

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное  
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет  
имени академика С.П. Королева»  
(Самарский университет)**

**Институт информатики и кибернетики  
Кафедра суперкомпьютеров и общей информатики**

**Отчет по лабораторной работе №1**

**Дисциплина: «Инженерия данных»**

**Тема: «Базовый пайплайн работы с данными»**

**Выполнил: Неженский М.С.**

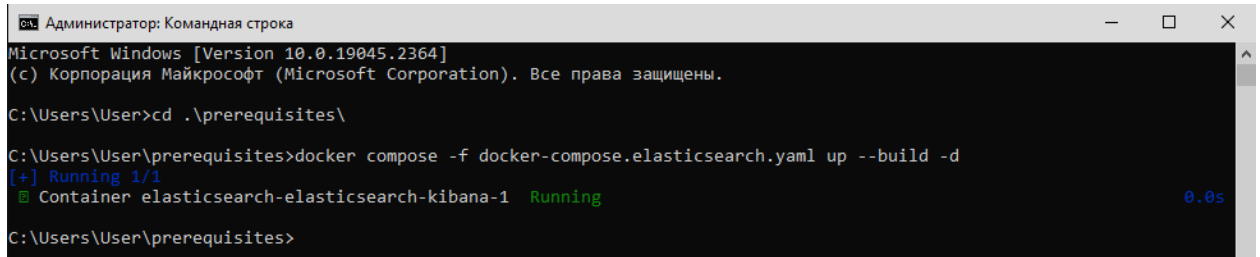
**Группа: 6233-010402D**

**Самара 2023**

## 1. Подготовка и проблемы

Все, команды необходимые для скачивания контейнеров брал тут:  
<https://github.com/ssau-data-engineering/Prerequisites>.

Установка Apache Airflow, Apache NiFi, ElasticSearch, MLflow, PostgreSQL:



```
Администратор: Командная строка
Microsoft Windows [Version 10.0.19045.2364]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\User>cd .\prerequisites\

C:\Users\User\prerequisites>docker compose -f docker-compose.elasticsearch.yaml up --build -d
[+] Running 1/1
  Container elasticsearch-elasticsearch-kibana-1 Running 0.0s

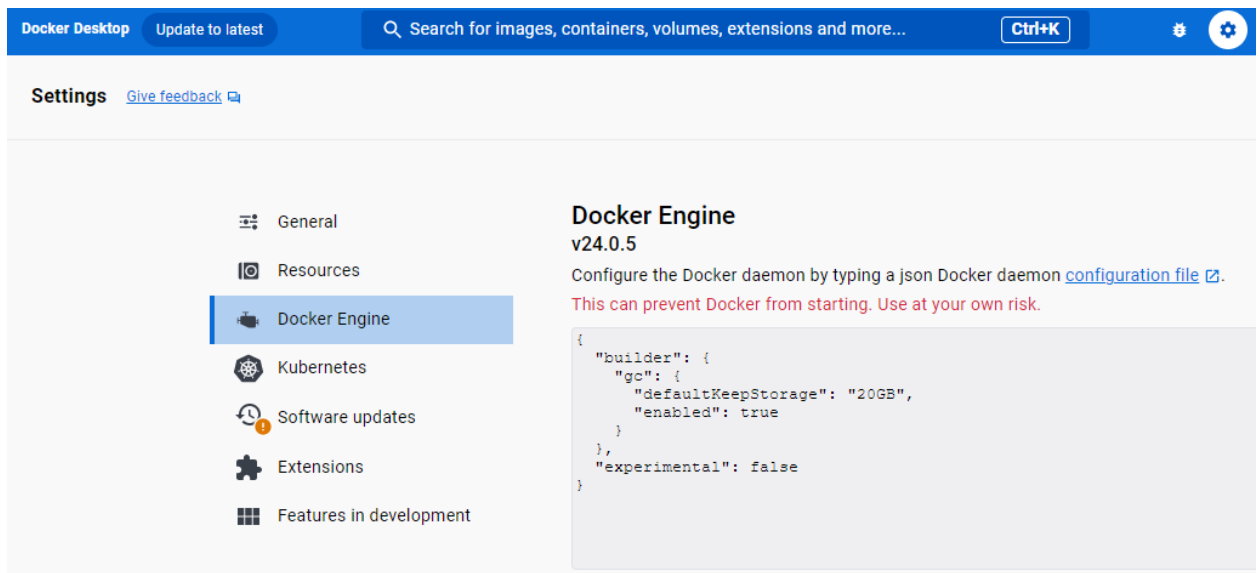
C:\Users\User\prerequisites>
```

В ходе подготовки был установлен [vscode](#), со следующими расширениями:

- [ms-python.python](#)
- [ms-toolsai.jupyter](#)
- [ms-vscode-remote.vscode-remote-extensionpack](#)
- [ms-azuretools.vscode-docker](#)

Для быстрой настройки среды воспользовался следующим профилем: [Python Jupyter Remote Docker.code-profile](#).

Проблем была масса с Docker Desktop. Первая проблема была в медленном интернете, хотя скачивал через роутер. Решение – две строчки прописать на следующем рисунке в Docker Engine. Что-то про ограничение пакетов передачи данных, либо найти лучше интернет. При ограничении пакетов скорость становится еще ниже, вплоть до 2 часов загрузки всех 5 контейнеров. Прописывать строчки надо на одном уровне с "experimental": false через запятую.



Вторая проблема – оперативная память, если память как у меня – 8ГБ, то ограничение сразу надо ставить в 4ГБ, так как я поставил 6ГБ и когда запускал контейнеры, все настолько висло, что даже чат в телеграмме становился черным. Если нажать на приложение докера, то он зависнет и предложит завершить процесс, если его завершить, то все скачанные тобой контейнеры за два часа времени исчезнут. Скачивание начинается заново. После настройки ограничения вот такие показатели работы ПК. ЦП и Память загружена на 95-99%, но хотя бы люто не лагает и Windows всегда отвечает. Был создан файл C:/Users/%Имя пользователя%/.wslconfig со следующим содержимым:

```
[wsl2]
memory=4GB
processors=4
```

Включать контейнеры надо всегда отдельно если конечно они не необходимы для совместной работы, как например: 1) Apache NiFi и ElasticSearch, 2) Apache Airflow и ElasticSearch в данной работе.

Третья проблема – это работа контейнеров. Решения я так и не нашел. В NiFi удалось зайти пару раз и еще три раза в Airflow. Сидишь и ждешь как у моря погоды и не знаешь заработают ли контейнеры по ссылке в этот раз или нет. В докере все отображается отлично. Я переустанавливал

контейнеры, после удаления, перезапускал систему, выключал и запускал снова, через консоль, через докер и внутри каждого контейнера перезапускал части. Бывает что сел делать лабораторную, а в итоге целый день просидел за запуском контейнера. А потом попробовал еще раз и заработала. С данной аномалией я так и не разобрался.

Добавленные контейнеры:

Containers

Images

Volumes

Dev Environments BETA

Docker Scout EARLY ACCESS

Learning center

Extensions

+

Add Extensions

Containers [Give feedback](#)

Container CPU usage ⓘ  
No containers are running.

Container memory usage ⓘ  
No containers are running.

Show charts ▾

Search

☐

Only show running containers

<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	> airflow		Exited	N/A		59 minutes ago	<div>▶ ⋮ ☒</div>
<input type="checkbox"/>	> mlflow		Exited	N/A		1 day ago	<div>▶ ⋮ ☒</div>
<input type="checkbox"/>	> postgresql		Exited	N/A		1 day ago	<div>▶ ⋮ ☒</div>
<input type="checkbox"/>	> elasticsearch		Exited	N/A		1 hour ago	<div>▶ ⋮ ☒</div>
<input type="checkbox"/>	> nifi		Exited	N/A		24 hours ago	<div>▶ ⋮ ☒</div>

Showing 5 items

RAM 1.26 GB CPU 15.95% Connected to Hub

v4.22.0

## 2. Apache NiFi

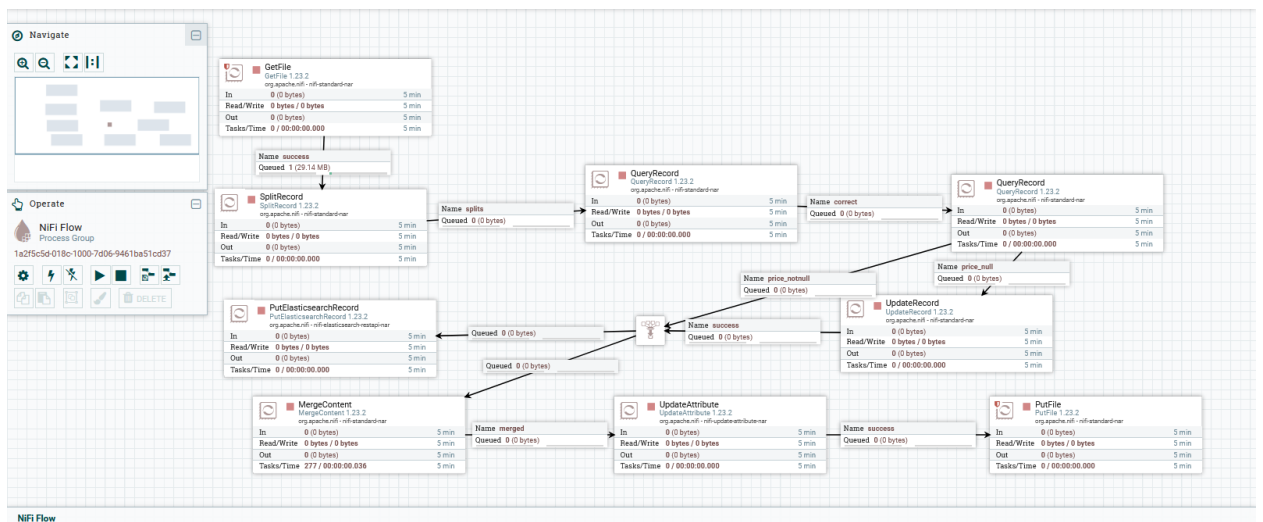
Знакомство с основными инструментами построения пайплайнов началось с Apache NiFi - простой в использовании, мощной и надежной система для обработки и распространения данных.

CSV файлы были перенесены в папку, к которой **NiFi** имеет доступ (в моем случае в папку `nifi/data/input`).

Для реализации пайплайна воспользовался следующими процессами:

- **GetFile** - создает FlowFiles из файлов в каталоге. NiFi будет игнорировать файлы, для которых у него нет хотя бы прав на чтение.
- **SplitRecord** - разбивает входной FlowFile в формате данных, ориентированном на записи, на несколько меньших FlowFile.
- **QueryRecord** - оценивает один или несколько запросов SQL по содержимому FlowFile. Первый QueryRecord нужен для того, чтобы поля `designation` и `region_1` не были пустыми, второй для того, чтобы разделить поля, где `price` принимает значение `null` и не `null`.
- **UpdateRecord** - обновляет содержимое FlowFile, содержащего данные, ориентированные на запись (т. е. данные, которые можно прочитать через `RecordReader` и записать с помощью `RecordWriter`).
- **PutElasticsearchHttpRecord** - записывает записи из FlowFile в Elasticsearch, используя указанные параметры, такие как индекс для вставки и тип документа, а также тип операции (индекс, обновление, удаление и т. д.).
- **MergeContent** - объединяет группу FlowFile на основе определяемой пользователем стратегии и упаковывает их в один FlowFile.
- **UpdateAttribute** - этот процессор обновляет атрибуты FlowFile, используя свойства или правила, добавленные пользователем, то есть переименовывает файл в нашем случае.
- **PutFile** - записывает содержимое FlowFile в локальную файловую систему.

## Итоговая схема построенная в Apache NiFi:



Далее приведены скриншоты для настройки данной схемы, приведены как настройки, так и отношения. В GetFile прописан путь загрузки входных данных и фильтр для используемых файлов.

Configure Processor | GetFile 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

success

☐ terminate ☐ retry

All files are routed to success

CANCEL

APPLY

Configure Processor | GetFile 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Input Directory	/opt/nifi/nifi-current/data
File Filter	[^\\].*
Path Filter	No value set
Batch Size	25
Keep Source File	false
Recurse Subdirectories	true
Polling Interval	0 sec
Ignore Hidden Files	true
Minimum File Age	0 sec
Maximum File Age	No value set
Minimum File Size	0 B
Maximum File Size	No value set

CANCEL

APPLY

Для чтения и записи используется CSVReader и CSVRecordSetWriter, после данных настроек следует нажать на стрелочку около CSVReader.

Configure Processor | SplitRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Record Reader	CSVReader
Record Writer	CSVRecordSetWriter
Records Per Split	100000

CANCEL

APPLY

Configure Processor | SplitRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships

failure

☒ terminate
 ☐ retry

If a FlowFile cannot be transformed from the configured input format to the configured output format, the unchanged FlowFile will be routed to this relationship.

original

☒ terminate
 ☐ retry

Upon successfully splitting an input FlowFile, the original FlowFile will be sent to this relationship.

splits

☐ terminate
 ☐ retry

The individual 'segments' of the original FlowFile will be routed to this relationship.

CANCEL

APPLY

В появившемся окне стоит нажать на молнию и в появившемся окне нажать кнопку Enable для включения данных функций в программу, чтобы получилось вот так:



GENERAL		CONTROLLER SERVICES			
Name	Type	Bundle	State	Scope	
CSVReader	CSVReader 1.23.2	org.apache.nifi - nifi-record-serialization-services-nar	Enabled	NiFi Flow	
CSVReader	CSVReader 1.23.2	org.apache.nifi - nifi-record-serialization-services-nar	Enabled	NiFi Flow	
CSVRecordSetWriter	CSVRecordSetWriter 1.23.2	org.apache.nifi - nifi-record-serialization-services-nar	Enabled	NiFi Flow	
ElasticSearchClientServiceImpl	ElasticSearchClientServiceImpl 1.23.2	org.apache.nifi - nifi-elasticsearch-client-service-nar	Enabled	NiFi Flow	

Last updated: 21:56:09 UTC

Listed services are available to all descendant Processors and services of this Process Group.

Конечно CSVReader должен быть один, но я забыл его удалить.

В QueryRecord мы нажимаем на плюсики и создаем новую строчку с названием correct и SQL-запросом в значении.

### Configure Processor | QueryRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Record Reader	CSVReader
Record Writer	CSVRecordSetWriter
Include Zero Record FlowFiles	false
Cache Schema	true
Default Decimal Precision	10
Default Decimal Scale	0
correct	SELECT * FROM FLOWFILE WHERE designation <> " AND...

CANCEL

APPLY

## Configure Processor | QueryRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

correct

☐ terminate ☐ retry

User-defined relationship that specifies where data that matches the specified SQL query should be routed

failure

☒ terminate ☐ retry

If a FlowFile fails processing for any reason (for example, the SQL statement contains columns not present in input data), the original FlowFile it will be routed to this relationship

original

☒ terminate ☐ retry

The original FlowFile is routed to this relationship

CANCEL

APPLY

## Configure Processor | QueryRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field



Property	Value	
Record Reader	<input type="text" value="CSVReader"/>	→
Record Writer	<input type="text" value="CSVRecordSetWriter"/>	→
Include Zero Record FlowFiles	<input type="text" value="false"/>	
Cache Schema	<input type="text" value="true"/>	
Default Decimal Precision	<input type="text" value="10"/>	
Default Decimal Scale	<input type="text" value="0"/>	
price_notnull	<input type="text" value="SELECT * FROM FLOWFILE WHERE price is not null"/>	🗑️
price_null	<input type="text" value="SELECT * FROM FLOWFILE WHERE price is null"/>	🗑️

CANCEL

APPLY

## Configure Processor | QueryRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

failure

☒ terminate ☐ retry

If a FlowFile fails processing for any reason (for example, the SQL statement contains columns not present in input data), the original FlowFile it will be routed to this relationship

original

☒ terminate ☐ retry

The original FlowFile is routed to this relationship

price\_notnull

☐ terminate ☐ retry

User-defined relationship that specifies where data that matches the specified SQL query should be routed

price\_null

☐ terminate ☐ retry

User-defined relationship that specifies where data that matches the specified SQL query should be routed

CANCEL

APPLY

## Configure Processor | UpdateRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field



Property	Value	
Record Reader	<input type="text" value="CSVReader"/>	→
Record Writer	<input type="text" value="CSVRecordSetWriter"/>	→
Replacement Value Strategy	<input type="text" value="Literal Value"/>	
/price	<input type="text" value="0.0"/>	🗑️

CANCEL

APPLY

Configure Processor | UpdateRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

failure

☒ terminate ☐ retry

If a FlowFile cannot be transformed from the configured input format to the configured output format, the unchanged FlowFile will be routed to this relationship

success

☐ terminate ☐ retry

FlowFiles that are successfully transformed will be routed to this relationship

CANCEL

APPLY

Configure Processor | PutElasticsearchRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field 

?

+

Property	Value
Index Operation	<div>?</div> index
Index	<div>?</div> nifi
Type	<div>?</div> _doc
@timestamp Value	<div>?</div> No value set
Client Service	<div>?</div> ElasticSearchClientServiceImpl →
Record Reader	<div>?</div> CSVReader →
Batch Size	<div>?</div> 100
ID Record Path	<div>?</div> No value set
Index Operation Record Path	<div>?</div> No value set
Index Record Path	<div>?</div> No value set
Type Record Path	<div>?</div> No value set
@timestamp Record Path	<div>?</div> No value set

CANCEL

APPLY

## Configure Processor | PutElasticsearchRecord 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

### errors

☒ terminate ☐ retry

If a "Result Record Writer" is set, any Record(s) corresponding to Elasticsearch document(s) that resulted in an "error" (within Elasticsearch) will be routed here.

### failure

☒ terminate ☐ retry

All flowfiles that fail for reasons unrelated to server availability go to this relationship.

### retry

☒ terminate ☐ retry

All flowfiles that fail due to server/cluster availability go to this relationship.

### success

☒ terminate ☐ retry

All flowfiles that succeed in being transferred into Elasticsearch as new Documents assigned by the

CANCEL

APPLY

## Configure Processor | MergeContent 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

### failure

☒ terminate ☐ retry

If the bundle cannot be created, all FlowFiles that would have been used to create the bundle will be transferred to failure

### merged

☐ terminate ☐ retry

The FlowFile containing the merged content

### original

☒ terminate ☐ retry

The FlowFiles that were used to create the bundle

CANCEL

APPLY

## Configure Processor | MergeContent 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field



Property		Value	
Merge Format	?	Binary Concatenation	
Attribute Strategy	?	Keep Only Common Attributes	
Correlation Attribute Name	?	No value set	
Minimum Number of Entries	?	1	
Maximum Number of Entries	?	1000	
Minimum Group Size	?	0 B	
Maximum Group Size	?	No value set	
Max Bin Age	?	No value set	
Maximum number of Bins	?	5	
Delimiter Strategy	?	Text	
Header	?	id,country,designation,points,price,province,region_1,reg...	
Footer	?	No value set	
Demarcator	?	No value set	

CANCEL

APPLY

## Configure Processor | UpdateAttribute 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field



Property		Value	
Delete Attributes Expression	?	No value set	
Store State	?	Do not store state	
Stateful Variables Initial Value	?	No value set	
Cache Value Lookup Cache Size	?	100	
filename	?	final.csv	

ADVANCED

CANCEL

APPLY

SETTINGS | SCHEDULING | PROPERTIES | RELATIONSHIPS | COMMENTS

Automatically Terminate / Retry Relationships ?

success

☐ terminate ☐ retry

All successful FlowFiles are routed to this relationship

SETTINGS | SCHEDULING | PROPERTIES | RELATIONSHIPS | COMMENTS

Required field

Property	Value
Directory	/opt/nifi/nifi-current/data
Conflict Resolution Strategy	fail
Create Missing Directories	true
Maximum File Count	No value set
Last Modified Time	No value set
Permissions	No value set
Owner	No value set
Group	No value set

Configure Processor | PutFile 1.23.2

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Automatically Terminate / Retry Relationships ?

failure

☒ terminate
 ☐ retry

Files that could not be written to the output directory for some reason are transferred to this relationship

success

☒ terminate
 ☐ retry

Files that have been successfully written to the output directory are transferred to this relationship

CANCEL

APPLY

Появившийся итоговый файл в нашей дирректории.

<< Пользователи > User > prerequisites > nifi > data					Поиск в: data
Доступ	Имя	Дата изменения	Тип	Размер	
Пол	final.csv	30.11.2023 1:25	Файл Microsoft O...	1 103 КБ	
ы					
ния					
Приключени					
й					
ения					
ии ИИ					

Сама схема будет лежать в формате .xml в моем репозитории.

После заходим в Elasticsearch и проверяем, что данные обработаны верно. Был создан Index с полями внутри.



elastic

Search Elastic

Stack Management

Index Management

Management

Ingest

Data

Alerts and Insights

Kibana

Stack

Ingest Pipelines

Index Lifecycle Policies

Snapshot and Restore

Rollup Jobs

Transforms

Remote Clusters

Rules and Connectors

Reporting

Machine Learning Jobs

Index Patterns

Saved Objects

Tags

Search Sessions

Spaces

Advanced Settings

Index Management

Index Management docs

Indices

Data Streams

Index Templates

Component Templates

Update your Elasticsearch indices individually or in bulk. [Learn more.](#)

Include rollout indices

Include hidden indices

Search

Reload indices

Name	Health	Status	Primaries	Replicas	Docs count	Storage size	Data stream
nifi	yellow	open	1	1	75036	49.7mb	

Rows per page: 10

< 1 >

elastic

Search Elastic

Stack Management

Index patterns

nifi\*

Ingest Pipelines

Data

Alerts and Insights

Kibana

Stack

Ingest Pipelines

Index Management

Index Lifecycle Policies

Snapshot and Restore

Rollup Jobs

Transforms

Remote Clusters

Rules and Connectors

Reporting

Machine Learning Jobs

Index Patterns

Saved Objects

Tags

Search Sessions

Spaces

Advanced Settings

License Management

Upgrade Assistant

nifi\*

Default

View and edit fields in nifi\*. Field attributes, such as type and searchability, are based on [field mappings](#) in Elasticsearch.

Fields (30)

Scripted fields (0)

Field filters (0)

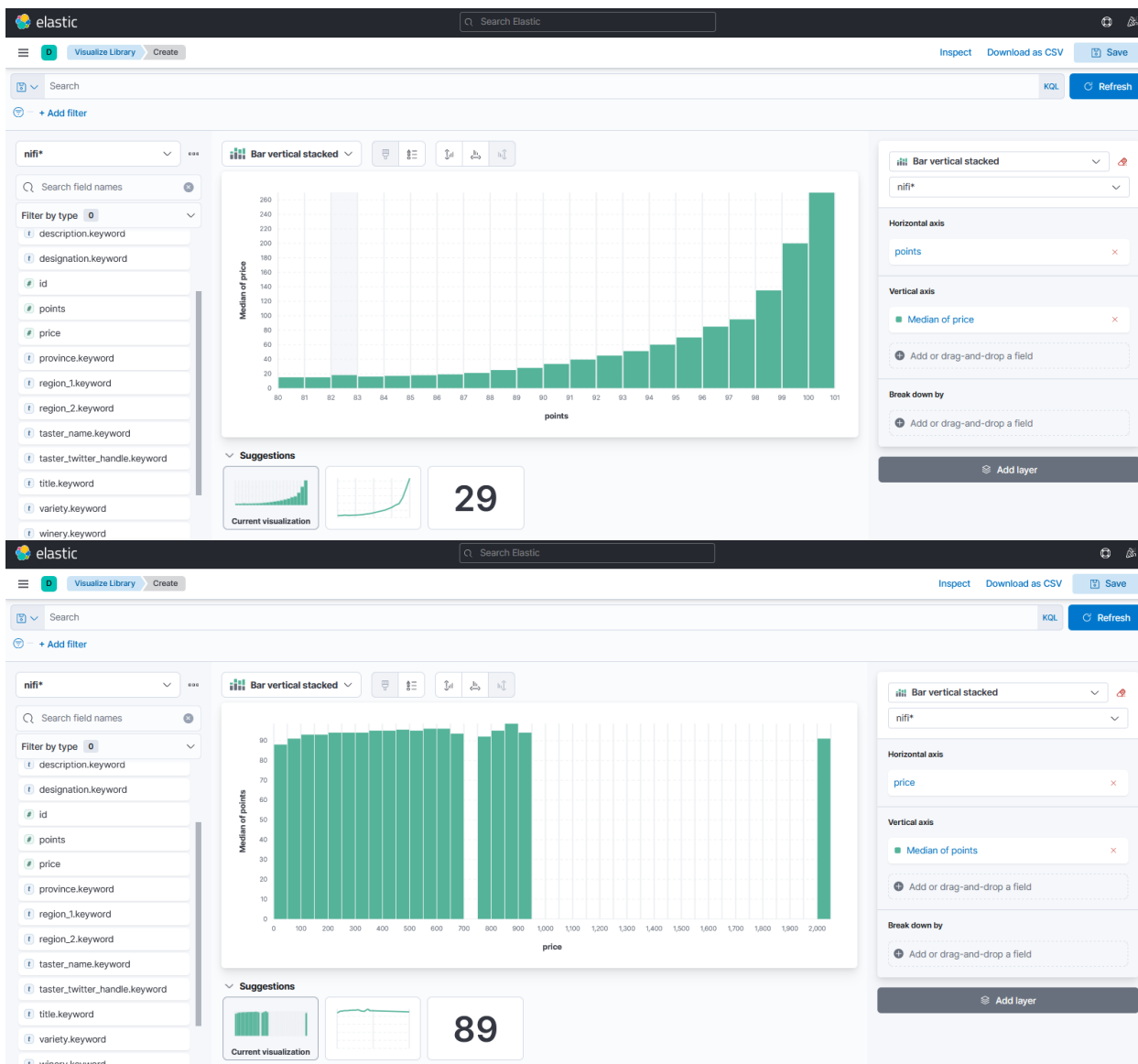
Search

All field types

Add field

Name	Type	Format	Searchable	Aggregatable	Excluded
_id	_id				
_index	_index				
_score					
_source	_source				
_type	_type				
country	text				
country.keyword	keyword				
description	text				
description.keyword	keyword				
designation	text				

Построенны несколько графиков с использованными данными.



### 3. Apache Airflow

Apache Airflow – это платформа для создания пайплайнов. Для генерации которых надо написать код и запустить его. Код представлен далее и будет лежать в моем репозитории. В коде представлены функции для предварительной предобработки данных и загрузки в Elasticsearch. CSV файлы были перенесены в папку, к которой **Airflow** имеет доступ, аналогично Nifi. Ниже представлен код Dag и структура содержимого в папке Airflow. Данный код перенесен в папку dags как можно видеть, для того, чтобы в интерфейсе Airflow он был виден.

The image shows a VS Code editor window with a file explorer on the left and a code editor on the right. The file explorer shows a project structure for 'AIRFLOW' with folders 'config', 'dags', and 'data'. The 'dags' folder contains 'pipeline.py' (8 lines). The 'data' folder contains 25 CSV files named 'chunk0.csv' through 'chunk24.csv'. The code editor shows the content of 'pipeline.py'.

```
1 from airflow import DAG
2 from airflow.operators.python import PythonOperator
3 from elasticsearch import Elasticsearch
4 from datetime import datetime
5 import pandas as pd
6 import numpy as np
7 from elasticsearch import Elasticsearch
8 from elasticsearch.helpers import bulk
9
10 with DAG('DAG_Airflow',
11         schedule_interval=None,
12         start_date=datetime(2023, 11, 30),
13         catchup=False) as dag:
14
15     def read_data():
16         result = pd.DataFrame()
17         for i in range(26):
18             result = pd.concat([result, pd.read_csv(f"/opt/airflow/data/chunk{i}.csv")])
19         return result
20
21     def transform_data():
22         result = read_data()
23         result = result[result['region_1'].notnull()]
24         result = result[result['designation'].notnull()]
25         result['price'] = result['price'].replace(np.nan, 0)
26         result = result.drop(['id'], axis=1)
27         result.to_csv('/opt/airflow/data/data.csv', index=False)
```

```

33     def load_to_elastic():
34         patch = Elasticsearch("http://elasticsearch-kibana:9200")
35         data = pd.read_csv(f"/opt/airflow/data/data.csv")
36
37         for i, row in data.iterrows():
38             doc = {
39                 "_index": 'lab1',
40                 "_type": "_doc",
41                 "_source": data,
42             }
43
44             if i < data.shape[0] - 1:
45                 patch.index(index="wines", id=i, document=doc)
46
47         transform_data = PythonOperator(
48             task_id='transform_data',
49             python_callable=transform_data,
50             dag=dag)
51
52         load_to_elastic = PythonOperator(
53             task_id='load_to_elastic',
54             python_callable=load_to_elastic,
55             dag=dag)
56
57         transform_data >> load_to_elastic

```

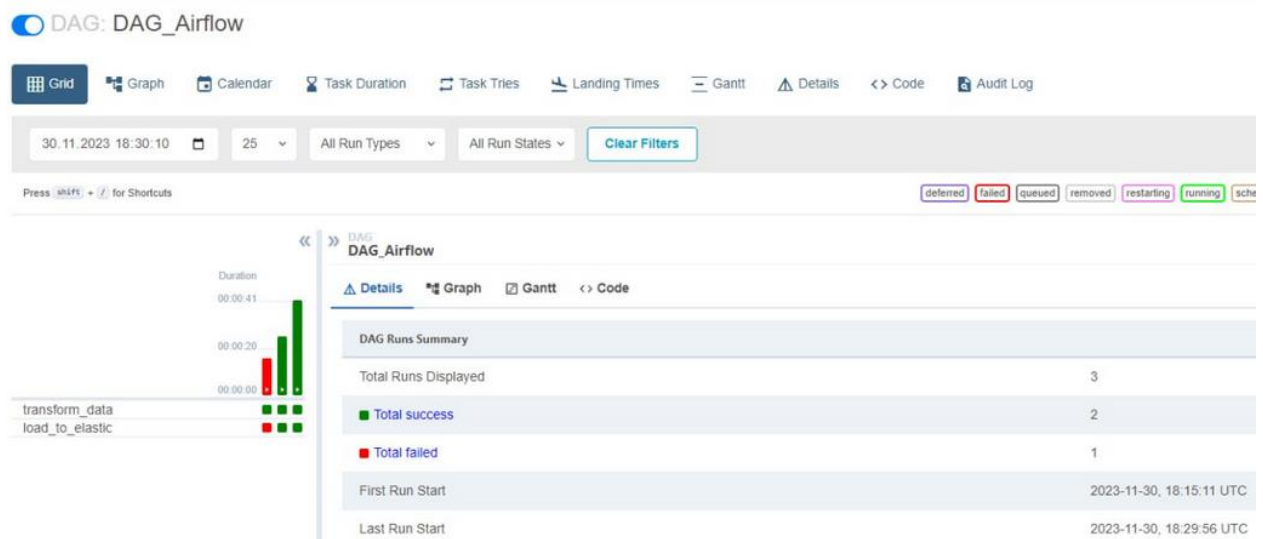
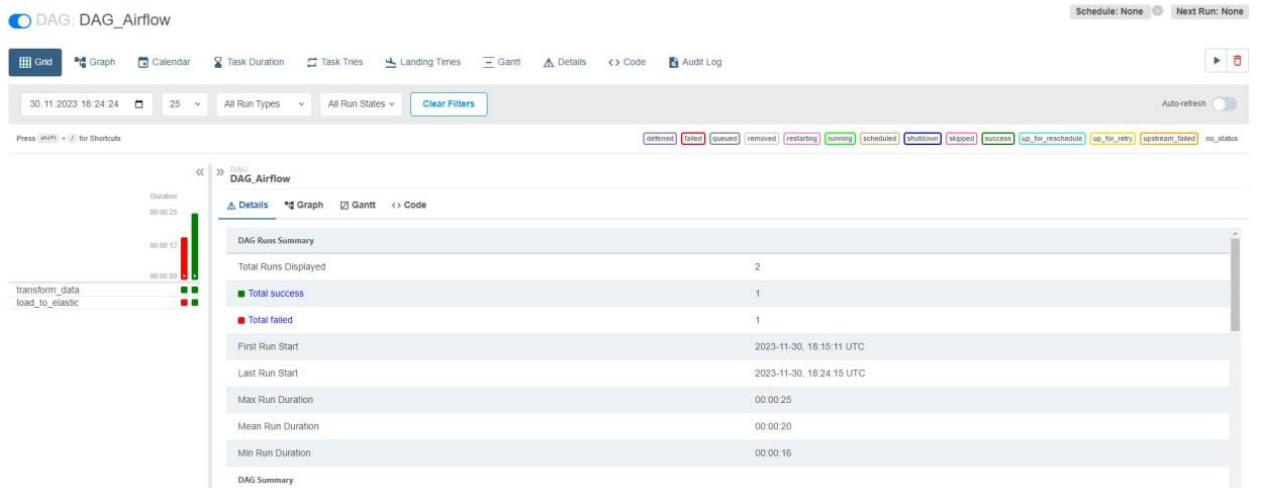
Интерфейс Airflow с загруженным Dag с названием DAG\_Airflow.

## DAGs

All 57
Active 1
Paused 56
Running 0
Failed 0

DAG ↕	Owner ↕
<input checked="" type="checkbox"/> DAG_Airflow	airflow
<input type="checkbox"/> dataset_consumes_1 consumes dataset-scheduled	airflow

## Запущенный Dag:



## ЗАКЛЮЧЕНИЕ

В результате выполнения лабораторной работы были получены навыки по работе с Docker, Apache Airflow, Apache NiFi, ElasticSearch. Была создана схема пайплайна в Apache NiFi, код для генерации данных в Apache Airflow и использован ElasticSearch для визуализации данных.