

30 Days of AI

The #30DaysOfAI is a coding challenge designed to help you get started in building AI apps with Streamlit.

Start the Challenge 

Day 1: Connect to Snowflake

Contents

1. Introduction:	2
2. Prerequisites:	2
3. How It Works: Step-by-Step	2
3.1. Import Libraries	3
3.2. Connect to Snowflake	4
3.3. Query Snowflake Version	4
3.4. Display Results	4
4. Resources	4

1. Introduction:

For Day 1, our goal is to establish a connection between our Streamlit app and a Snowflake database. Once that's done, we'll run a simple query to confirm the connection is working and display the Snowflake version in the app.

So to get started, sign up for a [free Snowflake trial account](#) using this link and you'll have access for 120 days. The trial includes plenty of credits to get you through the entire 30-day challenge, plus extra for experimenting with other features.

Note: The core learning of the 30 Days of AI challenge is tech agnostic and can therefore be adapted to work with other platforms and LLM API providers.

See the code:

```
import streamlit as st

st.title(":material/vpn_key: Day 1: Connect to Snowflake")

# Connect to Snowflake
try:
    # Works in Streamlit in Snowflake
    from snowflake.snowpark.context import get_active_session
    session = get_active_session()
except:
    # Works locally and on Streamlit Community Cloud
    from snowflake.snowpark import Session
    session = Session.builder.configs(st.secrets["connections"]["snowflake"]).

# Query Snowflake version
version = session.sql("SELECT CURRENT_VERSION()").collect()[0][0]

# Display results
st.success(f"Successfully connected! Snowflake Version: {version}")
```

2. Prerequisites:

- 1) Snowflake trial account
- 2) Python 3.10+
- 3) Streamlit installed
- 4) Basic SQL knowledge

3. How It Works: Step-by-Step

Connection Setup

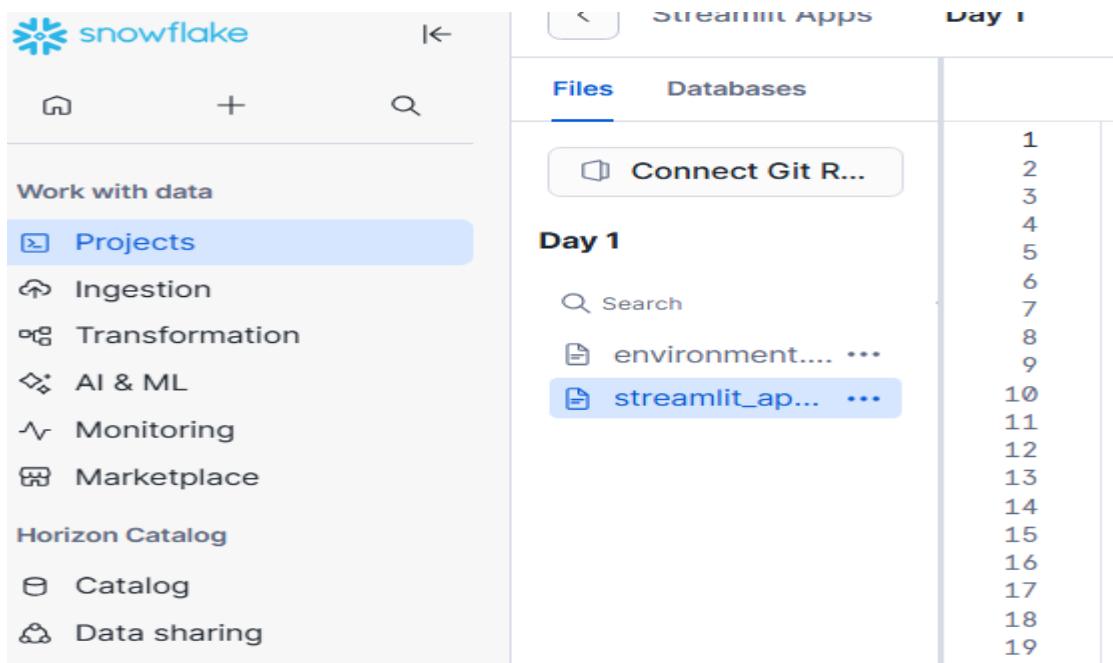
Before running Day 1, you need to configure your Snowflake connection based on where you're deploying:

Streamlit in Snowflake (Recommended)

No setup needed! Just create a Streamlit app in Snowsight and the connection works automatically. Skip to Step 1 below.

After login to snowflake, select Project and then click on streamlit to create streamlit app in snowflake.

One-time setup required. Create `.streamlit/secrets.toml` in your project folder:



Local Development or Streamlit Community Cloud

```
[connections.snowflake]
account = "xy12345.us-east-1"          #
user = "yourusername"                   # Find in Snowsight → Account → View accou
password = "yourpassword"              # Your Snowflake username
role = "ACCOUNTADMIN"                 # Your role
warehouse = "COMPUTE_WH"               # Your warehouse
database = "SNOWFLAKE_LEARNING_DB"     # Your database
schema = "PUBLIC"                     # Your schema
```

3.1. Import Libraries

```
import streamlit as st
```

- `import streamlit as st`: Imports the Streamlit library, which is used to build the web app's user interface (UI).

3.2. Connect to Snowflake

```
# Auto-detect environment and connect
try:
    from snowflake.snowpark.context import get_active_session
    session = get_active_session()
except:
    from snowflake.snowpark import Session
    session = Session.builder.configs(st.secrets["connections"]["snowflake"]).
```

Why the try/except? This pattern makes your code work in all three environments: Streamlit in Snowflake (production), local development, and Streamlit Community Cloud. One codebase works everywhere!

3.3. Query Snowflake Version

```
# Query and display Snowflake version
```

```
# Query and display Snowflake version
version = session.sql("SELECT CURRENT_VERSION()").collect()[0][0]
```

3.4. Display Results

```
st.success(f"Successfully connected! Snowflake Version: {version}")
```

Output result.

🔑 Day 1: Connect to Snowflake

Successfully connected! Snowflake Version: 10.0.0

4. Resources

- [Streamlit in Snowflake Documentation](#)
- [Streamlit Secrets Management](#)
- [Snowpark Python API](#)
- [Get started with Streamlit](#)