Name: Khac Minh Dai Vo.

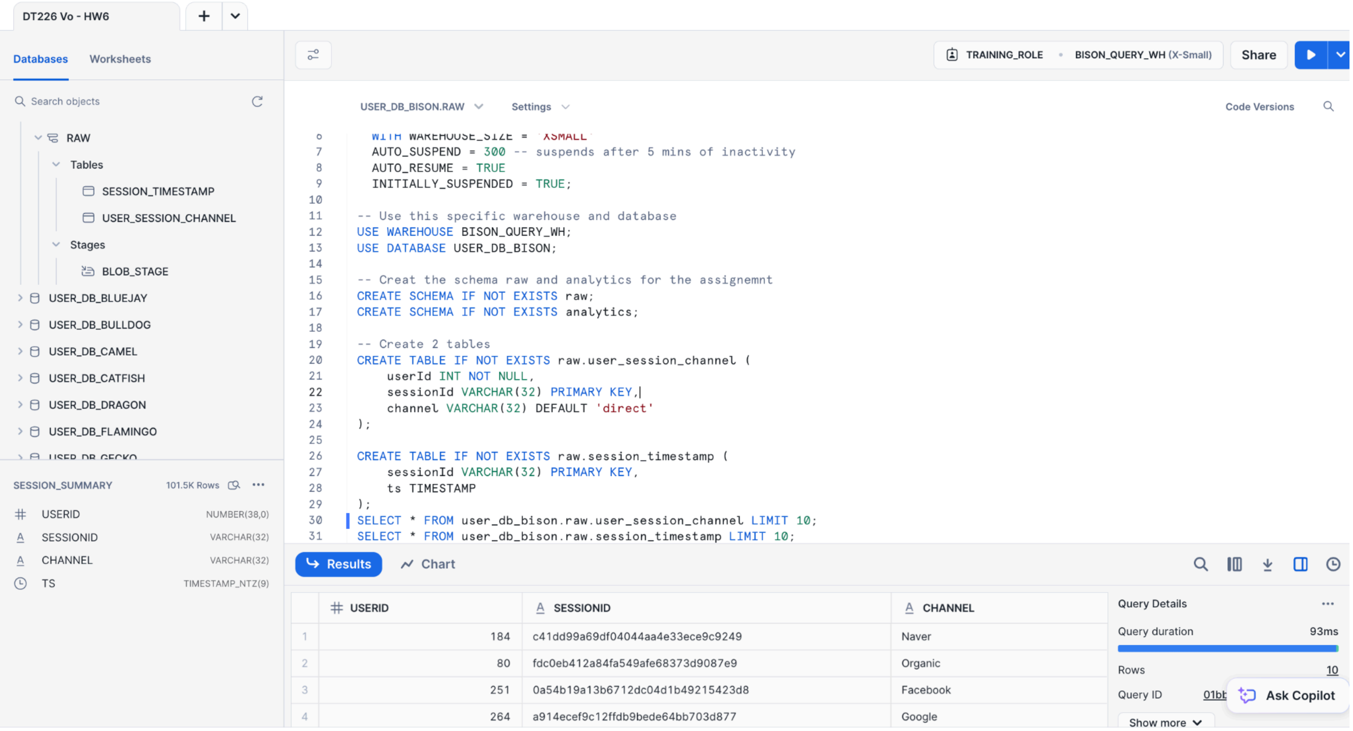
Student ID: 014705119.

Course: DATA226

Assignment #6

Create a WAU (Weekly Active User) chart from any BI tool. Total 10 points

1. **Import two tables in your Snowflake Links to an external site as an ETL DAG in your Airflow (+3pt)**
    a. user_session_channel and session_timestamp (under raw schema or equivalent)



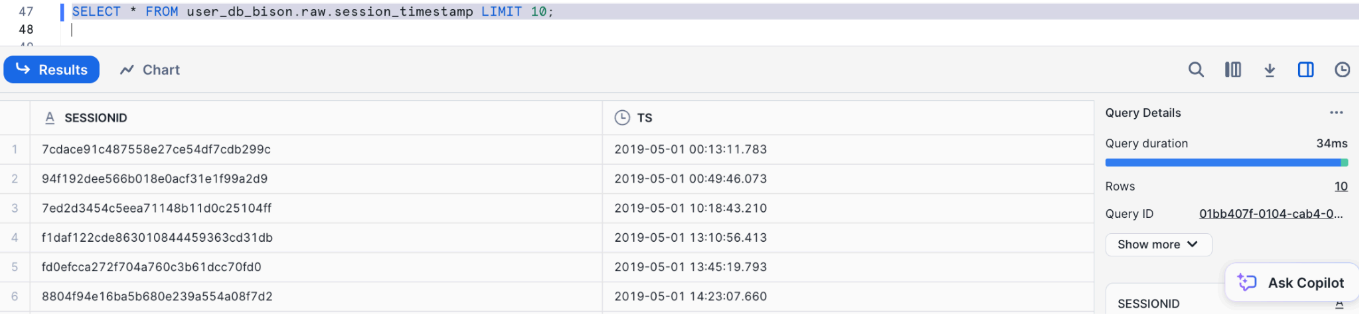
SESSION_SUMMARY 101.5K Rows

USERID	NUMBER(38,0)
SESSIONID	VARCHAR(32)
CHANNEL	VARCHAR(32)
TS	TIMESTAMP_NTZ(9)

```
  WITH WAREHOUSE_SIZE = 'XSMALL'
  AUTO_SUSPEND = 300 -- suspends after 5 mins of inactivity
  AUTO_RESUME = TRUE
  INITIALLY_SUSPENDED = TRUE;

-- Use this specific warehouse and database
USE WAREHOUSE BISON_QUERY_WH;
USE DATABASE USER_DB_BISON;

-- Creat the schema raw and analytics for the assignemnt
CREATE SCHEMA IF NOT EXISTS raw;
CREATE SCHEMA IF NOT EXISTS analytics;

-- Create 2 tables
CREATE TABLE IF NOT EXISTS raw.user_session_channel (
    userId INT NOT NULL,
    sessionId VARCHAR(32) PRIMARY KEY,
    channel VARCHAR(32) DEFAULT 'direct'
);

CREATE TABLE IF NOT EXISTS raw.session_timestamp (
    sessionId VARCHAR(32) PRIMARY KEY,
    ts TIMESTAMP
);
SELECT * FROM user_db_bison.raw.user_session_channel LIMIT 10;
SELECT * FROM user_db_bison.raw.session_timestamp LIMIT 10;
```

	USERID	SESSIONID	CHANNEL
1	184	c41dd99a69df04044aa4e33ece9c9249	Naver
2	80	fdc0eb412a84fa549afe68373d9087e9	Organic
3	251	0a54b19a13b6712dc04d1b49215423d8	Facebook
4	264	a914ecef9c12ffdb9bede64bb703d877	Google

Query Details
Query duration 93ms
Rows 10
Query ID 01bb



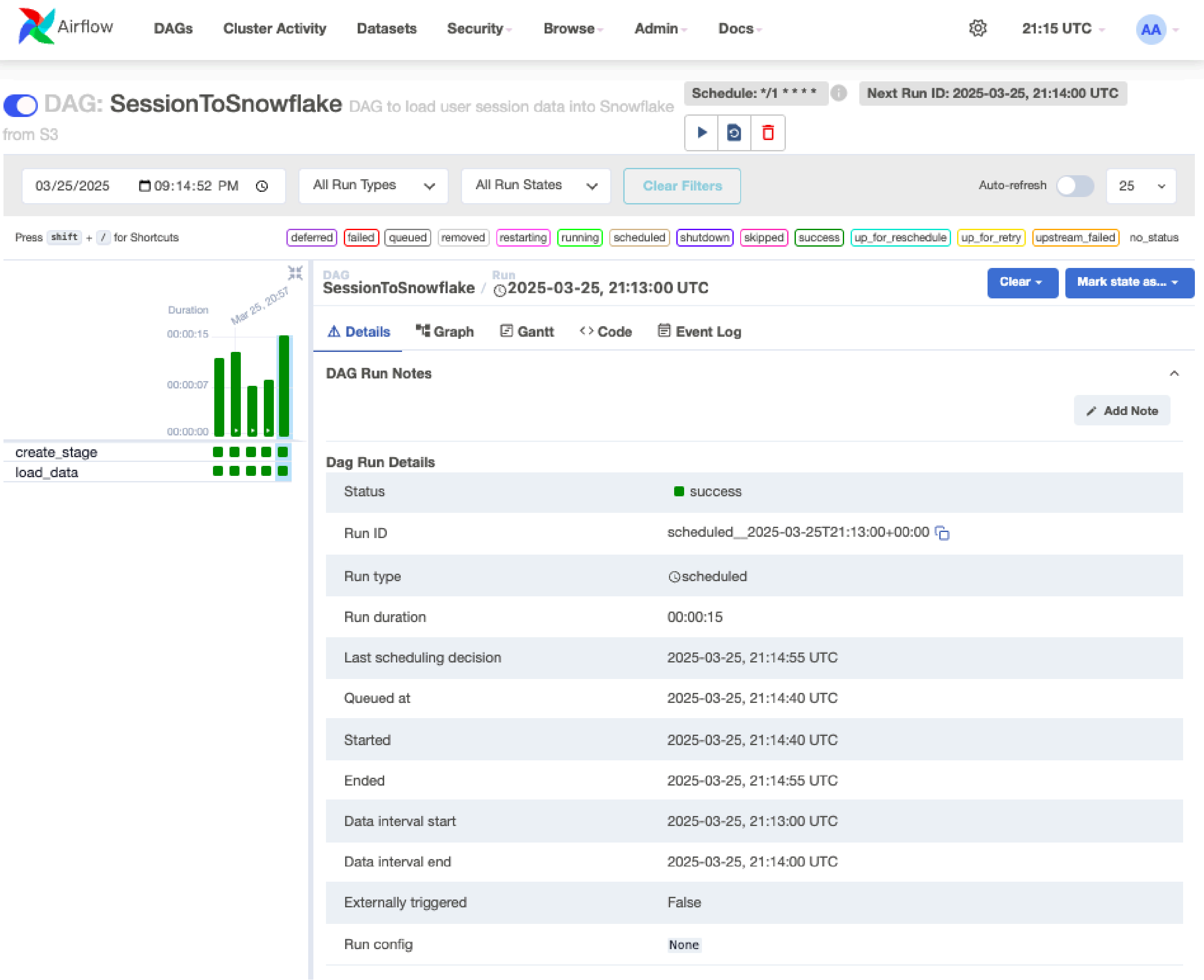
```
SELECT * FROM user_db_bison.raw.session_timestamp LIMIT 10;
```

	SESSIONID	TS
1	7cdace91c487558e27ce54df7cdb299c	2019-05-01 00:13:11.783
2	94f192dee566b018e0acf31e1f99a2d9	2019-05-01 00:49:46.073
3	7ed2d3454c5eea71148b11d0c25104ff	2019-05-01 10:18:43.210
4	f1daf122cde863010844459363cd31db	2019-05-01 13:10:56.413
5	fd0efcca272f704a760c3b61dcc70fd0	2019-05-01 13:45:19.793
6	8804f94e16ba5b680e239a554a08f7d2	2019-05-01 14:23:07.660

Query Details
Query duration 34ms
Rows 10
Query ID 01bb407f-0104-cab4-0...
SESSIONID

Note: the 2 tables were created and tested successfully with outputs.

b. Capture the screenshot of this DAG's detailed page from the Web UI (#1)



DAG: SessionToSnowflake DAG to load user session data into Snowflake from S3

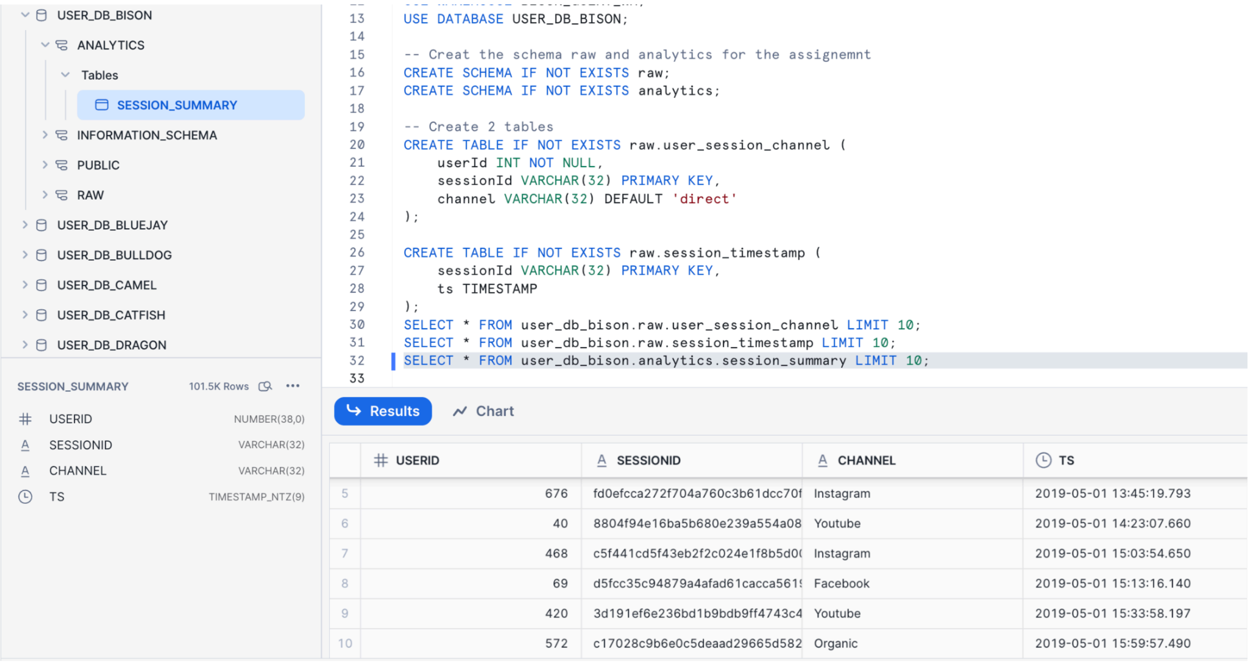
DAG SessionToSnowflake / Run 2025-03-25, 21:13:00 UTC

DAG Run Notes

Dag Run Details

Status	🟩 success
Run ID	scheduled__2025-03-25T21:13:00+00:00
Run type	scheduled
Run duration	00:00:15
Last scheduling decision	2025-03-25, 21:14:55 UTC
Queued at	2025-03-25, 21:14:40 UTC
Started	2025-03-25, 21:14:40 UTC
Ended	2025-03-25, 21:14:55 UTC
Data interval start	2025-03-25, 21:13:00 UTC
Data interval end	2025-03-25, 21:14:00 UTC
Externally triggered	False
Run config	None

2. **Create a ELT DAG in your Airflow to create a JOINED table of the two as described during the class (+3pt)**
    a. session_summary (under analytics)



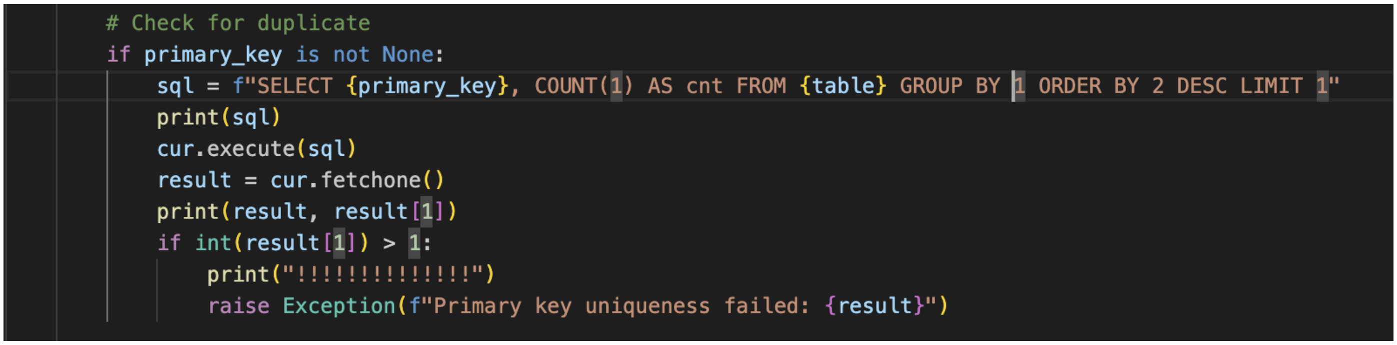
USERID	NUMBER(38,0)
SESSIONID	VARCHAR(32)
CHANNEL	VARCHAR(32)
TS	TIMESTAMP_NTZ(9)

```
USE DATABASE USER_DB_BISON;

-- Creat the schema raw and analytics for the assignemnt
CREATE SCHEMA IF NOT EXISTS raw;
CREATE SCHEMA IF NOT EXISTS analytics;

-- Create 2 tables
CREATE TABLE IF NOT EXISTS raw.user_session_channel (
    userId INT NOT NULL,
    sessionId VARCHAR(32) PRIMARY KEY,
    channel VARCHAR(32) DEFAULT 'direct'
);

CREATE TABLE IF NOT EXISTS raw.session_timestamp (
    sessionId VARCHAR(32) PRIMARY KEY,
    ts TIMESTAMP
);
SELECT * FROM user_db_bison.raw.user_session_channel LIMIT 10;
SELECT * FROM user_db_bison.raw.session_timestamp LIMIT 10;
SELECT * FROM user_db_bison.analytics.session_summary LIMIT 10;
```

	USERID	SESSIONID	CHANNEL	TS
5	676	fd0efcca272f704a760c3b61dcc70f	Instagram	2019-05-01 13:45:19.793
6	40	8804f94e16ba5b680e239a554a08	Youtube	2019-05-01 14:23:07.660
7	468	c5f441cd5f43eb2f2c024e1f8b5d0(	Instagram	2019-05-01 15:03:54.650
8	69	d5fcc35c94879a4afad61cacca561!	Facebook	2019-05-01 15:13:16.140
9	420	3d191ef6e236bd1b9bdb9ff4743c4	Youtube	2019-05-01 15:33:58.197
10	572	c17028c9b6e0c5deaad29665d582	Organic	2019-05-01 15:59:57.490

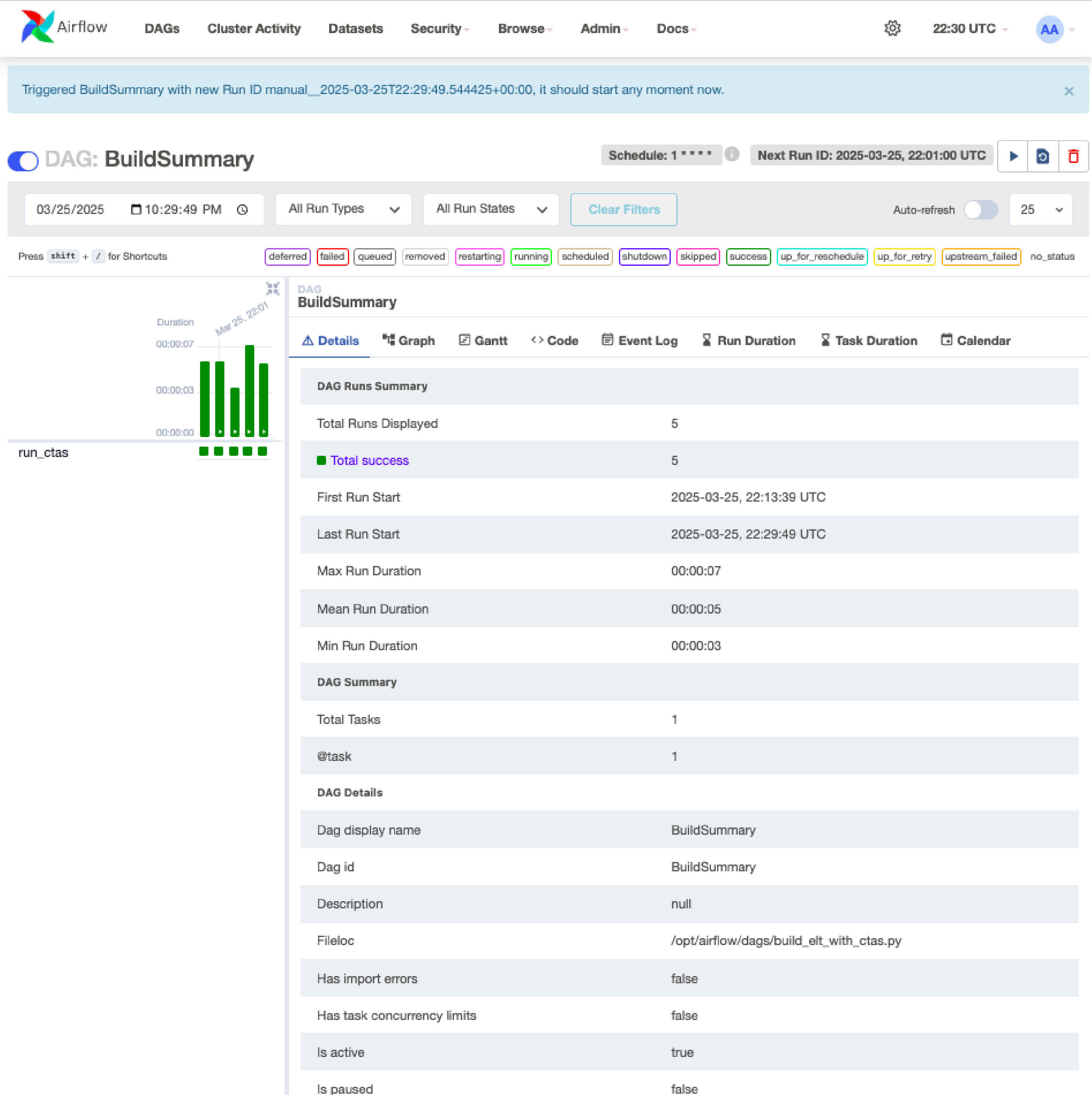
Note: Successfully created session_summary table under analytics.

b. **Extra point**: add one more condition to check duplicate records (+1pt)



```
    # Check for duplicate
    if primary_key is not None:
        sql = f"SELECT {primary_key}, COUNT(1) AS cnt FROM {table} GROUP BY 1 ORDER BY 2 DESC LIMIT 1"
        print(sql)
        cur.execute(sql)
        result = cur.fetchone()
        print(result, result[1])
        if int(result[1]) > 1:
            print("!!!!!!!!!!!!!!")
            raise Exception(f"Primary key uniqueness failed: {result}")
```

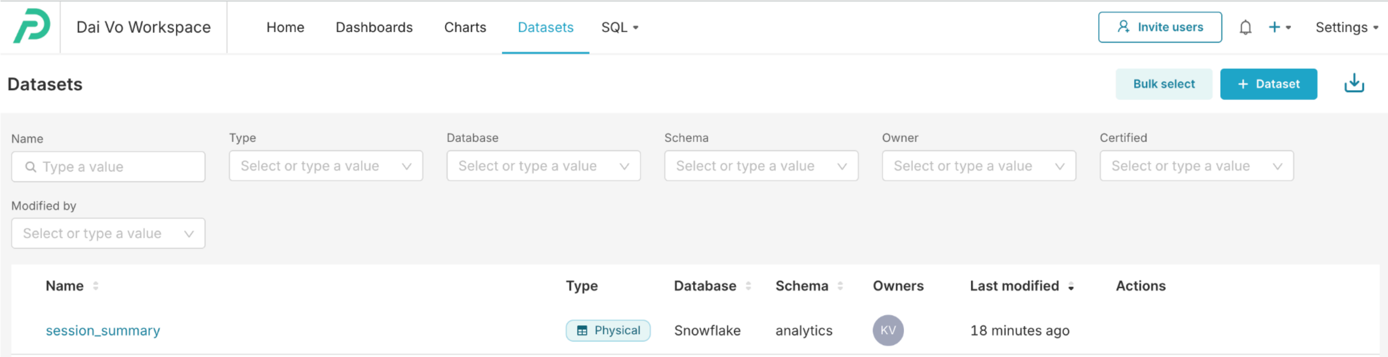
c. Capture the screenshot of this DAG’s detailed page from the Web UI (#2)



Triggered BuildSummary with new Run ID manual__2025-03-25T22:29:49.544425+00:00, it should start any moment now.

DAG Runs Summary	
Total Runs Displayed	5
🟩 Total success	5
First Run Start	2025-03-25, 22:13:39 UTC
Last Run Start	2025-03-25, 22:29:49 UTC
Max Run Duration	00:00:07
Mean Run Duration	00:00:05
Min Run Duration	00:00:03
DAG Summary	
Total Tasks	1
@task	1
DAG Details	
Dag display name	BuildSummary
Dag id	BuildSummary
Description	null
Fileloc	/opt/airflow/dags/build_elt_with_ctas.py
Has import errors	false
Has task concurrency limits	false
Is active	true
Is paused	false

3. **Set up your Preset account or [Docker Superset environment Links to an external site.](#) or Tableau account (+2pt)**
   a. This includes setting up Snowflake connection and import session_summary from step2
   b. Capture the screenshot of your Datasets (#3)



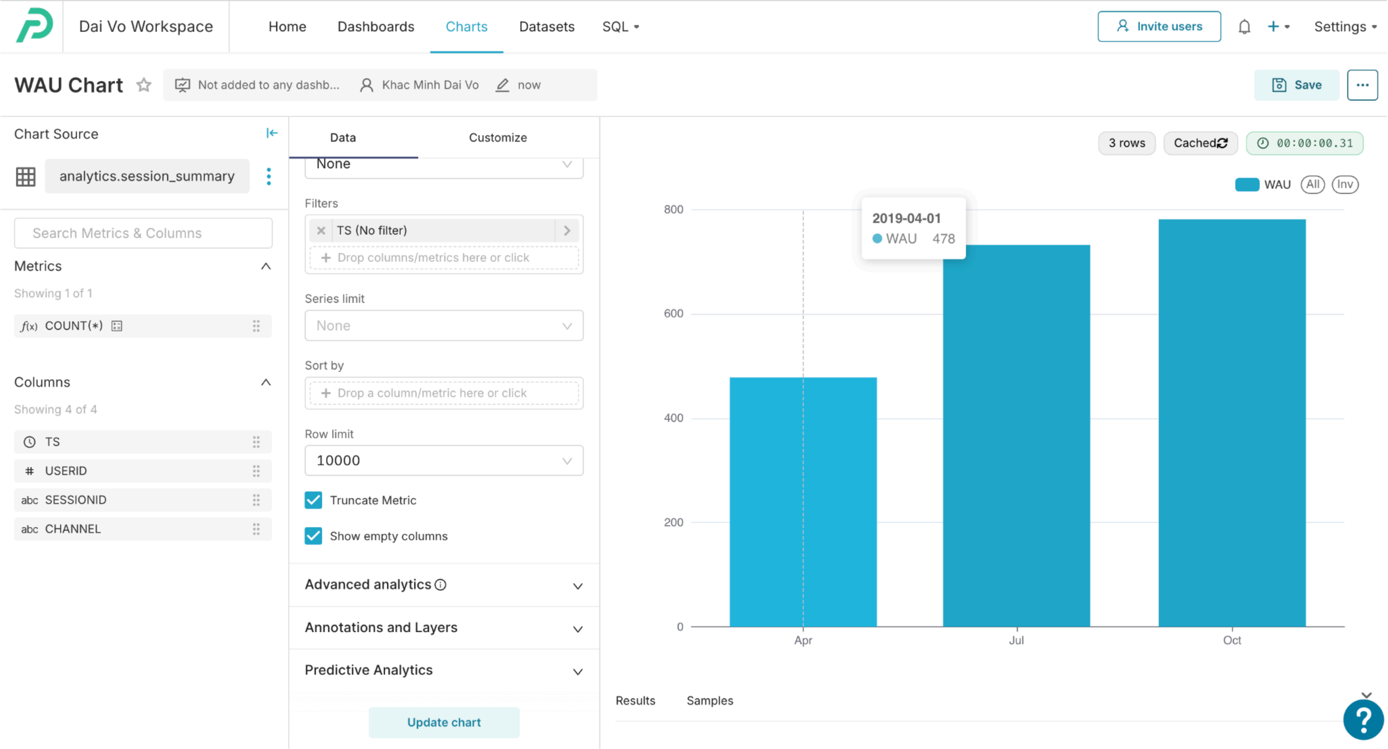
Datasets

Bulk select | + Dataset

Name: Type a value
Type: Select or type a value
Database: Select or type a value
Schema: Select or type a value
Owner: Select or type a value
Certified: Select or type a value
Modified by: Select or type a value

Name	Type	Database	Schema	Owners	Last modified	Actions
session_summary	Physical	Snowflake	analytics	KV	18 minutes ago	

4. **Create your WAU chart (+2pt)**
   a. Make sure you rename the metrics field to **WAU**
   b. Capture the screenshot of the chart (#4)



WAU Chart

Not added to any dashb... | Khac Minh Dai Vo | now

Chart Source: analytics.session_summary

Search Metrics & Columns

Metrics — Showing 1 of 1
- COUNT(*)

Columns — Showing 4 of 4
- TS
- USERID
- SESSIONID
- CHANNEL

None

Filters: TS (No filter)

Series limit: None

Sort by: Drop a column/metric here or click

Row limit: 10000

☑ Truncate Metric

☑ Show empty columns

Advanced analytics

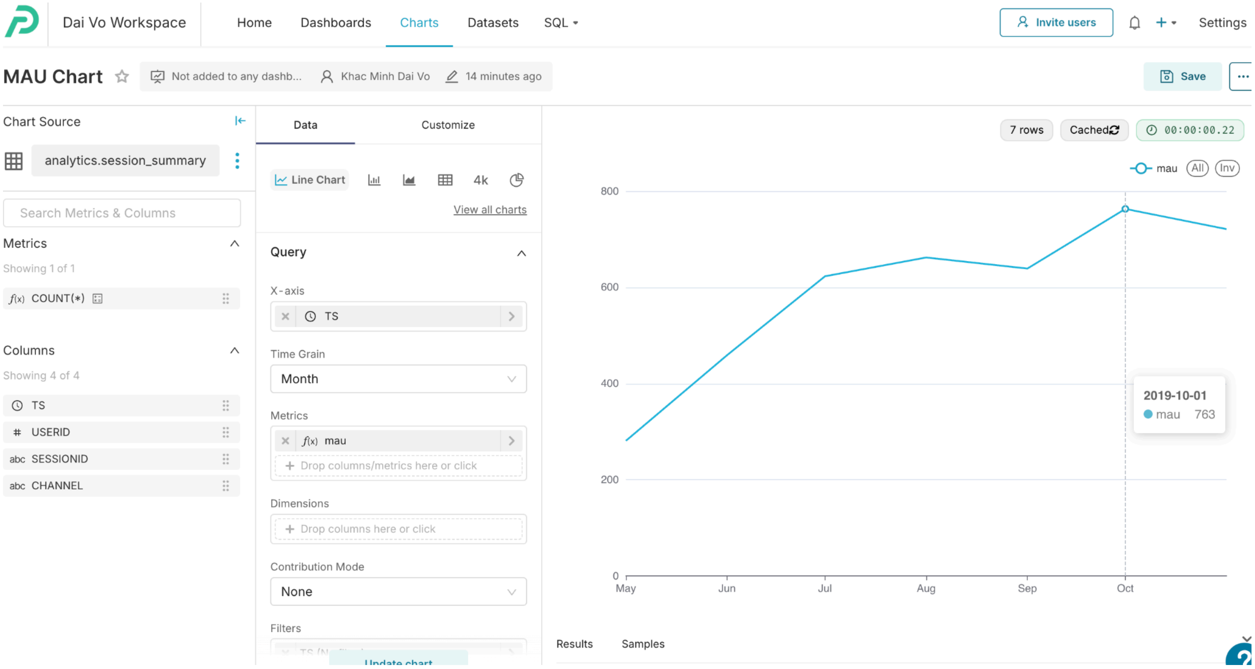
Annotations and Layers

Predictive Analytics

3 rows | Cached | 00:00:00.31

2019-04-01
● WAU 478

Note: Create a bar chart that is named WAU instead. I provided an extra chart line chart below too.



SQL Code in Snowflake:

```
-- Create Database
CREATE DATABASE IF NOT EXISTS USER_DB_BISON;

-- Create Warehouse
CREATE WAREHOUSE IF NOT EXISTS BISON_QUERY_WH
  WITH WAREHOUSE_SIZE = 'XSMALL'
  AUTO_SUSPEND = 300 -- suspends after 5 mins of inactivity
  AUTO_RESUME = TRUE
  INITIALLY_SUSPENDED = TRUE;

-- Use this specific warehouse and database
USE WAREHOUSE BISON_QUERY_WH;
USE DATABASE USER_DB_BISON;

-- Create the schema raw and analytics for the assignment
CREATE SCHEMA IF NOT EXISTS raw;
CREATE SCHEMA IF NOT EXISTS analytics;

-- Create 2 tables
CREATE TABLE IF NOT EXISTS raw.user_session_channel (
    userId INT NOT NULL,
    sessionId VARCHAR(32) PRIMARY KEY,
    channel VARCHAR(32) DEFAULT 'direct'
);
```

```
CREATE TABLE IF NOT EXISTS raw.session_timestamp (
    sessionId VARCHAR(32) PRIMARY KEY,
    ts TIMESTAMP
);
SELECT * FROM user_db_bison.raw.user_session_channel LIMIT 10;
SELECT * FROM user_db_bison.raw.session_timestamp LIMIT 10;
SELECT * FROM user_db_bison.analytics.session_summary LIMIT 10;

-- Checking for current account that used for airflow connections
SELECT CURRENT_ACCOUNT();
```