11. Packages

What are DBT Packages?

DBT packages are collections of models, macros, and tests that can be reused across projects. They provide a way to:

- Share common transformations
- Leverage community-created functionality
- Maintain modular code across projects

Using External Packages

1. Configuring Packages

To use external packages, create a `packages.yml` file in your project root:

yaml

```yaml
packages:
  - package: dbt-labs/dbt_utils
    version: 0.8.0
  - package: calogica/dbt_expectations
    version: 0.5.0
  - git: https://github.com/fishtown-analytics/dbt-audit-helper.git
    revision: v0.4.0
```

2. Installing Packages

Install the packages using:

bash

```bash
dbt deps
```

This command will:

- Download the specified packages
- Install them in the `dbt_packages` directory
- Make their contents available to your project

3. Using Package Functionality

Once installed, you can use functionality from the packages:

sql

```sql
-- Using a macro from dbt_utils
SELECT
    customer_id,
    {{ dbt_utils.generate_surrogate_key(['customer_id', 'order_date']) }}
as surrogate_key
FROM {{ ref('stg_orders') }}
```

Popular DBT Packages

1. **dbt-utils**: Provides utility macros for common needs
   - String manipulation
   - Date operations
   - Schema tests
   - SQL helpers
2. **dbt-expectations**: Implements Great Expectations-style tests
   - Statistical validations
   - Data quality checks
   - Pattern matching
3. **dbt-audit-helper**: Helps with data auditing
   - Compares query results
   - Validates transformations
4. **dbt-date**: Advanced date and time operations
   - Fiscal periods
   - Calendar tables
   - Date manipulation

Creating Your Own Package

You can create your own package for reuse across projects:

1. **Create a Package Project**:
   - Set up a normal DBT project
   - Add models, macros, and tests you want to share
   - Create a `README.md` explaining usage

2. **Set Up Package Configuration**:
   - Add a `dbt_project.yml` file with package settings
   - Configure exports to specify what should be available

3. **Publish Your Package**:
   - Host it on GitHub or another Git repository
   - Reference it in other projects: yaml

```yaml
packages:
  - git: https://github.com/your-username/your-dbt-package.git
    revision: v1.0.0
```