

Департамент образования и науки города Москвы
Государственное автономное образовательное учреждение высшего
образования города Москвы
«Московский городской педагогический университет»
Институт цифрового образования
Департамент информатики, управления и технологий

Выполнила: st_95

ЛАБОРАТОРНАЯ РАБОТА 6_1

Разработка полного ETL-процесса. Оркестровка конвейера данных

Проектный практикум по разработке ETL-решений

Направление подготовки

38.03.05 Бизнес-информатика

Профиль подготовки

Аналитика данных и эффективное управление

Москва

2025

Выполнение общего задания

Ход работы:

1. Клонирование репозитория и переход в директорию (Рисунок 1)

```
dev@dev-vm: ~/workshop-on-ETL/business_case_stocksense_25
dev@dev-vm:~$ ls
Desktop      google-chrome-stable_current_amd64.deb  Public      thinclient_drives
Documents    Music                                    snap        Videos
Downloads    Pictures                                Templates
dev@dev-vm:~$ git clone https://github.com/BosenkoTM/workshop-on-ETL.git
Cloning into 'workshop-on-ETL'...
remote: Enumerating objects: 637, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 637 (delta 18), reused 1 (delta 1), pack-reused 607 (from 1)
Receiving objects: 100% (637/637), 5.83 MiB | 4.43 MiB/s, done.
Resolving deltas: 100% (316/316), done.
dev@dev-vm:~$ cd workshop-on-ETL
dev@dev-vm:~/workshop-on-ETL$ ls
business_case_rocket      data_for_lessons
business_case_rocket_25   lecture_0_airflow
business_case_stocksense  lectures
business_case_stocksense_25 'Processing supermarket promotions data'
business_case_umbrella    README.md
business_case_umbrella_25
dev@dev-vm:~/workshop-on-ETL$ cd business_case_stocksense_25
dev@dev-vm:~/workshop-on-ETL/business_case_stocksense_25$ ls
dags  docker-compose.yml  Dockerfile  README.md  scripts
dev@dev-vm:~/workshop-on-ETL/business_case_stocksense_25$
```

Рисунок 1

2. Подготовка работы (настройка рабочей среды):

2.1. Удаление всех контейнеров (Рисунок 2)

```
dev@dev-vm:~/workshop-on-ETL/business_case_stocksense_25$ sudo docker rm -f $(sudo docker ps -a -q)
14541fab7425
e8646cc9632b
eec2bc8fe5f9
ca2b45dadbf9
87f99a92f565
4620c870eea3
fe9c2bc51651
b9be47acb410
8af79c5c0b06
1e2fd0c99ff8
3ee727dea212
1fcafbd1e5c8
de682d7b1d90
e77328282a3b
032fd71df113
98fc34d4f718
b26bed0c5630
9801b3cb26db
```

Рисунок 2

Проверка контейнеров (Рисунок 3)

```
dev@dev-vm:~/workshop-on-ETL/business_case_stocksense_25$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
```

Рисунок 3

2.2. Удаление всех образов (Рисунок 4)

```

dev@dev-vm: ~/workshop-on-ETL/business_case_stocksense_25$ sudo docker rmi -f $(sudo docker images -q)
Untagged: lab_0_webinar-faker-api:latest
Deleted: sha256:01365d6a726d9035daf08f05938af1e991f9c2ae031acdb1f77ef7864b38f15d
Untagged: quay.io/minio/minio:latest
Untagged: quay.io/minio/minio@sha256:46b3009bf7041ee9bd90bd0d2b38c6ddc24d20a35d609551a1802c558c1c958f
Deleted: sha256:2eaf94c71682e852c0c74304cfc9bb88fa4e1d2fba86c6170bc85c4e5d3d88f5
Deleted: sha256:73bb4294054969a778b76102058aef041d3e4b08e6e23c8cbafaafa808481520
Deleted: sha256:e94618657f68b0b919604d8b7baf9e67990e53a6d6151ecc00de85eb96162759
Deleted: sha256:39530d2fd149231fdb8615f10994d789219e250d90bf880293fd019d62798b26
Deleted: sha256:a80df31a7b8c2e1c783b78a4ee7be82750a6831302f3e1de481b068565a52b49
Deleted: sha256:c501b2615161b1101dc14af0d1908c65ba2bf7de3d25e725225aef4801f993a
Deleted: sha256:f0cca71c48727b48fad4b4180f37c60453d52bc8e8fe48fdb5e6cb0c6315c95
Deleted: sha256:8f2981c601c3fffe8b654322601355762d26dbf148658022cbb578b63d2f861
Deleted: sha256:30a72e192ad254406ca3d2a1180b6b854c78b3600c1686f51f90c3c958821aa7
Deleted: sha256:919873557493c9ed2d61c98f006ff126f5b96121c051793cac9954e1e2822e07
Deleted: sha256:639ae6ef0f6cee73213a007849ae6cca6f10a06b6894da4cd6a175cd75e3c4a2

```

Рисунок 4

2.3. Создание нового образа (Рисунок 5)

```

dev@dev-vm: ~/workshop-on-ETL/business_case_stocksense_25$ sudo docker build -t custom-airflow:slim-2.8.1-python3.11 .
2025/04/04 11:08:35 in: [{}string{}]
2025/04/04 11:08:35 Parsed entitlements: []
[+] Building 111.5s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 615B
=> [internal] load metadata for docker.io/apache/airflow:slim-2.8.1-python3.11
=> [internal] load .dockerignore
=> transferring context: 2B
[1/3] FROM docker.io/apache/airflow:slim-2.8.1-python3.11@sha256:751badd58a83e44ae23c393fe1552196c25f3e2683c97db1a6d98b7d15e7a0e8
=> resolve docker.io/apache/airflow:slim-2.8.1-python3.11@sha256:751badd58a83e44ae23c393fe1552196c25f3e2683c97db1a6d98b7d15e7a0e8
=> sha256:3ee88b8d12ebb0fbb9be864918a05a7621f1b4e1881154b2a0bd64e9476c333 4.47kB / 4.47kB
=> sha256:e1caac4eb9d2ec24aa3618e5992208321a92492aef5fef5eb9e470895f771c56 29.12MB / 29.12MB
=> sha256:a205efa96734ac8633bf8d388ed9b6cd527835d31ebec070ba1cedfb880b4ca4 25.59kB / 25.59kB
=> sha256:fe87ad6b112e2dfa9a52f49adf5bb70a80d9af2c737f71a947cb5017a12b148 12.87MB / 12.87MB
=> sha256:751badd58a83e44ae23c393fe1552196c25f3e2683c97db1a6d98b7d15e7a0e8 1.61kB / 1.61kB
=> sha256:51d1f07906b71fd60ac43c61035514996a8ad8dbfd39d4f570ac5446b064ee5d 3.51MB / 3.51MB
=> sha256:4d8ccb72bbadfe34ab482a41ca4c7c07b97dfa2e523cb6317b4ff5948244765b 2448 / 2448
=> extracting sha256:e1caac4eb9d2ec24aa3618e5992208321a92492aef5fef5eb9e470895f771c56

```

Рисунок 5

2.4. Запуск среды (Рисунок 6)

```

dev@dev-vm: ~/workshop-on-ETL/business_case_stocksense_25$ sudo docker compose up --build
[+] Running 13/13
✓ wiki_results Pulled
✓ postgres Pulled
✓ 1f3e46996e29 Pull complete
✓ 47e20ba03731 Pull complete
✓ 101b82465a4f Pull complete
✓ 319529a7ccb0 Pull complete
✓ c2f9392cfd4c Pull complete
✓ 4e04446ce95d Pull complete
✓ 47bfe778b869 Pull complete
✓ b1d66b287aa8 Pull complete
✓ 7865e52a4759 Pull complete
✓ 7d75f14147c2 Pull complete
✓ 11052a5424e7 Pull complete
[+] Running 8/8
✓ Network business case stocksense 25 default Created

```

Рисунок 6

2.5. Подключение к БД (Рисунок 7):

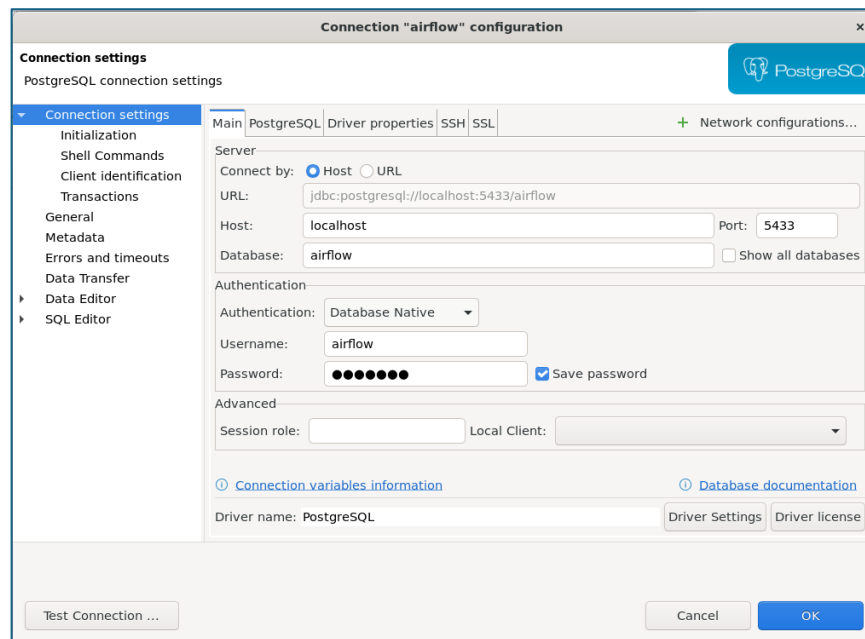


Рисунок 7

Тестирование подключения (Рисунок 8).

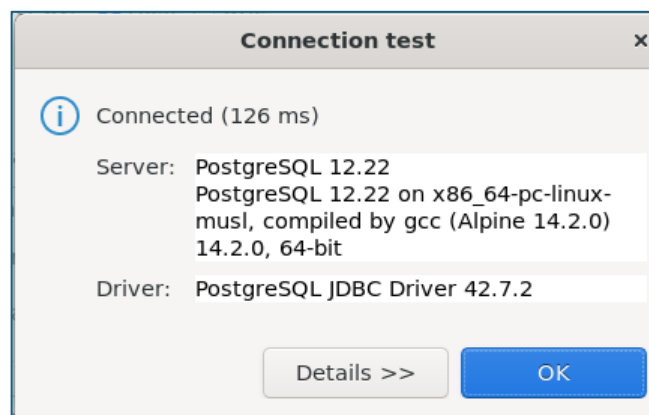


Рисунок 8

Проверка таблицы (Рисунок 9)

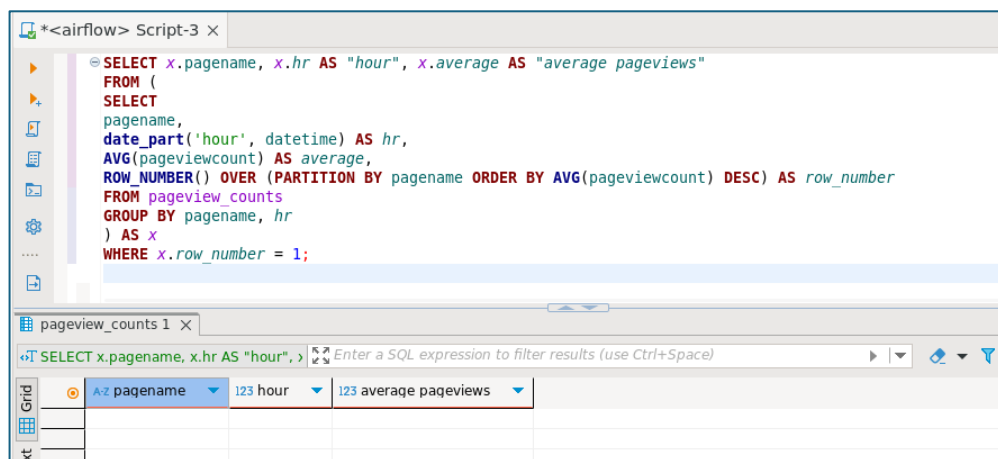


Рисунок 9

2.6. Проверка подключения к Airflow через браузер (Рисунок 10)

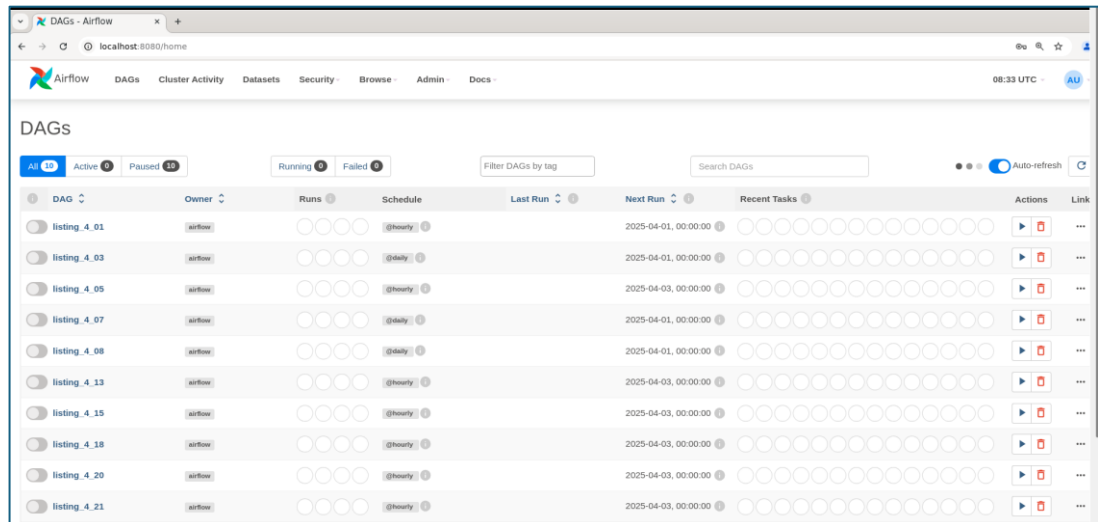


Рисунок 10

3. Проанализируем имеющиеся DAG:

- listing_4_1 - загружает данные о просмотрах страниц Wikipedia за определенный час в формате .gz с помощью curl (Bash-команда).
- listing_4_3 - демонстрирует передачу контекста Airflow в Python-функцию.
- listing_4_5 - аналог listing_4_1, но загрузка реализована на Python через urllib.request.
- listing_4_8 - демонстрирует использование execution_date и next_execution_date из контекста.
- listing_4_13 - загружает данные Wikipedia (как listing_4_5.py), но использует шаблонизацию Airflow ({{ ... }}) для передачи параметров.
- listing_4_15 - полноценный ETL-пайплайн:
 - Загружает данные Wikipedia.
 - Распаковывает .gz-архив.
 - Анализирует просмотры для 5 компаний (Google, Amazon и др.).
- listing_4_18 - расширенная версия listing_4_15.py с генерацией SQL-запроса для PostgreSQL.
- listing_4_20 и listing_4_21 (идентичны) - полный ETL с загрузкой в PostgreSQL:

- Загрузка данных.
- Распаковка.
- Анализ просмотров.
- Сохранение в БД через PostgresOperator.

Создание connection (Рисунок 11)

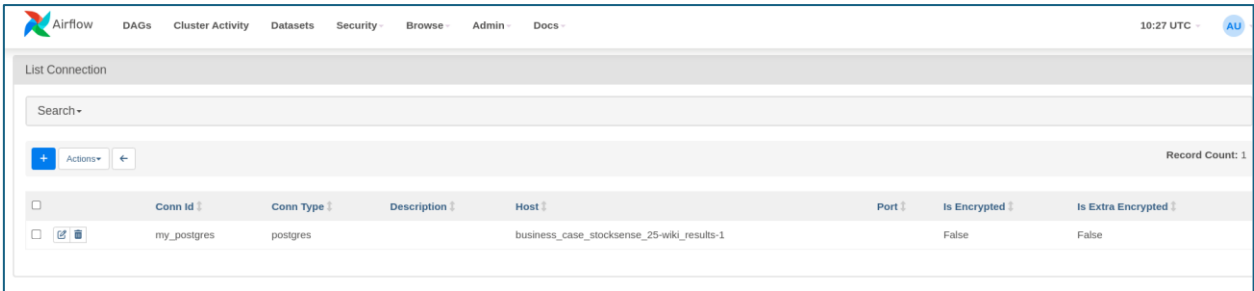


Рисунок 11

Запуск Dag (Рисунок 12), все задачи были успешно выполнены.

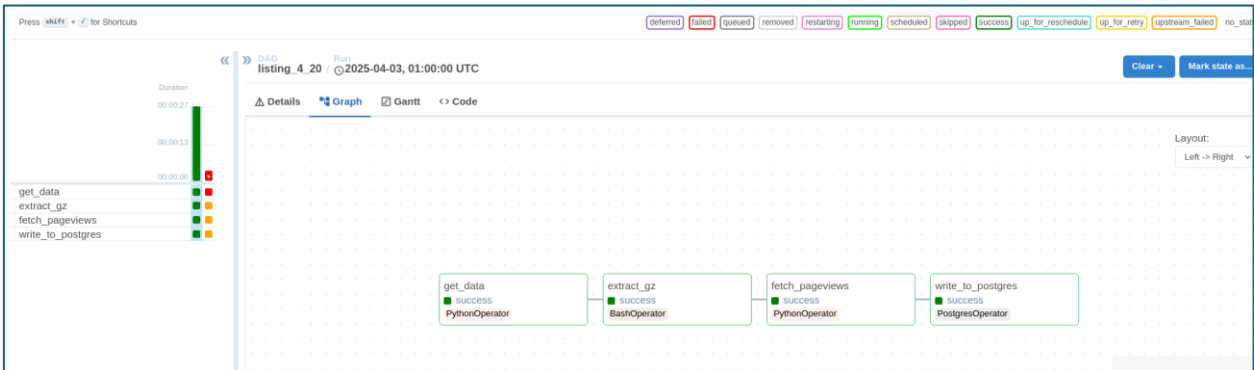


Рисунок 12

Проверка данных в postgresql (Рисунок 13), данные были успешно получены и записаны в базу данных.

pageview_counts			
Enter a SQL expression to filter results (use Ctrl+Space)			
	A-z pagename	123 pageviewcount	datetime
1	Amazon	6	2025-04-03 00:00:00.000
2	Facebook	215	2025-04-03 00:00:00.000
3	Apple	38	2025-04-03 00:00:00.000
4	Microsoft	137	2025-04-03 00:00:00.000
5	Google	330	2025-04-03 00:00:00.000

Рисунок 13

Google является самой просматриваемой страницей на википедии.

Индивидуальное задание:

Вариант 8

Получить данные за день для сайта Tinkoff. Скачайте данные за день для страницы Tinkoff и сохраните их в базе данных. Напишите SQL-запрос для подсчета количества просмотров за день, а затем визуализируйте данные.

Гипотеза:

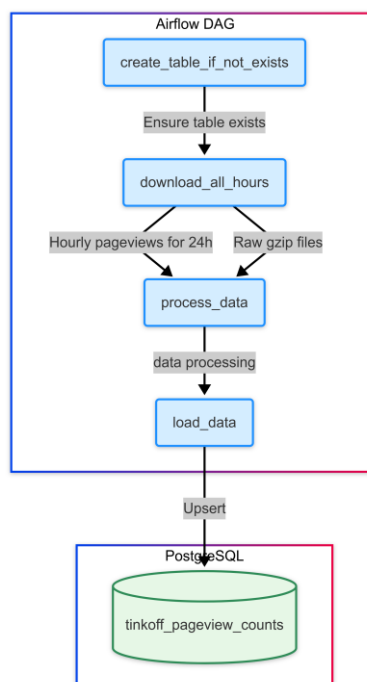
Просмотры страниц о компании в Википедии могут отражать интерес инвесторов и публики к бренду. Резкий рост запросов часто предшествует изменениям котировок, так как:

- Увеличение интереса → Потенциальный рост числа инвесторов
- Обсуждение в СМИ/соцсетях → Часто ведет к поиску информации в Википедии
- Корпоративные события (например, выпуск новых продуктов) → Одновременно влияют и на интерес, и на акции.

Постановка задачи:

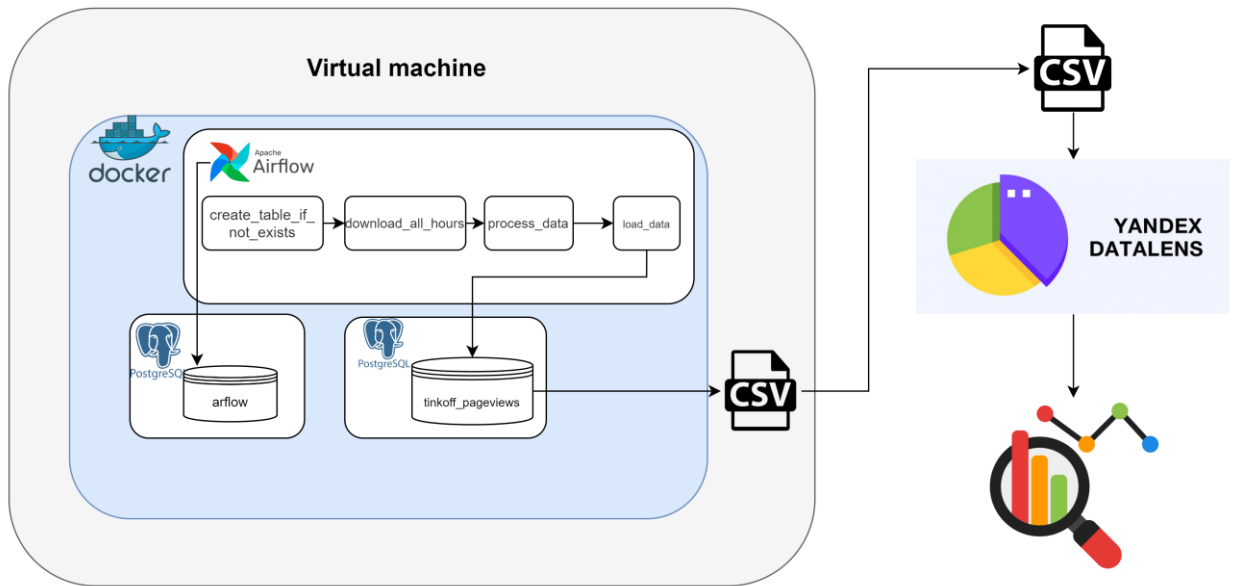
Получить и проанализировать данные о количестве просмотров страниц википедии, связанных с Т-Банком за 23 февраля 2025 года и сравнить с ростом акции за следующий день.

Архитектура DAG



Архитектура решения

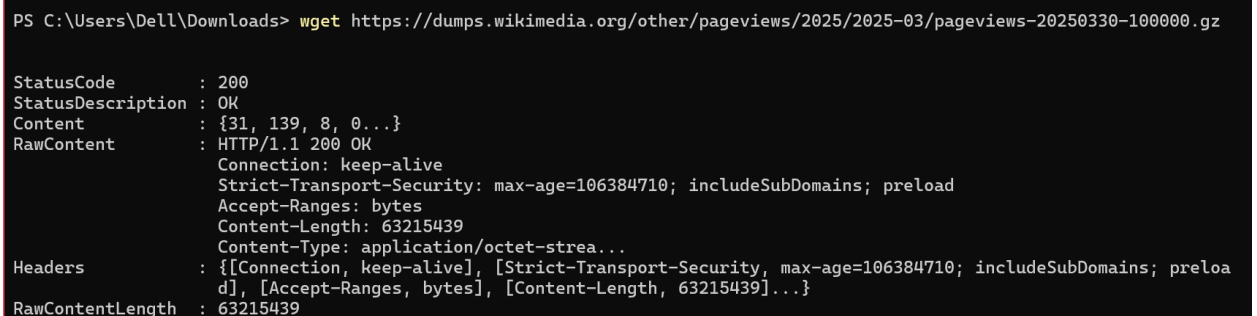
HOST OS



Ход работы:

1. Проанализируем данные, которые есть в википедии

Выгрузим дампы на локальной машине для того, чтобы проверить, какие данные есть, которые относятся к Tinkoff за 10 часов 30 марта (Рисунок 14)



```
PS C:\Users\Dell\Downloads> wget https://dumps.wikimedia.org/other/pageviews/2025/2025-03/pageviews-20250330-100000.gz

StatusCode      : 200
StatusDescription : OK
Content         : {31, 139, 8, 0...}
RawContent      : HTTP/1.1 200 OK
                  Connection: keep-alive
                  Strict-Transport-Security: max-age=106384710; includeSubDomains; preload
                  Accept-Ranges: bytes
                  Content-Length: 63215439
                  Content-Type: application/octet-stream
Headers         : {[Connection, keep-alive], [Strict-Transport-Security, max-age=106384710; includeSubDomains; preload], [Accept-Ranges, bytes], [Content-Length, 63215439]...}
RawContentLength : 63215439
```

Рисунок 14

Откроем файл и найдём все данные с page_name на английском и русском языке:

- ru Тинькофф_банк
- ru.m Тинькофф
- ru.m Тинькофф_банк
- en.m Tinkoff
- en.m Tinkoff_Bank

Так как у Тинькофф был ребрейдинг, найдём данные также по новому названию «Т-Банк»:

- de T-Bank
- en T-Bank
- en.m T-Bank
- ru Т-Банк
- ru.m Т-Банк
- uk.m Т-Банк

Возьмём данные за день – 23 марта 2025 г.

2. Напишем листинг для DAG, который будет собирать данные за 24 часа, фильтровать их по нужным значениям и загружать в базу данных (Рисунок 15)

```

dags > tinkoff_dag.py
1 from airflow import DAG
2 from airflow.operators.python import PythonOperator
3 from airflow.providers.postgres.hooks.postgres import PostgresHook
4 from datetime import datetime, timedelta
5 import logging
6 import gzip
7 from urllib import request
8 import os
9 from urllib.error import HTTPError, URLError
10
11 default_args = {
12     'owner': 'airflow',
13     'depends_on_past': False,
14     'retries': 3,
15     'retry_delay': timedelta(minutes=5),
16 }
17
18 dag = DAG(
19     dag_id="tinkoff_page_views",
20     start_date=datetime(2025, 2, 23),
21     end_date=datetime(2025, 2, 24),
22     schedule_interval="@once",
23     catchup=False,
24     default_args=default_args,
25     max_active_runs=1
26 )
27
28 # 1. Функция для скачивания данных
29 def download_all_hours(**context):
30     logical_date = context['logical_date']
31     output_dir = "/tmp/wikimedia_pageviews_split"
32     os.makedirs(output_dir, exist_ok=True)
33
34     downloaded_files = []
35
36     for hour in range(24):
37         url = (
38             f"https://dumps.wikimedia.org/other/pageviews/"
39             f"{logical_date.year}/{logical_date.year}-{logical_date.month:02d}/"
40             f"pageviews-{logical_date.year}-{logical_date.month:02d}"
41             f"{logical_date.day:02d}-{hour:02d}0000.gz"
42         )

```

Рисунок 15

3. Запускаем DAG и дожидаемся завершения выполнения задач (Рисунок 16)

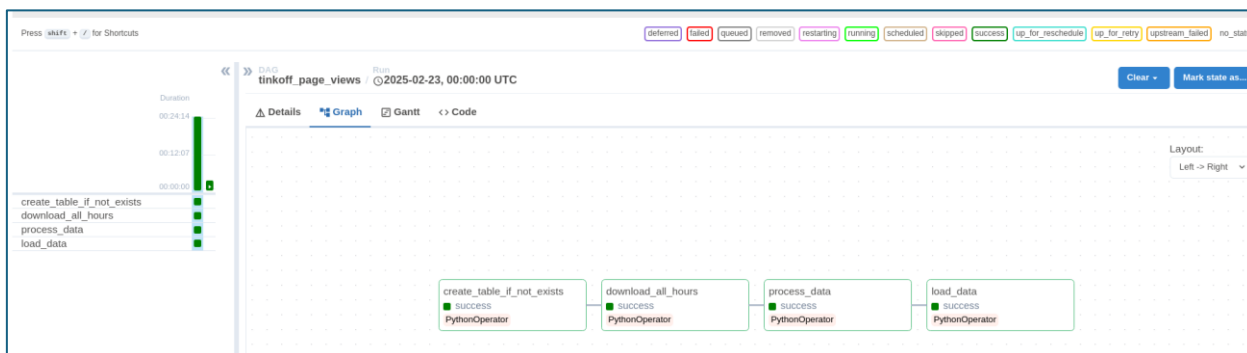


Рисунок 16

Диаграмма Ганта выполненного Dag (Рисунок 17).

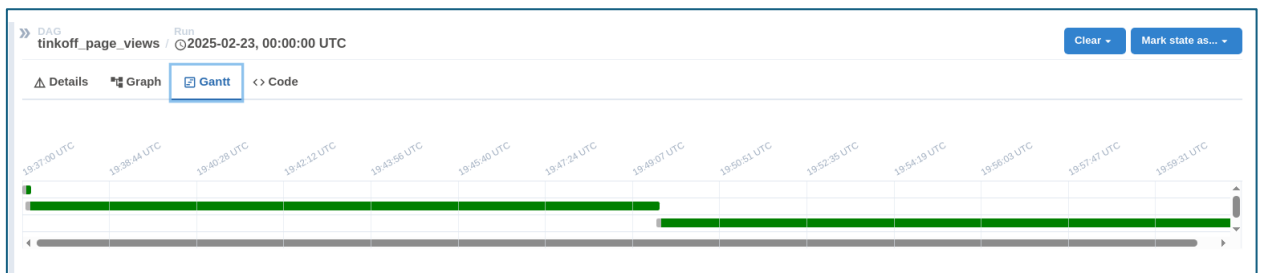


Рисунок 17

Благодаря тому, чтобы настроено логирование выполнения задач, можно было отслеживать ход выполнения (Рисунок 18).

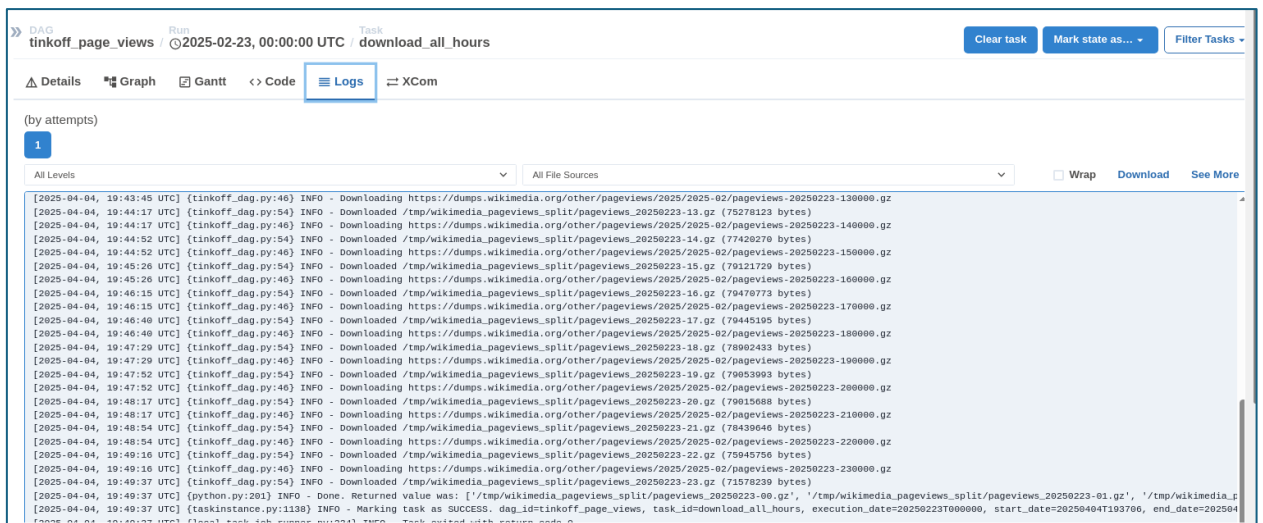


Рисунок 18

4. Проверка данных в СУБД

Перейдём в базу данных и выполним запрос на просмотре всех данных (Рисунок 19).

<input type="button" value="id"/>	<input type="button" value="page name"/>	<input type="button" value="views count"/>	<input type="button" value="view date"/>	<input type="button" value="language code"/>	<input checked="" type="checkbox"/> is mobile	<input type="button" value="created at"/>
1	T-Bank	66	2025-02-23	en	[]	2025-04-04 20:01:14.668
2	T-Bank	70	2025-02-23	en	[v]	2025-04-04 20:01:14.668
3	Tinkoff Bank	27	2025-02-23	en	[v]	2025-04-04 20:01:14.668
4	Т-Банк	484	2025-02-23	ru	[]	2025-04-04 20:01:14.668
5	Тинькофф	78	2025-02-23	ru	[]	2025-04-04 20:01:14.668
6	Т-Банк	988	2025-02-23	ru	[v]	2025-04-04 20:01:14.668
7	Тинькофф банк	25	2025-02-23	ru	[v]	2025-04-04 20:01:14.668
8	Tinkoff Bank	6	2025-02-23	en	[]	2025-04-04 20:01:14.668
9	Тинькофф	31	2025-02-23	ru	[v]	2025-04-04 20:01:14.668
10	Тинькофф банк	14	2025-02-23	ru	[]	2025-04-04 20:01:14.668
11	Tinkoff	2	2025-02-23	de	[v]	2025-04-04 20:01:14.668
12	Тинькофф Банк	2	2025-02-23	ru	[]	2025-04-04 20:01:14.668
13	Tinkoff	2	2025-02-23	de	[]	2025-04-04 20:01:14.668
14	Tinkoff	2	2025-02-23	en	[]	2025-04-04 20:01:14.668
15	Tinkoff	2	2025-02-23	en	[v]	2025-04-04 20:01:14.668
16	Tinkoff	1	2025-02-23	fr	[]	2025-04-04 20:01:14.668

Рисунок 19

Были загружены данные по количеству просмотров страниц в википедии на разных доменах, также указан атрибут «is_mobile», который отражает с какой версии просматривали страницу.

ERD-диаграмма таблицы данных (Рисунок 20).

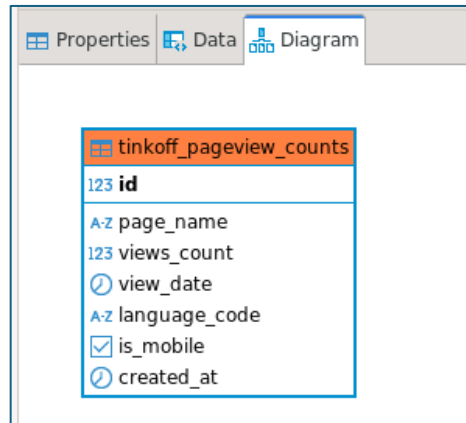


Рисунок 20

5. Запрос на подсчёт количества просмотров страниц с группировкой по наименованию (Рисунок 21).

```

    select page_name ,SUM(distinct views_count ) as total_view
    from tinkoff_pageview_counts tpc
    group by page_name ;
    
```

A-Z page name	123 total view
T-Bank	136
Tinkoff	3
Tinkoff Bank	33
Т-Банк	1,472
Тинькофф	109
Тинькофф Банк	2
Тинькофф банк	39

Рисунок 21

6. Выгрузка данных для дальнейшего перенесения на HOST (Рисунок 22).

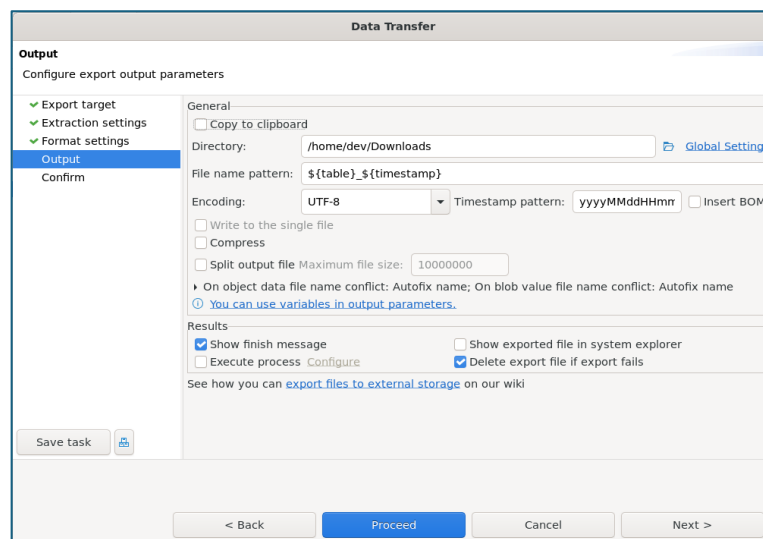


Рисунок 22

7. Визуализация в Yandex DataLens (Рисунок 23)

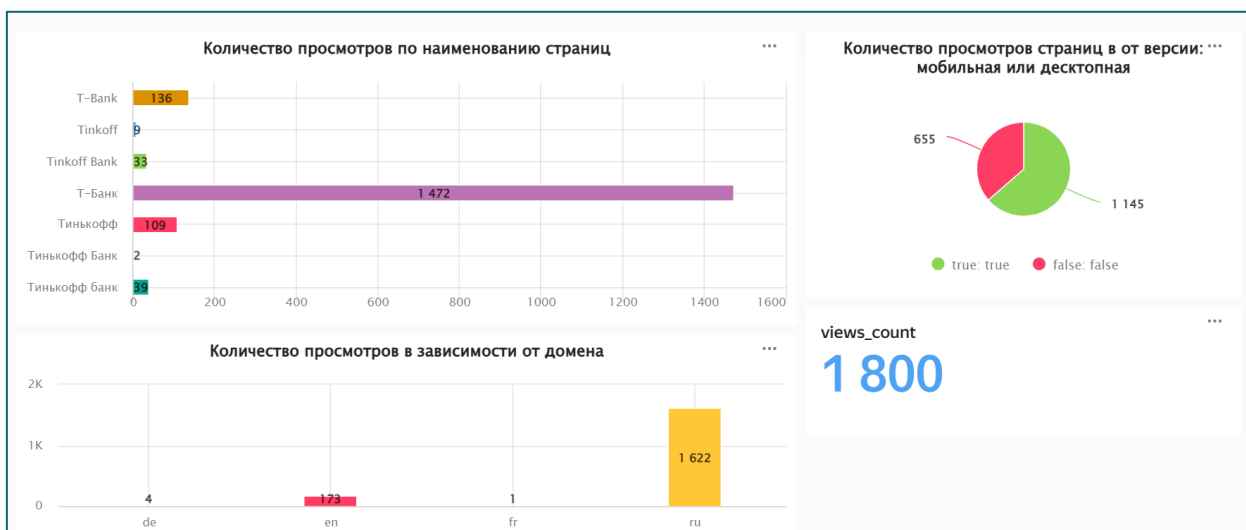


Рисунок 23

1) Большая часть просмотров сделана с мобильных устройств (1145) может указывать на:

- Обсуждение компании в соцсетях/мессенджерах
- Возможный ажиотаж среди частных инвесторов, что иногда приводит к краткосрочному росту котировок.

2) Доминирование русского языка (1622 из 1800 просмотров) и термина "Т-Банк" (1,472 просмотра) предполагает:

- Локализованное событие (новости о банке в РФ)
- Возможную связь с публикацией финансовых результатов или маркетинговой акцией.

Возможно интерес вызван маркетинговым ходом Т-банка – поздравление мужчин, в качестве подарка – акции BMW (Рисунок 24)

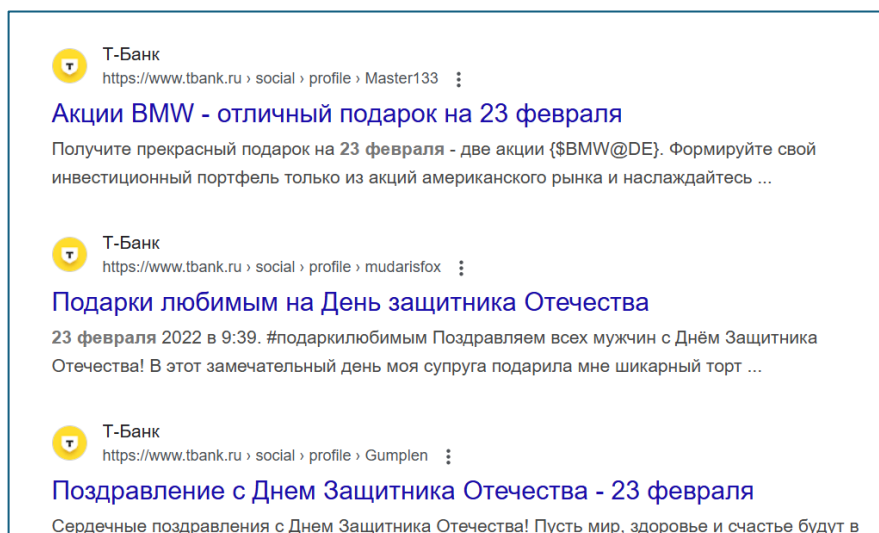


Рисунок 24

График стоимости акций Т-банка на 24 февраля 2025 года (Рисунок 25).



Рисунок 25

Вывод: Текущие данные не доказывают причинно-следственную связь и корреляцию, так как данных недостаточно, нужно брать больший отрезок времени.