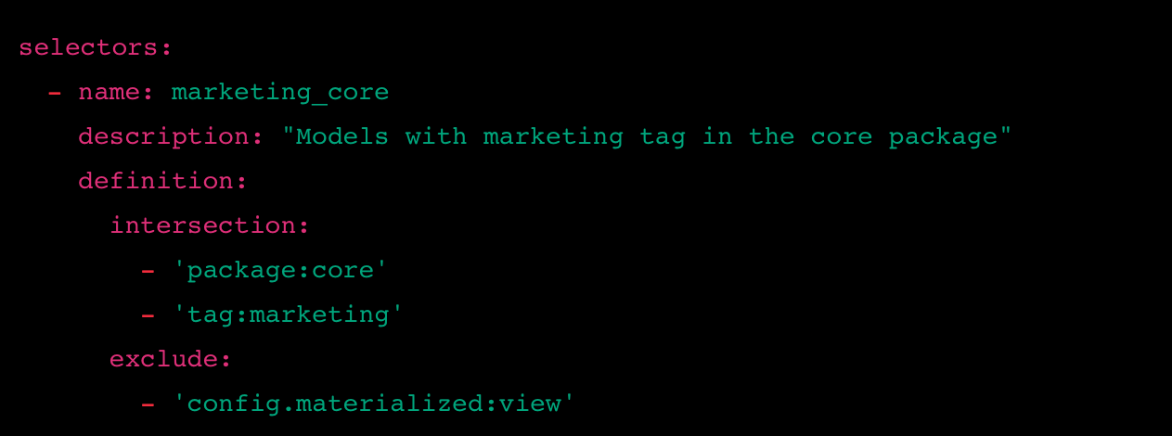
**A**



**B**



**C**



**This answer is incorrect. The correct answer is 'A'**

Answer: A

Explanation:

Selector A uses the 'intersection' clause to select models that meet both conditions: being part of the 'core' package and having a 'marketing' tag.

This fulfills the requirement of selecting models with a tag of "marketing" and part of the "core" package.

Selector B incorrectly uses the 'union' clause, which will select models that meet either condition, not both.

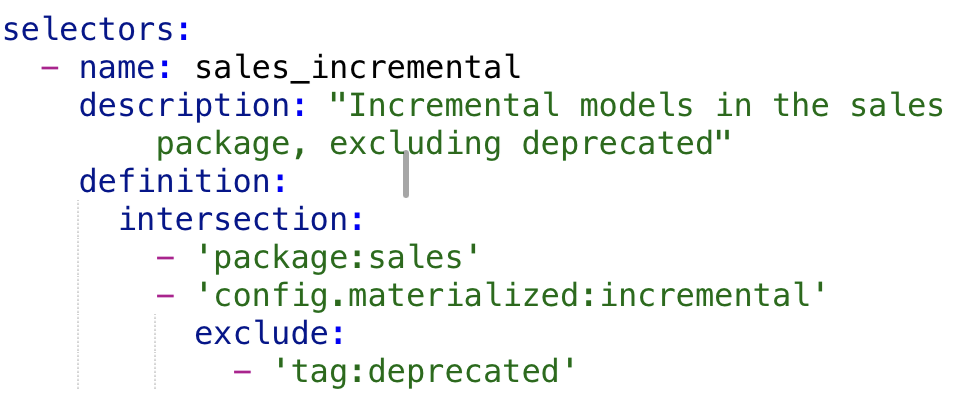
Selector C also includes an 'exclude' clause for models with a materialization of 'view', which is not required by the question.

Reference: <https://docs.getdbt.com/reference/node-selection/yaml-selectors>

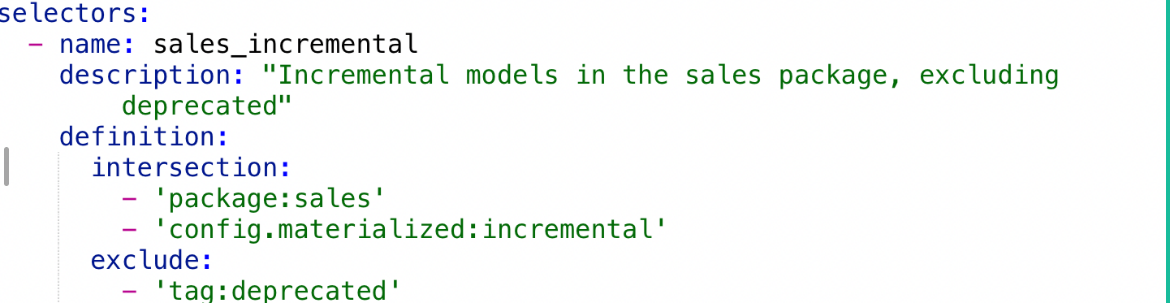
**You want to select models in the "sales" package that have a materialization setting of "incremental" but exclude those tagged as "deprecated". Which of the following YAML selectors will achieve this?**

Choose only ONE best answer.

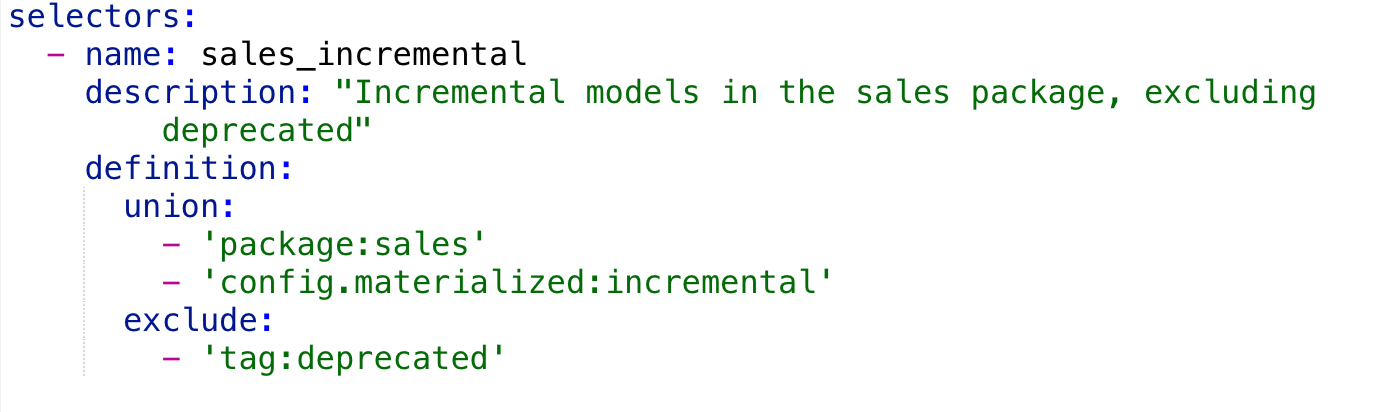
**A**



**B**



**C**



**This answer is incorrect. The correct answer is 'B'**

Answer: B

Explanation:

Selector A incorrectly nests the 'exclude' clause within the 'intersection' clause, which will result in a syntax error.

Selector B uses the 'intersection' clause to select models that are part of the 'sales' package and have a materialization setting of 'incremental'. Then, it uses the 'exclude' clause to remove models with a 'deprecated' tag from the selection. This meets the requirements of the question.

Selector C incorrectly uses the 'union' clause, which will select models that meet either the package or materialization condition, not both.

**The marketing team later requests that you exclude a specific model, customer\_lifetime\_value, while still executing the other customer\_analytics tagged models and their dependencies. How would you modify your dbt command to accommodate this request?**

Choose only ONE best answer.

**A**

dbt run --select tag:customer\_analytics+ --select -customer\_lifetime\_value

**B**

dbt run --select tag:customer\_analytics+ --exclude customer\_lifetime\_value

**C**

dbt run --select tag:customer\_analytics+ --select -+customer\_lifetime\_value

**D**

dbt run --select tag:customer\_analytics+ -customer\_lifetime\_value

**This answer is incorrect. The correct answer is 'B'**

**You have been asked to run a specific set of models related to the daily financial reports, but you need to exclude models tagged as "nightly" and those with a materialized configuration set to "table.”What command would you use to run the models contained in the "daily\_reports" directory, excluding those with the "nightly" tag and materialized as "table" ?**

Choose only ONE best answer.

**A**

dbt run --select path:models/daily\_reports --exclude tag:nightly --exclude config.materialized:table

**B**

dbt run --select path:models/daily\_reports --select -tag:nightly,config.materialized:table

**C**

dbt run --select path:models/daily\_reports,tag:nightly,config.materialized:table

**D**

dbt run --select path:models/daily\_reports --select -tag:nightly --select -config.materialized:table

**This answer is correct.**

dbt run --select path:models/daily\_reports --exclude tag:nightly,config.materialized:table  
  
  
  
Explanation: This command will run the models contained in the "daily\_reports" directory while excluding models with the "nightly" tag and those with a materialized configuration set to "table." The "--select" flag is used to specify the path to the models you want to include, and the "--exclude" flag is used to specify the criteria for models you want to exclude.

**What is the main purpose of the source\_status method in dbt?**

Choose only ONE best answer.

**A**

To calculate the total execution time of dbt commands

**B**

To evaluate the freshness of data sources and apply selectors based on the result

**C**

To optimize query performance

**D**

To manage user permissions and access control

**This answer is correct.**

Answer: B

Explanation: The source\_status method is used to evaluate the freshness of data sources by comparing the current state with the previous state. It allows users to apply selectors based on the freshness of the sources, which can be useful in ensuring that the data used in the transformation is up-to-date.

**Which of the following dbt commands produces a sources.json artifact that can be referenced in subsequent dbt invocations?**

Choose only ONE best answer.

**A**

dbt source new\_source

**B**

dbt test freshness

**C**

dbt source freshness

**D**

dbt run freshness

**This answer is correct.**

Answer: C

Explanation: The dbt source freshness command produces a sources.json artifact that contains information about the execution times and max\_loaded\_at dates for dbt sources. This artifact can be used to reference the source freshness results in subsequent dbt invocations.

**What is the primary function of the dbt source freshness command?**

Choose only ONE best answer.

**A**

To generate a list of all available data sources

**B**

To update the data sources to their latest versions

**C**

To evaluate the freshness of the data sources

**D**

To create a new source for the dbt project

**This answer is incorrect. The correct answer is 'C'**

Answer: C

Explanation: The dbt source freshness command is used to evaluate the freshness of the data sources by comparing the current state with the previous state. This helps users ensure that the data used in the transformation is up-to-date.

**How can you reference the source freshness results in a subsequent dbt command?**

Choose only ONE best answer.

**A**

Use the --source-status flag

**B**

Use the --select flag with the source\_status selector

**C**

Use the --refresh flag

**D**

Use the --state flag with the DBT\_ARTIFACT\_STATE\_PATH environment variable

**This answer is incorrect. The correct answer is 'B'**

Answer: B  
Explanation: To reference the source freshness results in a subsequent dbt command, you can use the --select flag followed by the source\_status selector (e.g., source\_status:fresher+). This allows you to apply selectors based on the freshness of the data sources.

**You are working on a dbt project and want to ensure that only the data sources with freshness greater than the previous state are used for the transformation. Which selector should you use in your dbt command?**

Choose only ONE best answer.

**A**

source\_status:fresher

**B**

source\_status:fresher+

**C**

source\_status:stale

**D**

 source\_status:stale+

**This answer is correct.**

Answer: B

Explanation: To select only the data sources with freshness greater than the previous state, you should use the 'source\_status:fresher+' selector in your dbt command. This selector will ensure that the transformation only uses data sources that are fresher than their previous state.

**In a dbt project, you need to build a model using data sources that are no older than their previous state. Which selector should you use to achieve this?**

Choose only ONE best answer.

**A**

source\_status:fresher

**B**

source\_status:fresher+

**C**

source\_status:equal

**D**

source\_status:stale

**This answer is correct.**

Answer: A

Explanation: To build a model using data sources that are no older than their previous state, you should use the 'source\_status:fresher' selector in your dbt command. This selector will ensure that the transformation only uses data sources that are either fresher or equal to their previous state.

**A data analyst in your team wants to analyze data sources that have become stale compared to their previous state. Which selector should you use in your dbt command to meet this requirement?**

Choose only ONE best answer.

**A**

source\_status:fresher

**B**

source\_status:fresher+

**C**

source\_status:stale

**D**

source\_status:stale+

**This answer is incorrect. The correct answer is 'C'**

Answer: C

Explanation: To analyze data sources that have become stale compared to their previous state, you should use the 'source\_status:stale' selector in your dbt command. This selector will enable the data analyst to focus on the data sources that have become less fresh compared to their previous state.

**You are working on a dbt project and need to exclude data sources that have become stale compared to their previous state. Which selector should you use in your dbt command?**

Choose only ONE best answer.

**A**

source\_status:exclude\_stale

**B**

source\_status:fresher

**C**

 source\_status:stale

**D**

source\_status:fresher+

**This answer is correct.**

Answer: B

Explanation: To exclude data sources that have become stale compared to their previous state, you should use the 'source\_status:fresher' selector in your dbt command. This selector will ensure that the transformation only uses data sources that are either fresher or equal to their previous state, effectively excluding the stale ones.

**In a dbt project, you want to select data sources that are as fresh as their previous state, but not fresher. Which selector should you use in your dbt command?**

Choose only ONE best answer.

**A**

source\_status:equal

**B**

source\_status:fresher

**C**

source\_status:stale

**D**

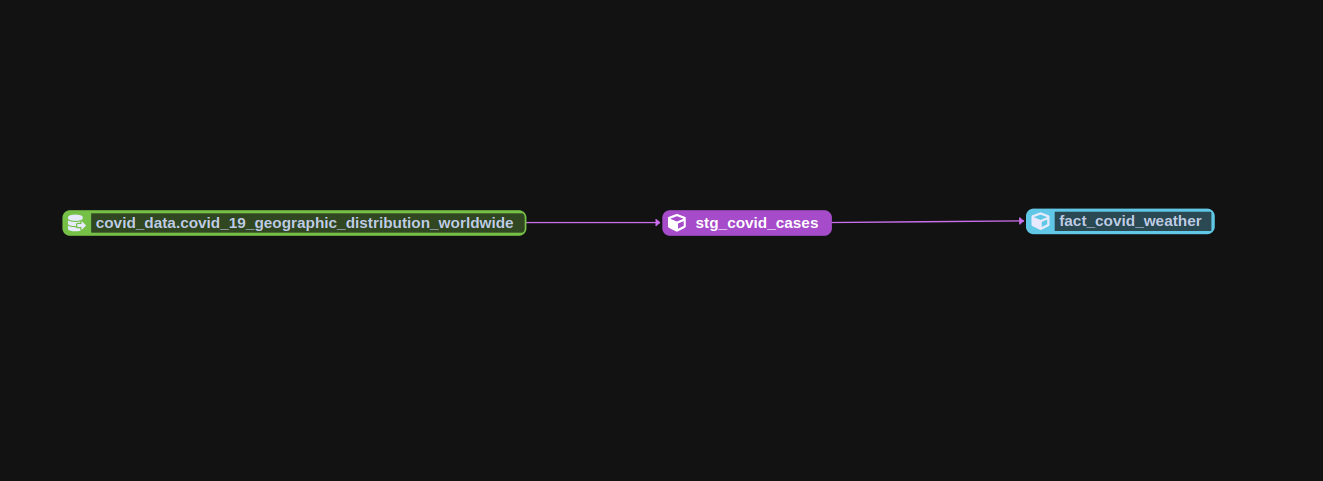
source\_status:fresher+

**This answer is incorrect. The correct answer is 'A'**

Answer: A

Explanation: To select data sources that are at least as fresh as their previous state but not fresher, you should use the 'source\_status:equal' selector in your dbt command. This selector will ensure that the transformation only uses data sources that have the same freshness as their previous state.

**You have the DAG below showing the 3 models, how do you ensure you run just the stg\_covid\_cases model?**

****

Choose only ONE best answer.

**A**

dbt run --s  stg\_covid\_cases

**B**

dbt run --select  model=+stg\_covid\_cases+

**C**

dbt run -s  stg\_covid\_cases

**D**

dbt run -select  stg\_covid\_cases

**This answer is incorrect. The correct answer is 'C'**

Explanation:

The right answer is C as we only need to run just that one model, so we do not need to add any of the graph operators.

**You initiated a dbt run as seen in the diagram below, How do you fix the cycle error encountered?**

****

Choose only ONE best answer.

**A**

Run the dbt self-heal command.

**B**

Rerun the model using dbt run --full-refresh

**C**

Update the ref functions to remove the cycle.

**D**

Rerun the model using dbt run --vars '{"cyclic": "false"}'

**This answer is correct.**

Explanation:

We observe cyclic errors in dbt when there is a circular dependency between two or more models, causing an infinite loop during the build process. This happens when a model depends on another model, which in turn depends on the first model, creating a cycle in the dependency graph. It can also occur with indirect dependencies involving more than two models. To fix this we need to update the ref function which builds dependencies between model and break the cycle.  
  
Reference: <https://docs.getdbt.com/guides/legacy/debugging-errors>