

Case Study- DE (Python)

Pre- Requestics:- Log-in to Snowflake Trial Account

Here are the step-by-step instructions to create a **Snowflake Trial Account**:

✓ **Step 1: Go to Snowflake Trial Page**

Visit: <https://signup.snowflake.com/>

✓ **Step 2: Fill in Basic Information**

You'll see a form with the following fields:

- **First Name**
- **Last Name**
- **Business Email** (Use a valid email; required for activation)
- **Company Name**
- **Country**
- **Phone Number** (*optional*)

Click **Continue** or **Get Started**.

✓ **Step 3: Choose a Cloud Provider & Region**

Select:

- **Cloud Provider**: AWS, Azure, or Google Cloud (Choose AWS)
- **Region**: Pick the closest region to your location for best performance

⚠ Choose this carefully — it cannot be changed later.

✅ Step 4: Account Creation & Confirmation

- You'll receive an **email from Snowflake** to verify your email.
- Click the link in the email to **activate your trial**.

✅ Step 5: Set Up Your Account

You'll be prompted to:

- **Create a username**
- **Set a password**
- **Agree to terms**

You will then be redirected to the **Snowflake Web UI (Snowsight)**.

CASE STUDY:-

Create a Configuration Page of Extraction Framework and in the Configuration Page you can do following task:

The **Configuration Page** allows users to perform the following tasks:

1. **Select Source**

- A dropdown menu to select the data source, with options including **Tables** and **Views**.

2. **Select or Create Configuration**

- A dropdown to choose an **existing configuration**, filtered based on the selected source.

- Option to **create a new configuration** if needed.

3. Global Extract Filters

- A button labeled **"Add Filter"** opens an expander.
- Inside the expander:
 - **Filter Type**: Options include **WHERE**, **IN**, and **BETWEEN**.
 - **Column**: Dropdown displaying all column names from the selected source.
 - **Condition**: Dropdown with operators **=**, **<**, **>**, **<=**, **>=**, **!=**, and **LIKE**.
 - **Case Sensitivity**: A select box for **case-sensitive matching** (only for string values).
 - **Combine with next filter**: Option to combine using **AND** / **OR** logic.

4. Attribute (Rules Engine)

- Supports **CASE-WHEN** conditions to define custom rules.
- A **"Load Existing Rule"** dropdown will auto-populate rules associated with the selected configuration.

5. Preview SQL Results

- Clicking the **"Preview SQL Results"** button will show a data preview based on the applied filters and rules.

6. Save Configuration

- Clicking the **"Save Configuration"** button saves the setup as a **view** in a database table.

- If a new rule is added to an existing configuration, it will be saved as a **new version**.

7. Load Existing Configuration

- When an existing configuration is selected:
 - The latest version is loaded.
 - All associated filters and rules are auto-populated.

Below are some screenshot what it should look like:

The screenshot shows a dark-themed form titled "Extract Configuration". In the top right corner, there is a blue button with a circular arrow icon and the text "Refresh". The form contains the following elements: a label "Select Source Name" above a dropdown menu showing "TEST_SOURCE"; a button labeled "Run and Preview Source Data"; a label "Select Existing Configuration:" above a dropdown menu showing "test_3"; and a label "Enter New Configuration Name:" above an empty text input field.

The screenshot shows a dark-themed form titled "Global Extract Filters". It features an "Add Filter" button at the top left. Below it is a "Filter Type" dropdown menu set to "Where". The main section contains three columns: "Column" with a dropdown showing "ADDRESS_ID", "Condition" with a dropdown showing "=", and "Value" with an empty text input field. Below these columns is a checkbox labeled "Case sensitive". At the bottom, there is a label "Combine with next filter using" above a dropdown menu set to "AND". Finally, there are "Cancel" and "Apply Filters" buttons at the bottom of the form.

Attributes (Rules Engine)

Add Simple Rule

Add Complex Rule

Load Existing Rule

None

Rule Name*

Description*

Column

Operator

Value

ADDRESS_ID

=

THEN

ELSE

New Column

Add Rule

Cancel

Preview SQL Results

SQL preview generated successfully!

Generated SQL

```
WITH base AS (
  SELECT * FROM TEST_SOURCE_SOURCE_VIEW
)

SELECT "ADDRESS_ID", "ADDRESS_LINE1", "ADDRESS_LINE2", "CITY", "ADDRESS_TYPE_ID", "NAME", "ROWGUID", C
  WHEN "ADDRESS_TYPE_ID" = '2' THEN 'go'
  ELSE 'stop'
END AS "new"
FROM base
```

Latest SQL Preview

	ADDRESS_ID	ADDRESS_LINE1	ADDRESS_LINE2	CITY	ADDRESS_TYPE_ID	NAME	ROWGUID
0	1	1970 Napa Ct.	None	Bothell	1	Billing	B84F78B1-4EFE-4A0E-8CB7-70E
1	2	9833 Mt. Dias Blv.	None	Bothell	2	Home	41BC2FF6-F0FC-475F-8EB9-CEC
2	3	7484 Roundtree Drive	None	Bothell	3	Main Office	8EEEC28C-07A2-4FB9-AD0A-42E
3	4	9539 Glenside Dr	None	Bothell	4	Primary	24CB3088-4345-47C4-86C5-17B
4	5	1226 Shoe St.	None	Bothell	5	Shipping	B29DA3F8-19A3-47DA-9DAA-15C
5	6	1399 Firestone Drive	None	Bothell	6	Archive	A67F238A-5BA2-444B-966C-046

Save Configuration

View TEST_3_V2 created successfully.

Configuration version 2 saved with view: TEST_3_V2

Configuration saved successfully!