

12. Documentation

Documenting Your Models

DBT provides a comprehensive documentation system that allows you to:

- Document models, columns, and tests
- Generate a documentation website
- Share knowledge across your team

Model Documentation

Add descriptions to your models in YAML files:

yaml

```
version: 2

models:
  - name: stg_customers
    description: Staged customer data from the CRM system
    columns:
      - name: customer_id
        description: Primary key for the customer
      - name: email
        description: Customer email address, used for communications
      - name: first_name
        description: Customer first name
      - name: last_name
        description: Customer last name
```

Advanced Documentation

You can use Markdown in your descriptions for rich formatting:

yaml

```

models:
  - name: customer_orders
    description: >
      # Customer Orders Summary

      This model provides an aggregated view of customer orders.

      ## Business Rules

      - Includes only completed orders
      - Currency is in USD
      - Grouped by customer and month

      ## Usage Examples

      ```sql
 SELECT * FROM customer_orders WHERE order_month = '2023-01-01'
      ```

columns:
  - name: customer_id
    description: Foreign key to the customers table

```

Generating Documentation

To generate and view the documentation:

bash

```

# Generate documentation
dbt docs generate

# Serve documentation locally
dbt docs serve

```

This will:

1. Compile all documentation from your project
2. Generate a static website
3. Serve it on a local web server (typically at <http://localhost:8080>)

The documentation site includes:

- Model descriptions

- Column descriptions
- Data lineage graphs
- Test definitions
- Macro documentation
- Project structure

Customizing Documentation

You can customize the documentation with:

1. Project-Level Documentation

Create an `overview.md` file in your project's root:

markdown

MyCompany Data Warehouse

This DBT project transforms raw data into analytics-ready models for the marketing team.

Getting Started

To use these models, connect to the `'analytics'` schema in Snowflake.

Model Categories

- **Staging**: Raw data with minimal transformations
- **Intermediate**: Joined and prepared data
- **Marts**: Business-specific data models

2. Custom Macros for Documentation

You can create macros that generate documentation:

sql

```
{% macro doc_table_description(model_name) %}  
  {% set model_description %}  
    This table contains information about {{ model_name }}.
```

```
Last updated: {{ modules.datetime.datetime.now().strftime('%Y-%m-%d') }}
}}

{% endset %}

{{ return(model_description) }}
{% endmacro %}
```

3. Documentation Override

You can override documentation settings in your `dbt_project.yml` :

yaml

```
docs:
  project_name: "Marketing Analytics"
  homepage: "overview.md"
```

