4. DBT Models

What are DBT Models?

DBT models are SQL select statements that transform your data. Each model:

- Is defined in a `.sql` file
- Contains a single `SELECT` statement
- Produces a table or view in your data warehouse
- Can reference other models, creating a dependency graph

Models are the core building blocks of DBT projects and represent the transformations applied to your data.

Creating Your First Model

Let's create a simple model that transforms customer data:

1. Create a new file `models/staging/stg_customers.sql`:

```
WITH source AS (
    SELECT * FROM {{ source('raw', 'customers') }}
),

cleaned AS (
    SELECT
        customer_id,
        first_name,
        last_name,
        TRIM(email) AS email,
        COALESCE(phone, 'unknown') AS phone,
        created_at,
        updated_at
    FROM source
)

SELECT * FROM cleaned
```

2. Define the source in `models/sources.yml`:

```yaml
version: 2

sources:
  - name: raw
    database: raw_data
    tables:
      - name: customers
        description: Raw customer data from the CRM system
        columns:
          - name: customer_id
            description: Primary key of the customers table
            tests:
              - unique
              - not_null
```

3. Build the model:

bash

```
dbt run --models stg_customers
```

Model Dependencies

Models can reference other models, creating a dependency graph. DBT automatically resolves these dependencies and builds models in the correct order.

Example of model dependencies:

1. `models/staging/stg_orders.sql`:

```sql
SELECT
    order_id,
    customer_id,
    order_date,
    status,
    amount
FROM {{ source('raw', 'orders') }}
```

2. `models/marts/customer_orders.sql`:

```sql
WITH customers AS (
    SELECT * FROM {{ ref('stg_customers') }}
```

```
),

orders AS (
    SELECT * FROM {{ ref('stg_orders') }}
),

customer_orders AS (
    SELECT
        customers.customer_id,
        customers.first_name,
        customers.last_name,
        customers.email,
        COUNT(orders.order_id) AS number_of_orders,
        SUM(orders.amount) AS total_order_amount
    FROM customers
    LEFT JOIN orders ON customers.customer_id = orders.customer_id
    GROUP BY 1, 2, 3, 4
)

SELECT * FROM customer_orders
```

In this example, `customer_orders` depends on both `stg_customers` and `stg_orders`. DBT will ensure that the staging models are built before attempting to build the `customer_orders` model.

Ref - https://docs.getdbt.com/docs/build/models