**# Build model**

* NOTE: query format different in snowflake and DBT
* Models Models are **.sql files** that live in the models folder.
* Models are simply **written as select statements** - there is **no DDL/DML** that needs to be written around this. This allows the developer to focus on the logic.
* In the Cloud IDE, the **Preview** button will run this select statement against your data warehouse. The results shown here are equivalent to what this model will return once it is materialized.
* After constructing a model, **dbt run** in the command line will actually **materialize the models into the data warehouse**. The default materialization is a view.

1. Perform same query run from snowflake to DBT cloud

* Using `Preview Selection` will only run the sql query to connected SNOWFLAKE, it will not build or make changes on snowflake

2. To build a object on snowflake warehouse

* We need to execute the command at the bottom. This will define the available models and executes the model

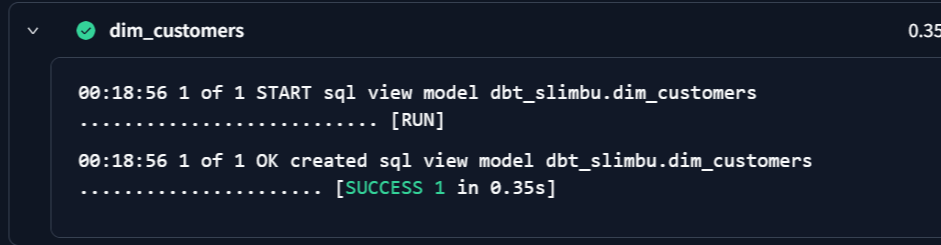
-`dbt run`

3. Compile:

* Displays the actual code. The actual table being pulled from. It shows the source of the records

4. Execute ‘**dbt run’:**

* Constructs the objects from the model inside the warehouse. Takes care of the ‘DDL’ queries
* Log provides more detailed information. EG: may create underlying ‘VIEW’ when building the model in the warehouse
* Creates a new view by ‘default’ on the target warehouse and schema from the model



5. Creates table and not view by **‘default’**:

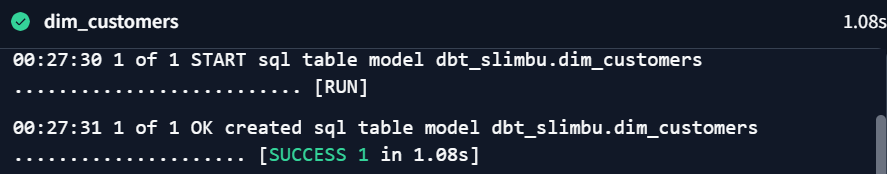
* Create configuration settings on same sql query at the top using ‘JINJA’ code
* Specify ‘TABLE’ on config to create a materialized as ‘table’
* EG:
  + {{

config(

materialized= ‘table’

)

}}



6. Run only targeted model:

* By specifying the model name, it will not run other available models and only runs the specified model
* The following command will only materialize the ‘dim\_customers.sql’ and its downstream models as well
  + ‘dbt run –select dim\_customers+’