Project Report: Weather Data Pipeline

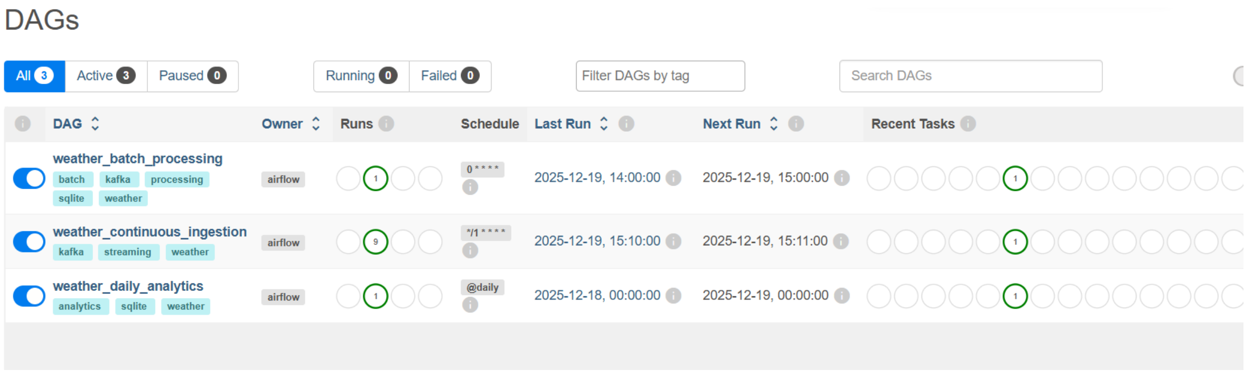
---

1. Team

Name	Role
Aubakirov Sanzhar	DAG 1 — Data Ingestion and Kafka Producer
Toremuratuly Abylay	DAG 2 — Data Cleaning and SQLite Writer
Toleu Bakhauddin	DAG 3 — Daily Analytics

Roles description:

- DAG 1: Setup and run Airflow process for periodic data collection from WeatherAPI and sending to Kafka.
- DAG 2: Cleaning and processing data via Pandas, writing to SQLite.
- DAG 3: Daily analytics — computing aggregates and saving results to summary table.



DAGs

All 3 | Active 3 | Paused 0 | Running 0 | Failed 0 | Filter DAGs by tag | Search DAGs

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks
weather_batch_processing batch kafka processing sqlite weather	airflow	1	0 * * * *	2025-12-19, 14:00:00	2025-12-19, 15:00:00	1
weather_continuous_ingestion kafka streaming weather	airflow	9	*/1 * * * *	2025-12-19, 15:10:00	2025-12-19, 15:11:00	1
weather_daily_analytics analytics sqlite weather	airflow	1	@daily	2025-12-18, 00:00:00	2025-12-19, 00:00:00	1

---

2. API

Selected API: WeatherAPI (http://api.weatherapi.com)

Selection criteria:

- Frequent data updates (every few minutes)
- Stability and well-documented
- JSON response format
- Provides real values for temperature, humidity, and weather conditions
- Not used in previous lab exercises

Example request:

GET http://api.weatherapi.com/v1/current.json?key=YOUR_API_KEY&q=Almaty&aqi=no

Example JSON response (key fields):

```
{
 "location": {"name": "Almaty"},
 "current": {
  "temp_c": -1.9,
  "humidity": 92,
  "condition": {"text": "Freezing fog"},
  "wind_kph": 5.4,
  "pressure_mb": 1012.0,
  "feelslike_c": -5.0
 }
}
```

Comment: API is stable and provides real-time weather data, meeting project requirements.

---

3. Pipeline Architecture

Overall flow:

WeatherAPI → DAG 1 (Producer) → Kafka (raw_weather_events)

  → DAG 2 (Batch Processing) → SQLite (events)

  → DAG 3 (Daily Analytics) → SQLite (daily_weather_summary)

DAG descriptions:

- DAG 1: Periodic collection of weather data and sending to Kafka.
- DAG 2: Reading from Kafka, cleaning and validation, storing into SQLite.
- DAG 3: Reading from SQLite, aggregating data (min/max/avg), storing into summary table.

---

4. Kafka Topic Schema

Field	Type	Description
timestamp	string	Request time
city	string	City
weather.current.temp_c	float	Temperature
weather.current.humidity	int	Humidity
weather.current.condition.text	string	Weather condition
metadata.source	string	Data source

Topic: raw_weather_events

Comment: Each new API event is sent to Kafka with metadata.

---

5. Cleaning Rules (DAG 2)

Processing with Pandas:

1. Check mandatory fields: timestamp, city, weather.current.
2. Type conversion: temperature_c → float, humidity → int.
3. Allowed value ranges:
    - Temperature: -100 … 100 °C
    - Humidity: 0 … 100 %
4. Extract weather condition (condition.text).
5. Additional fields: wind_kph, pressure_mb, feelslike_c.
6. Skip invalid or empty records.

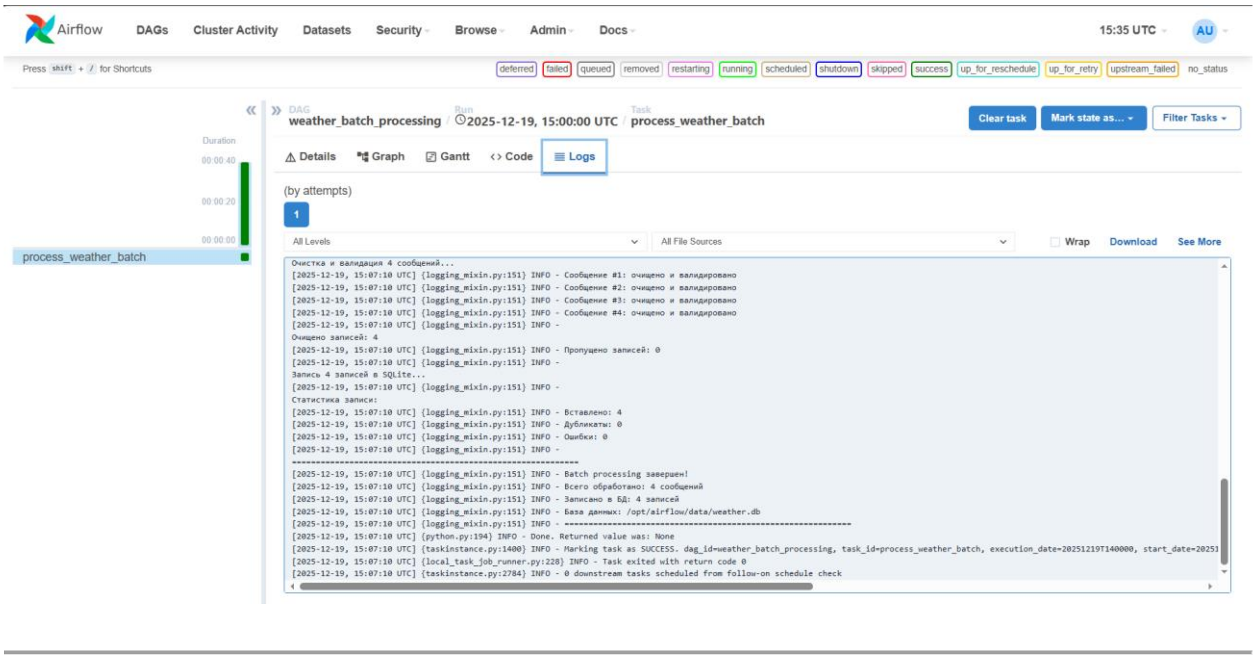
Example before/after cleaning:

Before cleaning (JSON):

timestamp	city	weather.current.temp_c	weather.current.humidity	weather.current.condition.text
2025-12-19T13:55:42	Almaty	-1.9	92	Freezing fog

After cleaning (DataFrame for SQLite):

timestamp	city	temperature_c	humidity	condition_text	wind_kph	pressure_mb	feelslike_c	source
2025-12-19T13:55:42	Almaty	-1.9	92	Freezing fog	5.4	1012.0	-5.0	weatherapi.com



```
Очистка и валидация 4 сообщений...
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Сообщение #1: очищено и валидировано
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Сообщение #2: очищено и валидировано
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Сообщение #3: очищено и валидировано
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Сообщение #4: очищено и валидировано
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO -
Очищено записей: 4
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Пропущено записей: 0
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO -
Запись 4 записей в SQLite...
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO -
Статистика записи:
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Вставлено: 4
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Дубликаты: 0
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Ошибки: 0
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO -
============================================================
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Batch processing завершен!
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Всего обработано: 4 сообщений
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - Записано в БД: 4 записей
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - База данных: /opt/airflow/data/weather.db
[2025-12-19, 15:07:10 UTC] {logging_mixin.py:151} INFO - ============================================================
[2025-12-19, 15:07:10 UTC] {python.py:194} INFO - Done. Returned value was: None
[2025-12-19, 15:07:10 UTC] {taskinstance.py:1400} INFO - Marking task as SUCCESS. dag_id=weather_batch_processing, task_id=process_weather_batch, execution_date=20251219T140000, start_date=20251
[2025-12-19, 15:07:10 UTC] {local_task_job_runner.py:228} INFO - Task exited with return code 0
[2025-12-19, 15:07:10 UTC] {taskinstance.py:2784} INFO - 0 downstream tasks scheduled from follow-on schedule check
```

6. SQLite Schema

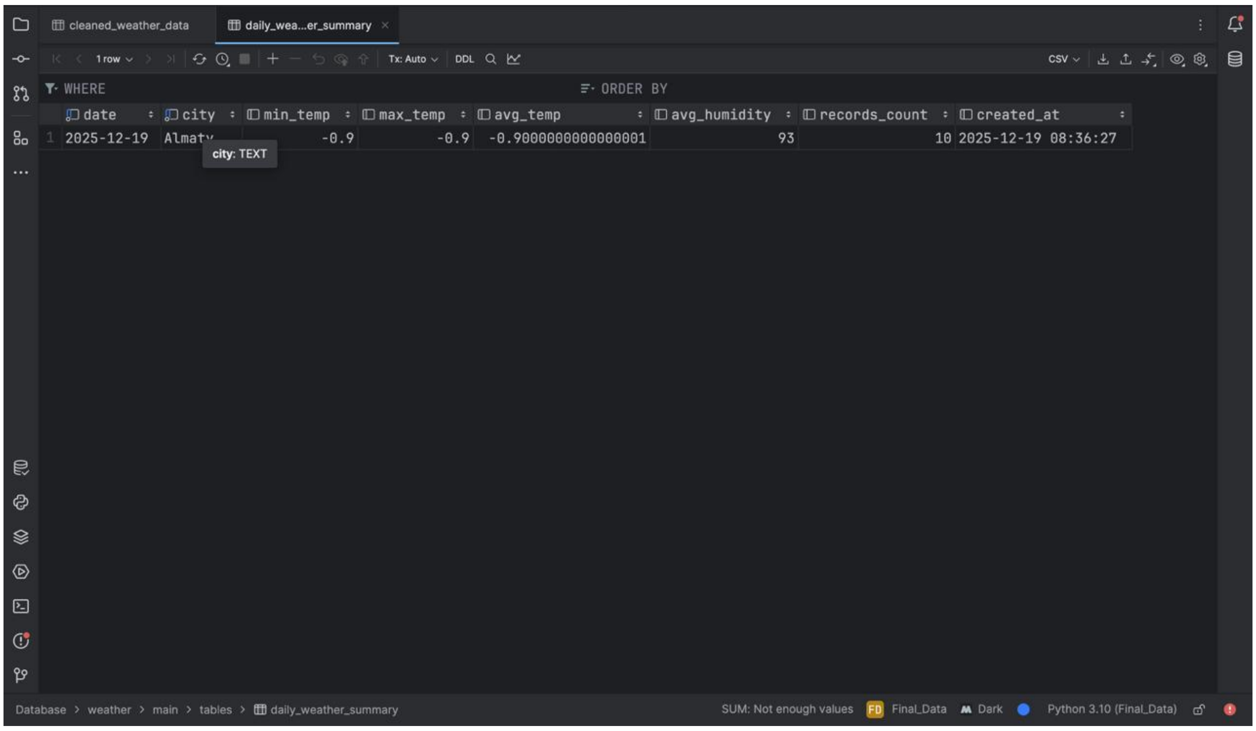
Table: events

Field	Type	Description
timestamp	TEXT	Event time
city	TEXT	City
temperature_c	REAL	Temperature
humidity	INTEGER	Humidity
condition_text	TEXT	Weather condition
wind_kph	REAL	Wind speed
pressure_mb	REAL	Pressure
feelslike_c	REAL	Feels like
source	TEXT	Data source

Table: daily_weather_summary

Field	Type	Description
date	TEXT	Date
city	TEXT	City
min_temp	REAL	Minimum temperature
max_temp	REAL	Maximum temperature
avg_temp	REAL	Average temperature
avg_humidity	REAL	Average humidity
records_count	INTEGER	Number of records
created_at	TEXT	Inserted timestamp
UNIQUE(date, city)	-	Uniqueness per date and city

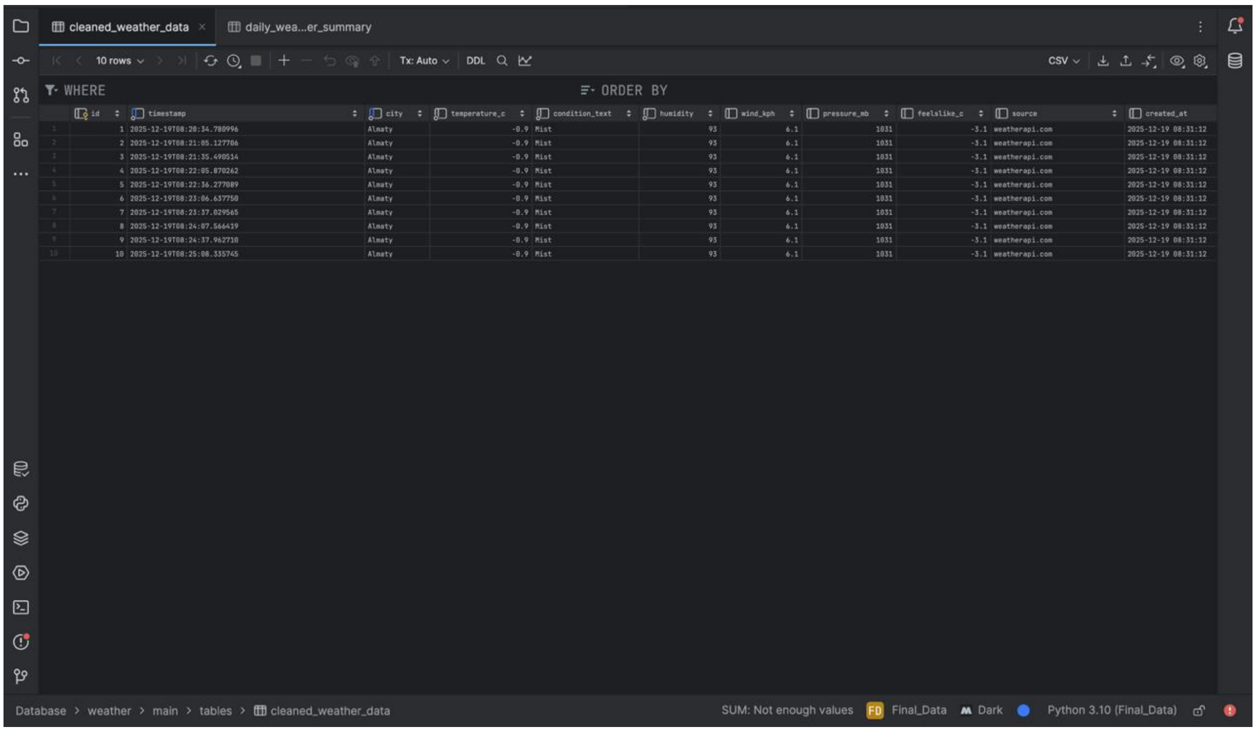
Comment: DAG 3 generates daily aggregates for each location.



	date	city	min_temp	max_temp	avg_temp	avg_humidity	records_count	created_at
1	2025-12-19	Almaty	-0.9	-0.9	-0.9000000000000001	93	10	2025-12-19 08:36:27

Database > weather > main > tables > daily_weather_summary

SUM: Not enough values   FD Final_Data   Dark   Python 3.10 (Final_Data)



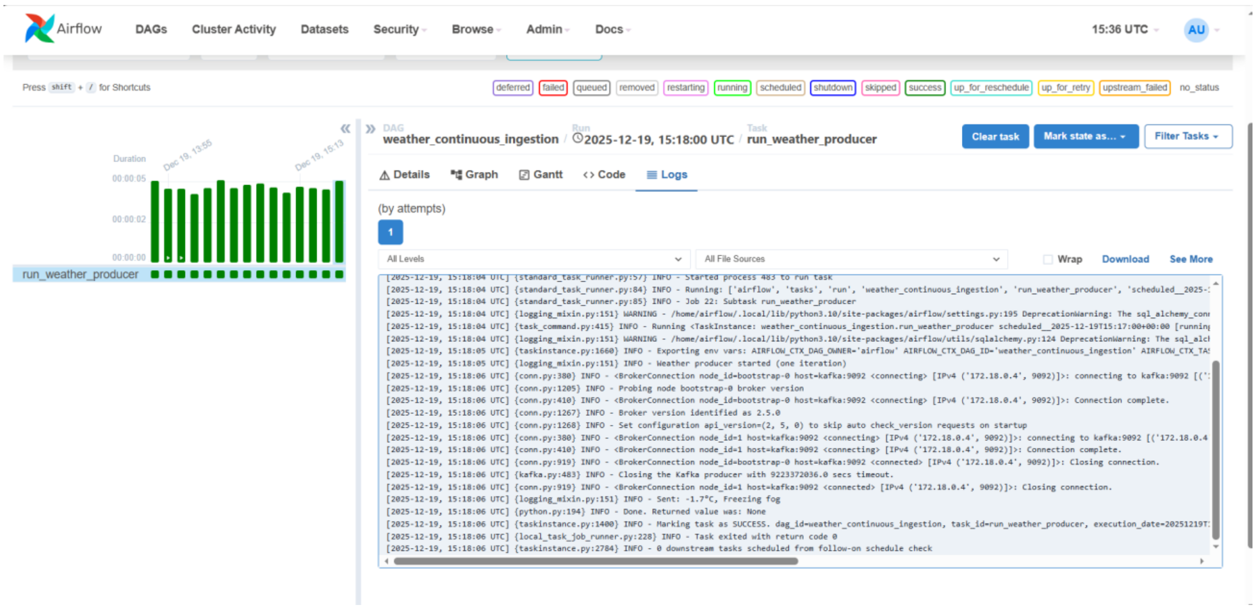
id	timestamp	city	temperature_c	condition_text	humidity	wind_kph	pressure_mb	feelslike_c	source	created_at
1	2025-12-19T08:20:34.780996	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
2	2025-12-19T08:21:05.127786	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
3	2025-12-19T08:21:35.490514	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
4	2025-12-19T08:22:05.870262	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
5	2025-12-19T08:22:36.277089	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
6	2025-12-19T08:23:06.637750	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
7	2025-12-19T08:23:37.029565	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
8	2025-12-19T08:24:07.566419	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
9	2025-12-19T08:24:37.962710	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12
10	2025-12-19T08:25:08.335745	Almaty	-0.9	Mist	93	6.1	1031	-3.1	weatherapi.com	2025-12-19 08:31:12

7. DAGs and Logs

DAG 1 — Weather Ingestion

- Periodic data collection from WeatherAPI, triggered every minute (one fetch per run).
- Sends data to Kafka topic raw_weather_events.
- Simulates streaming without blocking other DAGs.
- Log snippet:

```
Weather producer started (one iteration)
Sent: -1.9°C, Freezing fog
Task marked as SUCCESS
```



```
[2025-12-19, 15:18:04 UTC] {standard_task_runner.py:57} INFO - Started process 483 to run task
[2025-12-19, 15:18:04 UTC] {standard_task_runner.py:84} INFO - Running: ['airflow', 'tasks', 'run', 'weather_continuous_ingestion', 'run_weather_producer', 'scheduled__2025-1
[2025-12-19, 15:18:04 UTC] {standard_task_runner.py:85} INFO - Job 22: Subtask run_weather_producer
[2025-12-19, 15:18:04 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/settings.py:195 DeprecationWarning: The sql_alchemy_conn
[2025-12-19, 15:18:04 UTC] {task_command.py:415} INFO - Running <TaskInstance: weather_continuous_ingestion.run_weather_producer scheduled__2025-12-19T15:17:00+00:00 [running
[2025-12-19, 15:18:04 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/utils/sqlalchemy.py:124 DeprecationWarning: The sql_alch
[2025-12-19, 15:18:05 UTC] {taskinstance.py:1660} INFO - Exporting env vars: AIRFLOW_CTX_DAG_OWNER='airflow' AIRFLOW_CTX_DAG_ID='weather_continuous_ingestion' AIRFLOW_CTX_TAS
[2025-12-19, 15:18:05 UTC] {logging_mixin.py:151} INFO - Weather producer started (one iteration)
[2025-12-19, 15:18:06 UTC] {conn.py:380} INFO - <BrokerConnection node_id=bootstrap-0 host=kafka:9092 <connecting> [IPv4 ('172.18.0.4', 9092)]>: connecting to kafka:9092 [('1
[2025-12-19, 15:18:06 UTC] {conn.py:1205} INFO - Probing node bootstrap-0 broker version
[2025-12-19, 15:18:06 UTC] {conn.py:410} INFO - <BrokerConnection node_id=bootstrap-0 host=kafka:9092 <connecting> [IPv4 ('172.18.0.4', 9092)]>: Connection complete.
[2025-12-19, 15:18:06 UTC] {conn.py:1267} INFO - Broker version identified as 2.5.0
[2025-12-19, 15:18:06 UTC] {conn.py:1268} INFO - Set configuration api_version=(2, 5, 0) to skip auto check_version requests on startup
[2025-12-19, 15:18:06 UTC] {conn.py:380} INFO - <BrokerConnection node_id=1 host=kafka:9092 <connecting> [IPv4 ('172.18.0.4', 9092)]>: connecting to kafka:9092 [('172.18.0.4
[2025-12-19, 15:18:06 UTC] {conn.py:410} INFO - <BrokerConnection node_id=1 host=kafka:9092 <connecting> [IPv4 ('172.18.0.4', 9092)]>: Connection complete.
[2025-12-19, 15:18:06 UTC] {conn.py:919} INFO - <BrokerConnection node_id=bootstrap-0 host=kafka:9092 <connected> [IPv4 ('172.18.0.4', 9092)]>: Closing connection.
[2025-12-19, 15:18:06 UTC] {kafka.py:483} INFO - Closing the Kafka producer with 9223372036.0 secs timeout.
[2025-12-19, 15:18:06 UTC] {conn.py:919} INFO - <BrokerConnection node_id=1 host=kafka:9092 <connected> [IPv4 ('172.18.0.4', 9092)]>: Closing connection.
[2025-12-19, 15:18:06 UTC] {logging_mixin.py:151} INFO - Sent: -1.7°C, Freezing fog
[2025-12-19, 15:18:06 UTC] {python.py:194} INFO - Done. Returned value was: None
[2025-12-19, 15:18:06 UTC] {taskinstance.py:1400} INFO - Marking task as SUCCESS. dag_id=weather_continuous_ingestion, task_id=run_weather_producer, execution_date=20251219T1
[2025-12-19, 15:18:06 UTC] {local_task_job_runner.py:228} INFO - Task exited with return code 0
[2025-12-19, 15:18:06 UTC] {taskinstance.py:2784} INFO - 0 downstream tasks scheduled from follow-on schedule check
```

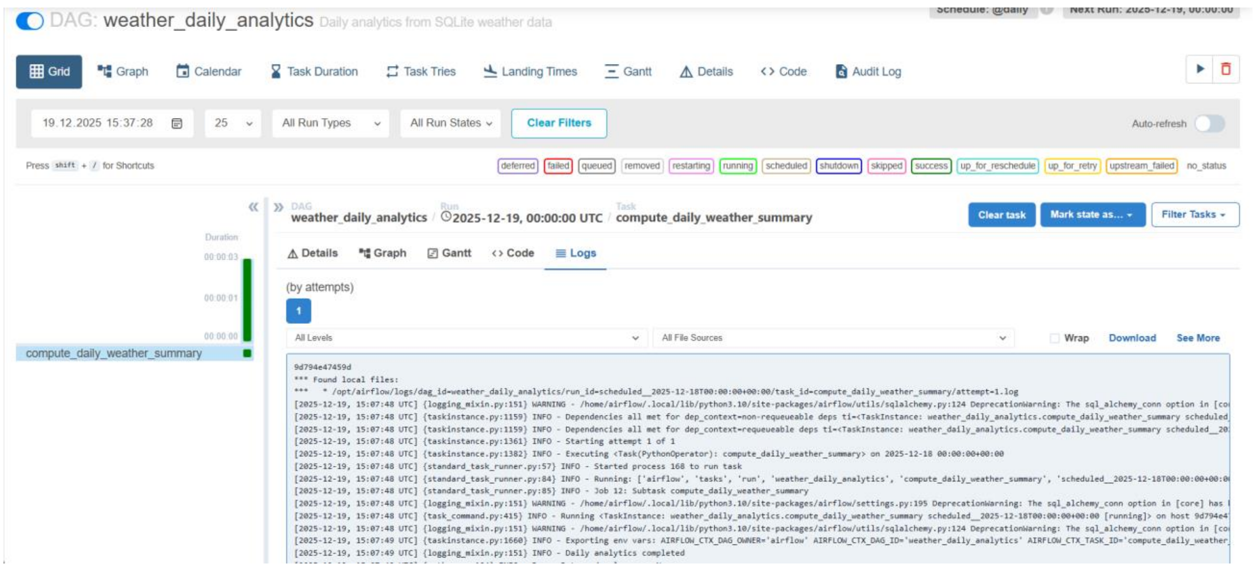
DAG 2 — Batch Processing

- Reads data from Kafka.
- Cleans and validates with Pandas.
- Stores into SQLite table events.
- Log snippet:

Total messages processed: 4

Inserted into DB: 4

Batch processing completed

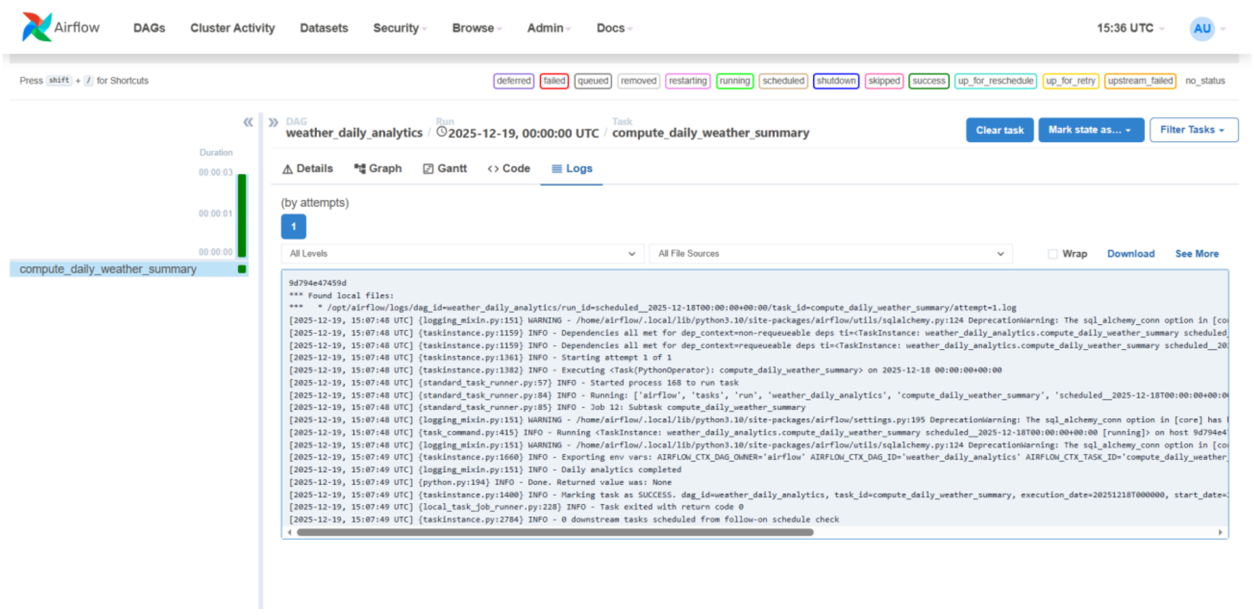


```
9d794e47459d
*** Found local files:
***   * /opt/airflow/logs/dag_id=weather_daily_analytics/run_id=scheduled__2025-12-18T00:00:00+00:00/task_id=compute_daily_weather_summary/attempt-1.log
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/utils/sqlalchemy.py:124 DeprecationWarning: The sql_alchemy_conn option in [co
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1159} INFO - Dependencies all met for dep_context=non-requeueable deps ti=<TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled_
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1159} INFO - Dependencies all met for dep_context=requeueable deps ti=<TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled__20
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1361} INFO - Starting attempt 1 of 1
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1382} INFO - Executing <Task(PythonOperator): compute_daily_weather_summary> on 2025-12-18 00:00:00+00:00
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:57} INFO - Started process 168 to run task
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:84} INFO - Running: ['airflow', 'tasks', 'run', 'weather_daily_analytics', 'compute_daily_weather_summary', 'scheduled__2025-12-18T00:00:00+00:0
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:85} INFO - Job 12: Subtask compute_daily_weather_summary
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/settings.py:195 DeprecationWarning: The sql_alchemy_conn option in [core] has
[2025-12-19, 15:07:48 UTC] {task_command.py:415} INFO - Running <TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled__2025-12-18T00:00:00+00:00 [running]> on host 9d794e4
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/utils/sqlalchemy.py:124 DeprecationWarning: The sql_alchemy_conn option in [co
[2025-12-19, 15:07:49 UTC] {taskinstance.py:1660} INFO - Exporting env vars: AIRFLOW_CTX_DAG_OWNER='airflow' AIRFLOW_CTX_DAG_ID='weather_daily_analytics' AIRFLOW_CTX_TASK_ID='compute_daily_weather_
[2025-12-19, 15:07:49 UTC] {logging_mixin.py:151} INFO - Daily analytics completed
```

DAG 3 — Daily Analytics

- Reads data from SQLite (events).
- Computes min/max/avg temperature and avg humidity.
- Writes to daily_weather_summary.
- Log snippet:

Daily analytics completed



```
9d794e47459d
*** Found local files:
***   * /opt/airflow/logs/dag_id=weather_daily_analytics/run_id=scheduled__2025-12-18T00:00:00+00:00/task_id=compute_daily_weather_summary/attempt=1.log
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/utils/sqlalchemy.py:124 DeprecationWarning: The sql_alchemy_conn option in [co
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1159} INFO - Dependencies all met for dep_context=non-requeueable deps ti=<TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled_
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1159} INFO - Dependencies all met for dep_context=requeueable deps ti=<TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled__20
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1361} INFO - Starting attempt 1 of 1
[2025-12-19, 15:07:48 UTC] {taskinstance.py:1382} INFO - Executing <Task(PythonOperator): compute_daily_weather_summary> on 2025-12-18 00:00:00+00:00
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:57} INFO - Started process 168 to run task
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:84} INFO - Running: ['airflow', 'tasks', 'run', 'weather_daily_analytics', 'compute_daily_weather_summary', 'scheduled__2025-12-18T00:00:00+00:0
[2025-12-19, 15:07:48 UTC] {standard_task_runner.py:85} INFO - Job 12: Subtask compute_daily_weather_summary
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/settings.py:195 DeprecationWarning: The sql_alchemy_conn option in [core] has
[2025-12-19, 15:07:48 UTC] {task_command.py:415} INFO - Running <TaskInstance: weather_daily_analytics.compute_daily_weather_summary scheduled__2025-12-18T00:00:00+00:00 [running]> on host 9d794e4
[2025-12-19, 15:07:48 UTC] {logging_mixin.py:151} WARNING - /home/airflow/.local/lib/python3.10/site-packages/airflow/utils/sqlalchemy.py:124 DeprecationWarning: The sql_alchemy_conn option in [co
[2025-12-19, 15:07:49 UTC] {taskinstance.py:1660} INFO - Exporting env vars: AIRFLOW_CTX_DAG_OWNER='airflow' AIRFLOW_CTX_DAG_ID='weather_daily_analytics' AIRFLOW_CTX_TASK_ID='compute_daily_weather_
[2025-12-19, 15:07:49 UTC] {logging_mixin.py:151} INFO - Daily analytics completed
[2025-12-19, 15:07:49 UTC] {python.py:194} INFO - Done. Returned value was: None
[2025-12-19, 15:07:49 UTC] {taskinstance.py:1400} INFO - Marking task as SUCCESS. dag_id=weather_daily_analytics, task_id=compute_daily_weather_summary, execution_date=20251218T000000, start_date=
[2025-12-19, 15:07:49 UTC] {local_task_job_runner.py:228} INFO - Task exited with return code 0
[2025-12-19, 15:07:49 UTC] {taskinstance.py:2784} INFO - 0 downstream tasks scheduled from follow-on schedule check
```

8. Analytics Example

Almaty, 19.12.2025

City	min_temp	max_temp	avg_temp	avg_humidity	records_count
Almaty	-2.0	0.5	-1.2	90	24

---

9. Repository

Project structure:

```
project/
│ README.md
│ requirements.txt
├─ src/
│   ├─ job1_producer.py
│   ├─ job2_processor.py
│   ├─ job3_analytics.py
│   └─ database.py
├─ airflow/
│   └─ dags/
│       ├─ dag1_weather_ingestion.py
│       ├─ dag2_weather_batch.py
│       └─ dag3_daily_analytics.py
├─ data/
│   └─ weather.db
└─ report/
    └─ report.pdf
```
