**dbt seed for snowflake**

In dbt, **seeds** are static CSV files that you can load directly into your data warehouse as tables. Seeds are often used for small, static datasets, reference tables, or test data that doesn’t change frequently. The dbt seed command reads these CSV files from the data folder in your dbt project and creates tables in your data warehouse with their contents.

**Why Use Seeds in dbt?**

* **Static Data**: For loading non-changing data such as country codes, region mappings, or other reference data.
* **Testing Data**: For creating small sets of data used in testing models.
* **Easy Loading**: dbt seeds let you manage small datasets easily without needing to write complex transformations.

**How dbt Seeds Work**

1. **CSV Files**: Place your CSV files inside the data folder in your dbt project.
2. **Load Command**: Run dbt seed to load CSV data from the data folder into tables in your data warehouse.
3. **Manage in Version Control**: Since seeds are CSV files, they can be tracked in version control (like Git), making them easy to update and maintain.

**Using Seeds in a dbt Project**

1. **Add CSV Files to the data Folder**

Inside your dbt project, create a folder named data (if it doesn’t exist) and place your CSV files there.

Example file structure:

my\_dbt\_project/

├── models/

├── data/

│ └── countries.csv

├── dbt\_project.yml

1. **Define the Seed Data in dbt**

Once the CSV file is in the data folder, you can load it using the dbt seed command.

1. **Run the Seed Command**

To load all seeds into your database, execute:

dbt seed

This command will create or replace tables in your data warehouse based on the CSV files.

1. **Configure Individual Seeds**

You can configure specific seeds to have different properties. For example, to set a seed to a view or specify a schema:

seeds:

my\_dbt\_project:

countries:

+schema: reference\_data

+materialized: table

1. **Query Seed Tables in Models**

Once loaded, you can use seed tables in models with ref:

SELECT \*

FROM {{ ref('countries') }}

**Example: Loading a Seed File and Using It in a Model**

1. **Create a Seed File**: countries.csv

country\_code,country\_name

US,United States

CA,Canada

MX,Mexico

1. **Run dbt seed** to load this data as a table in your data warehouse.
2. **Use the Seed in a Model**:

-- models/customer\_with\_country.sql

SELECT

customer\_id,

customers.country\_code,

countries.country\_name

FROM {{ ref('customers') }}

**Benefits of dbt Seeds**

* **Simplifies Small, Static Data Management**: No need for complex ETL.
* **Improves Data Consistency**: Ensure reference data is standardized and controlled.
* **Enables Testing**: Load testing datasets that can be reused in different scenarios.

**Limitations of dbt Seeds**

* **Not Meant for Large Datasets**: Seeds work best for small, static data.
* **Lack of Update Support**: To change seed data, you must modify the CSV file and re-run dbt seed.

**Summary**

dbt seeds are a simple and powerful way to manage small, static datasets in your dbt project. They streamline reference data handling, facilitate testing, and make it easy to track data changes in version control. By following the steps above, you can quickly set up and use dbt seeds for reliable, consistent data within your models.