# Deployment of model

* Create new environment
* Select the branch you want to use for deployment: MAIN or CUSTOM branch
* Deployment Environment and Credentials
  + Recommend to use ‘Service’ account which has username credentials and Schema

### **# Development vs. Deployment**

* Development in dbt is the process of building, refactoring, and organizing different files in your dbt project. This is done in a development environment using a development schema (dbt\_jsmith) and typically on a *non-default* branch (i.e. feature/customers-model, fix/date-spine-issue). After making the appropriate changes, the development branch is merged to main/master so that those changes can be used in deployment.
* Deployment in dbt (or running dbt in production) is the process of running dbt on a schedule in a deployment environment. The deployment environment will typically run from the *default* branch (i.e., main, master) and use a dedicated deployment schema (e.g., dbt\_prod). The models built in deployment are then used to power dashboards, reporting, and other key business decision-making processes.
* The use of development environments and branches makes it possible to continue to build your dbt project *without* affecting the models, tests, and documentation that are running in production.

## Contents:

* DBT Job deployment log features
* DBT Explore

# Deploy Job

* Description of the job name
* Environment: Production
* Execution settings:
  + //command selection
    - Dbt build (runs and tests all the models in DAG order)
  + Generate docs on run
  + Run source freshness; enables dbt source freshness as the first step of job, without breaking steps
* Set time-zone
  + To run at specific time of the day
  + Verify time-zone settings

# Review the scheduled jobs

* Deploy → Production → Daily Jobs -> Run <number>
* Provides what DBT steps were triggered, invoked and action items taken (docs generated), etc.
* Logs provide information regarding failures of different steps
* Lineage feature shows time taken to complete the steps
* Data Sources: provides the source of data
* Documentation of the job

# DBT Explore (Left section)

* Project Lineage:
  + Provides all the objects in DAG orders. Information includes the source, models, statuses, passed, etc.
* Recommendations:
  + Provides project summary with documentation of the models
  + Based on ‘severity’ of cases to mitigate the errors
* Resources, File Tree, and Databases:
  + Provides structure of different objects
* To check the source of data; if pulling from source or from the DBT models
  + Model level analysis:
    - Information on column level, renaming
  + Sources Level
    - Description
    - Source tables
    - Lineage of data
    - Column data types
  + Tests:
    - Different tests on different models
    - Provides the test that has been passed or failed