

# ETL: Финал

ФИО: Курепин Андрей Дмитриевич

Группа: МИНДА241

Факультет: Инженерия данных

## Задание 1

Разработаем приложение генератор данных на Python, полный код представлен в папке генератора `get_data/*`. Запустим код и сгенерируем 1 миллион записей, Рисунки 1-2.

```
final_2 > task_1 > gen_data > main.py > ...  
1  from models import Transactions  
2  
3  if __name__ == '__main__':  
4      transaction = Transactions()  
5      transaction_data = transaction.gen_array_data(1000 * 1000, True)  
6      transaction.export_to_txt(transaction_data)  
7
```

Рисунок 1 Код для генерации данных

The screenshot shows a code editor interface. On the left is the 'EXPLORER' pane showing a file tree for 'HSE\_HW\_ETL'. The tree includes folders 'final', 'final\_2', 'task\_1', 'data', 'gen\_data', and 'sql'. Under 'task\_1' is 'Transactions\_data.csv'. Under 'gen\_data' are '\_\_pycache\_\_', '\_\_init\_\_.py', 'main.py', 'models.py', and 'requirements.txt'. The main editor pane shows the file 'Transactions\_data.csv' with the following content:

	transaction_id	user_id	amount	currency	transaction_date	is_fraud
1	1,3325	-7387	EUR	2024-11-05 03:39:22	false	
2	2,4934	-1887	USD	2024-07-24 03:39:26	true	
3	3,7366	5589	GBP	2023-05-11 12:49:10	true	
4	4,4871	-3348	RUB	2022-01-20 03:18:40	false	
5	5,3322	400	USD	2022-03-10 17:43:11	true	
6	6,7030	-2275	EUR	2024-12-07 03:47:43	true	
7	7,4518	-8516	EUR	2024-07-18 11:29:14	false	
8	8,8916	-220	JPY	2023-01-07 14:28:15	true	
9	9,8012	-5798	EUR	2023-10-07 05:20:24	true	
10	10,7116	-7574	EUR	2022-11-24 15:14:33	false	
11	11,2588	-8126	JPY	2023-07-17 21:29:52	false	
12	12,4580	-2736	USD	2022-01-13 06:06:45	true	
13	13,9932	-8101	USD	2024-09-10 05:08:56	false	
14						

Рисунок 2 Сгенерированные данные

Создадим базу данных YDB, Рисунки 3-4. SQL код приведен в папке sql/\*.

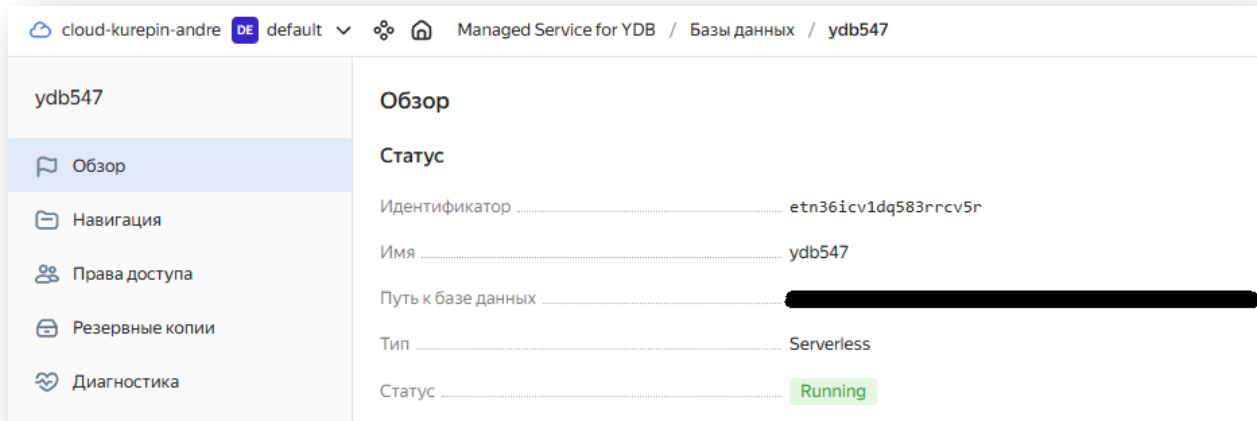
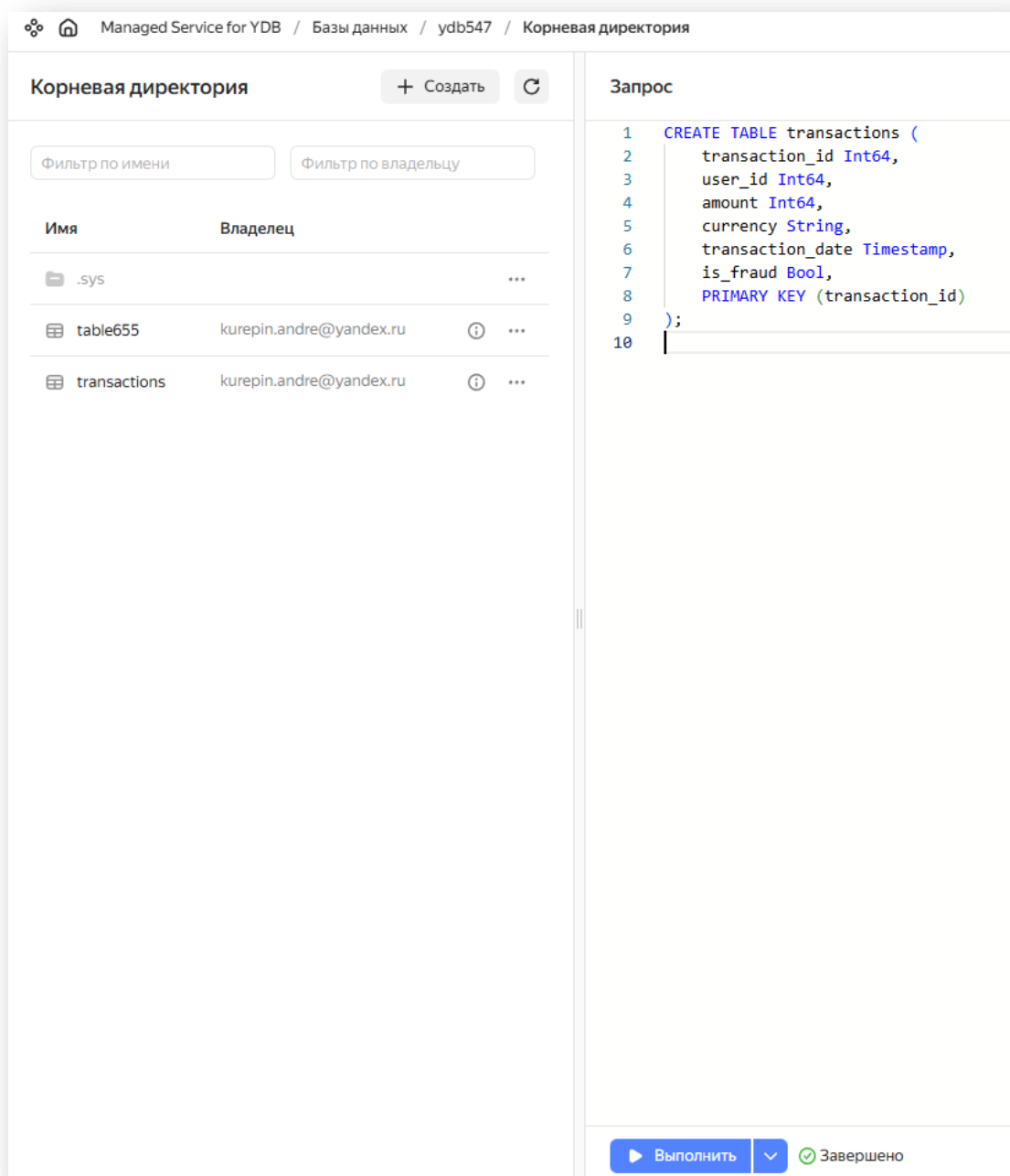


Рисунок 3 Создание БД



**Рисунок 4** Создание таблицы transactions

Выполним вставку сгенерированных данных в созданную таблицу, Рисунок 5-6. Проверим размер таблицы, Рисунок 7.

[illegible]

### Рисунок 5 Загрузка данных

[illegible]

### Рисунок 6 Успешная загрузка данных

Managed Service for YDB / Базы данных / ydb547 / Корневая директория / transactions

transactions

Строковая таблица

Показаны только первые 100 записей

#	transaction_id	user_id	amount	currency	transac
1	1	3325	-7387	"EUR"	2024-11 05T03:3
2	2	4934	-1887	"USD"	2024-07 24T03:3
3	3	7366	5589	"GBP"	2023-05 11T12:4
4	4	4871	-3348	"RUB"	2022-01 20T03:1
5	5	3322	400	"USD"	2022-03 10T17:4
6	6	7030	-2275	"EUR"	2024-12 07T03:4
7	7	4518	-8516	"EUR"	2024-07 18T11:2
8	8	8916	-220	"JPY"	2023-01 07T14:2
9	9	8012	-5798	"EUR"	2023-10 07T05:2

Запрос

1 select count(\*) from transactions;

Выполнить

Завершено

Результат #1

#	column0
1	1000000

**Рисунок 7** Размер таблицы transactions

Выполним задание, создадим трансфер из YDB в Object Storage,  
Рисунок 8. Результаты работы трансфера представлены на Рисунках 9-10.

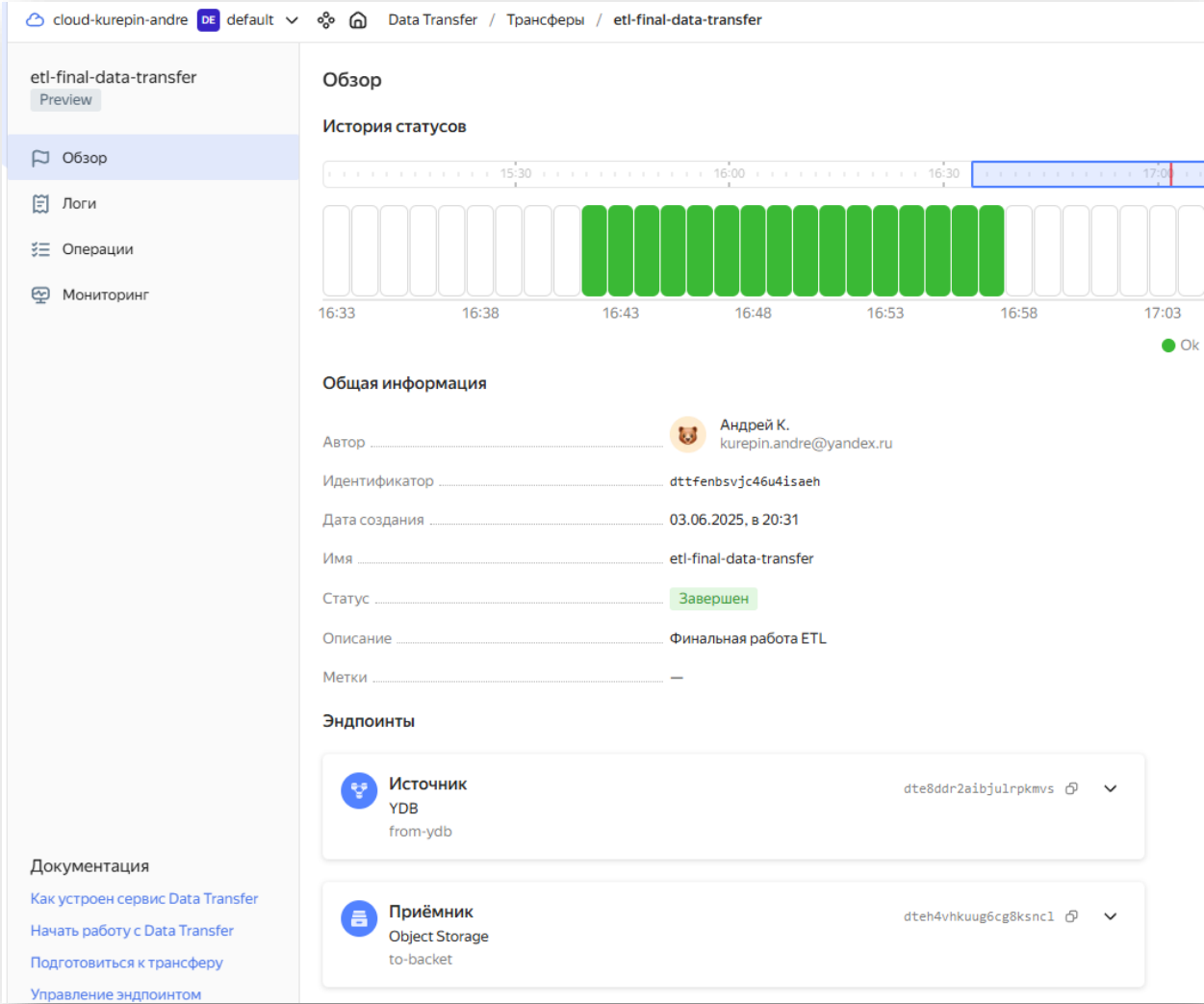


Рисунок 8 Трансфер данных

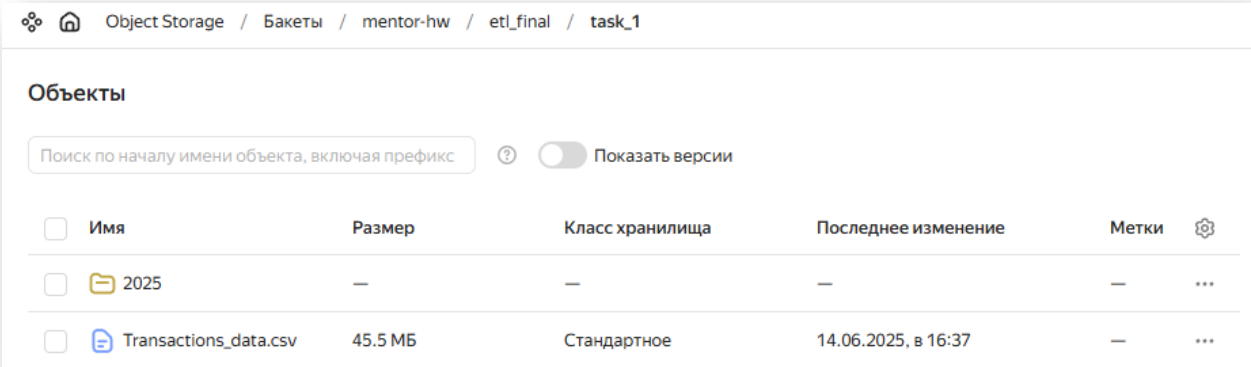
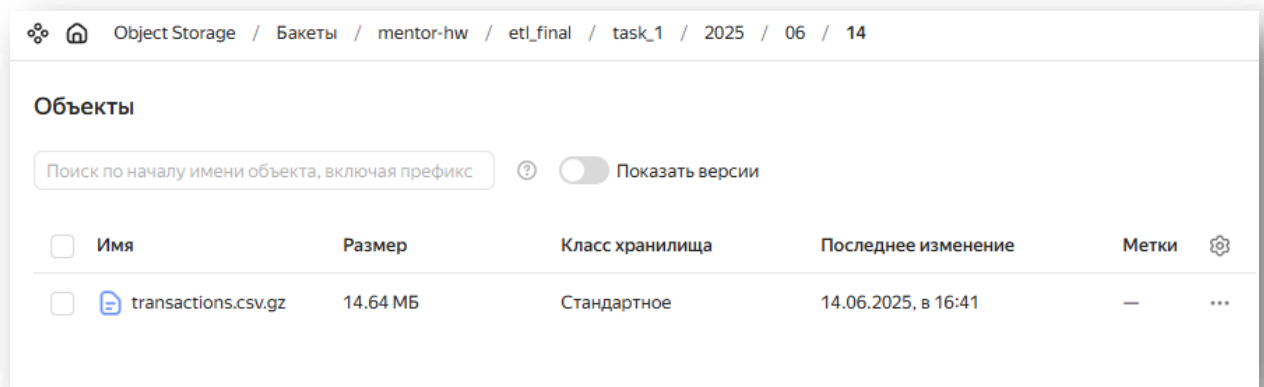


Рисунок 9 Папка с данными

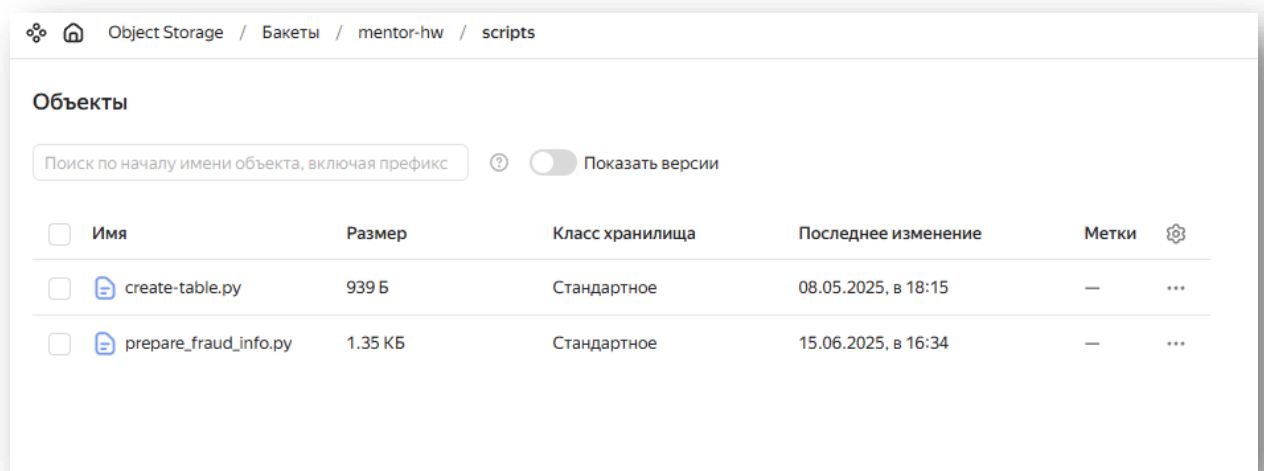


**Рисунок 10** Результирующий файл

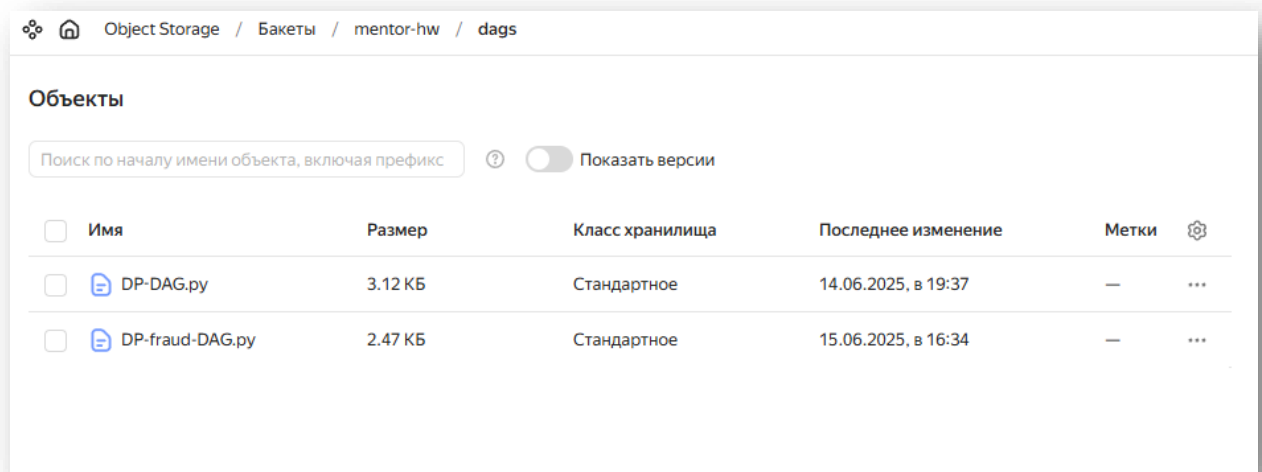
## Задание 2

Концепция – развернут Airflow, который запускает по расписанию DAG для анализа статистики по мошенническим транзакциям. На первом этапе поднимается Data Processing, на втором выполняется Spark задание, после вычислительный DP кластер удаляется. Spark задание берет файл полученный в результате выполнения первого задания.

Разработаем DAG и Spark скрипты, загрузим их в storage Рисунок 11 и 12. Полный код представлен в файлах *DP-fraud-DAG.py* и *prepare\_fraud\_info.py*.



**Рисунок 11** Spark скрипт



Object Storage / Бакеты / mentor-hw / dags

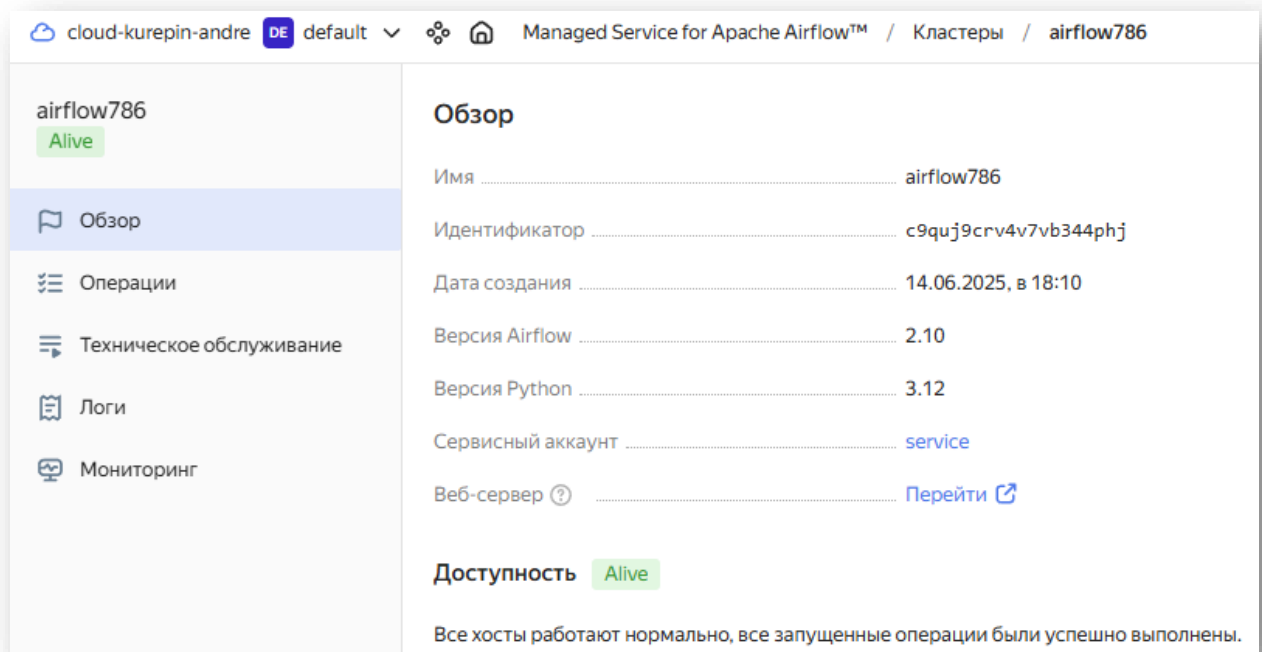
Объекты

Поиск по началу имени объекта, включая префикс ? ☐ Показать версии

<input type="checkbox"/> Имя	Размер	Класс хранилища	Последнее изменение	Метки	
<input type="checkbox"/> DP-DAG.py	3.12 КБ	Стандартное	14.06.2025, в 19:37	—	...
<input type="checkbox"/> DP-fraud-DAG.py	2.47 КБ	Стандартное	15.06.2025, в 16:34	—	...

Рисунок 12 DAG скрипт

Поднимем кластер Airflow, Рисунок 13. Откроем admin панель Airflow, в списке DAG-ов отображается my-final-hw-dag Рисунок 14.



cloud-kurepin-andre DE default Managed Service for Apache Airflow™ / Кластеры / airflow786

airflow786

Alive

Обзор

Операции

Техническое обслуживание

Логи

Мониторинг

### Обзор

Имя ..... airflow786

Идентификатор ..... c9quj9crv4v7vb344phj

Дата создания ..... 14.06.2025, в 18:10

Версия Airflow ..... 2.10

Версия Python ..... 3.12

Сервисный аккаунт ..... [service](#)

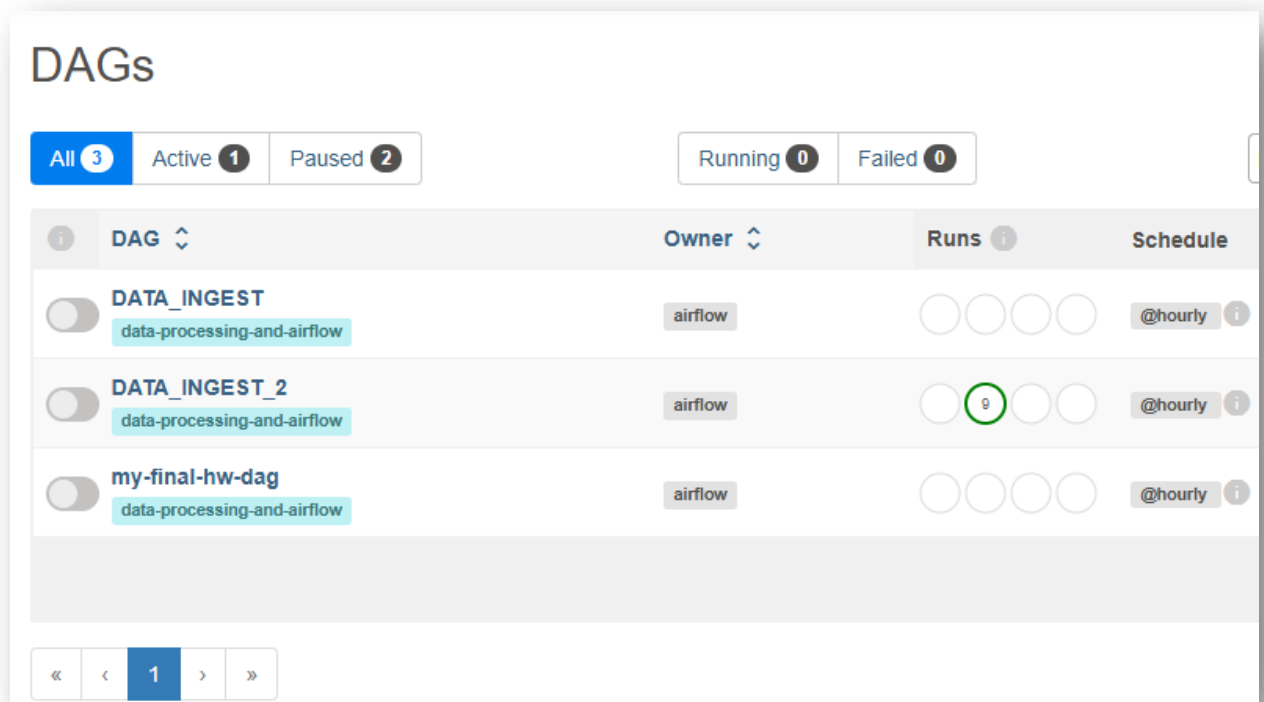
Веб-сервер ? ..... [Перейти](#)

**Доступность** Alive

Все хосты работают нормально, все запущенные операции были успешно выполнены.

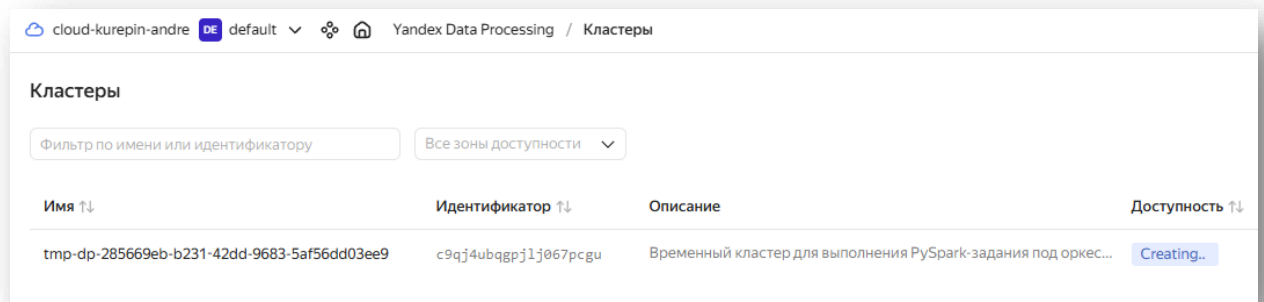
Рисунок 13 Кластер Airflow





**Рисунок 14** Веб интерфейс Airflow

Запустим данный DAG, на первом шаге будет создан вычислительный кластер, Рисунок 15. Успешное выполнение DAG, Рисунок 16. После успешного выполнения DAG сформировал файл, Рисунок 17-18.



**Рисунок 15** Создание вычислительного кластера

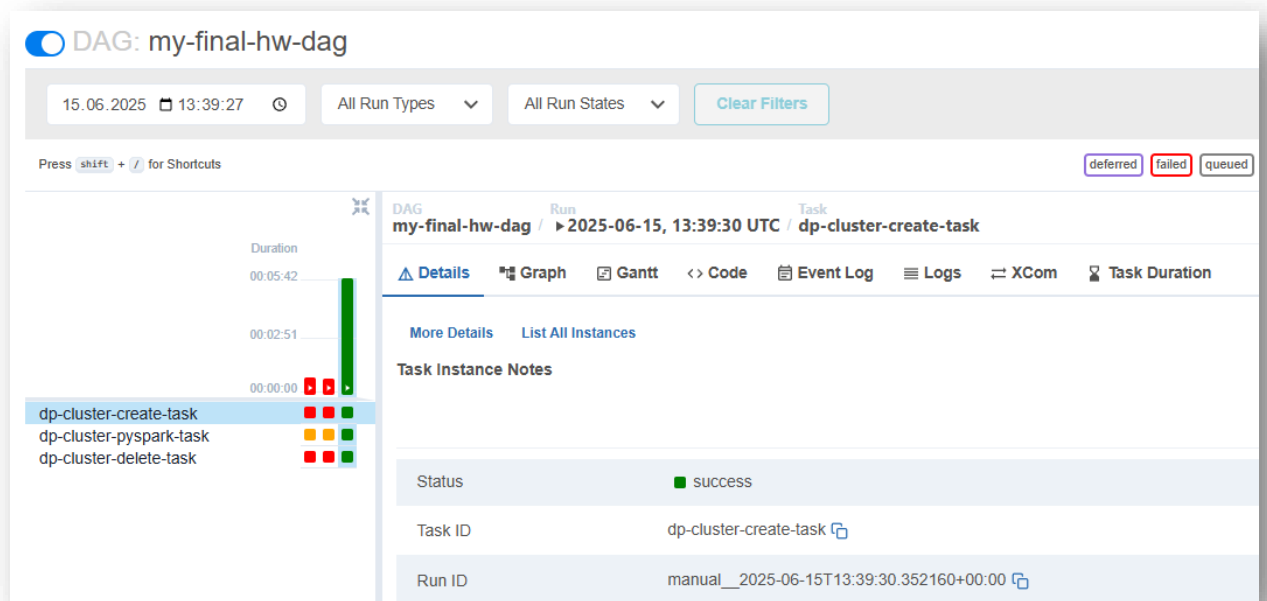


Рисунок 16 Успешное выполнение DAG-а

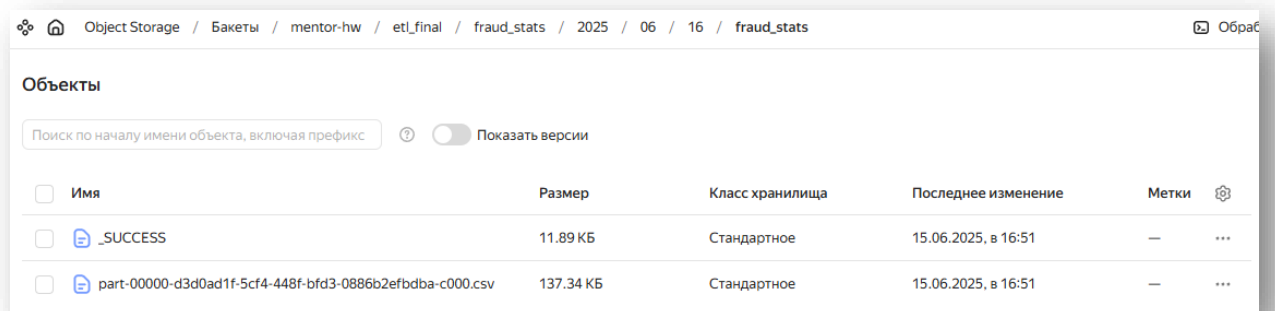


Рисунок 17 Результаты работы DAG-а

part-00000-d3d0ad1f-5cf4-448f-bfd3-0886b2efbdba-c000.csv - Excel

Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Вид Справка Что вы хотите сделать?

Вставить Шрифт Выравнивание Число Условное форматирование

Буфер обмена

Н13

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	user_id,fraud_transactions_sent,total_fraud_amount_sent													
2	2866,45,-34954													
3	1959,53,-18992													
4	833,51,-6388													
5	7554,47,-27983													
6	4935,51,105294													
7	6466,53,91805													
8	3794,45,63160													
9	463,38,33759													
10	496,50,9444													
11	7754,51,8004													
12	6357,54,46124													
13	1342,57,-20012													
14	1591,50,-3553													

Рисунок 18 Полученный файл

### Задание 3

TODO оформить

cloud-kurepin-andre default Managed Service for Kafka / Кластеры

Кластеры

Фильтр по имени или идентификатору Все зоны доступности

Имя ↑↓	Идентификатор ↑↓	Описание	Доступность ↑↓
kafka171	c9qo2fieah5u8va2ub9p	—	Alive

cloud-kurepin-andre default Yandex Data Processing / Кластеры

Кластеры

Фильтр по имени или идентификатору Все зоны доступности

Имя ↑↓	Идентификатор ↑↓	Описание	Доступность ↑↓
dataproc244	c9q2gcd6re652lgolenu	—	Alive

## Обзор 2025-06-16T18:46:45.196Z-restored-restored

Идентификатор	c9qgbht1caq2k9hq5h0p
Имя	2025-06-16T18:46:45.196Z-restored-restored
Имя кластера	<a href="#">dataproc244</a>
Статус	Done
Время выполнения	7.74 секунд
Начало	16.06.2025, в 22:10
Конец	16.06.2025, в 22:10
Пользователь	Андрей К.
Тип задания	PYSPARK

### Аргументы задания

Настройки	spark.submit.deployMode: cluster
Main python файл	s3a://mentor-hw/etl_final/task_3/kafka-writer.py

## Обзор 2025-06-15T18:35:25.607Z-restored-restored-restored

Идентификатор	c9qk1cu0a66s6i65h1oh
Имя	2025-06-15T18:35:25.607Z-restored-restored-restored
Имя кластера	<a href="#">dataproc244</a>
Статус	Done
Время выполнения	15.98 секунд
Начало	16.06.2025, в 22:14
Конец	16.06.2025, в 22:15
Пользователь	Андрей К.
Тип задания	PYSPARK

### Аргументы задания



Настройки	spark.submit.deployMode: cluster
Main python файл	s3a://mentor-hw/etl_final/task_3/kafka-read-stream.py

Объекты

Поиск по началу имени объекта, включая префикс

?

Показать версии

<input type="checkbox"/> Имя	Размер	Класс хранилища	Последнее изменение
<input type="checkbox"/>  _SUCCESS	0 Б	Стандартное	16.06.2025, в 22:14
<input type="checkbox"/>  part-00000-46c29d2b-0ff5-47c4-b2c5-f66370259c9d-c000.txt	614 Б	Стандартное	16.06.2025, в 22:14

etl\_final/task\_3/result/part-00000-46c29d2b-0ff5-47c4-b2c5-f66370259c9d-c000.txt

Последнее изменение 16.06.2025, в 22:14

Размер 614 Б

Класс хранилища Стандартное

Доступ Публичный

Метки 

Добавить метку

part-00000-46c29d2b-0ff5-47c4-b2c5-f66370259c9d-c000.txt – Блокнот

Файл Правка Формат Вид Справка

```
{ "transaction_id": 2, "user_id": 1002, "amount": 7500, "currency": "EUR", "transaction_date": "2025-06-15T12:05:00.000Z", "is_fraud": true }
{ "transaction_id": 1, "user_id": 1001, "amount": 5000, "currency": "USD", "transaction_date": "2025-06-15T12:00:00.000Z", "is_fraud": false }
```