

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

Программирование на Python

Отчет по лабораторной работе

Тема: «Основы работы с DuckDB»

Выполнил студент группы ИВТ-б-о-21-1

Кочкаров Умар Ахматович

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2023


Цель работы: исследовать базовые возможности системы управления базами данных DuckDB.

Ход работы:

1. Создать общедоступный репозиторий с лицензией MIT и языком Python.

Create a new repository



A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *  umarkochkarov ▾ / Repository name *

✔ duckdb_1 is available.

Great repository names are short and memorable. Need inspiration? How about [legendary-parakeet?](#)

Description (optional)

- ☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:


- ☒ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set  **main** as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

Create repository

Рисунок 1. Создание репозитория

2. Клонировать репозиторий на ПК:

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/python/duckdb1
$ git clone https://github.com/umarkochkarov/duckdb_1.git
Cloning into 'duckdb_1'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 2. Клонирование репозитория

3. Организовать репозиторий в соответствии с моделью ветвления git-flow.

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/python/duckdb1/duckdb_1 (main)
$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/] Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/erken/Desktop/python/duckdb1/duckdb_1/.git/hooks]
```

Рисунок 3. Организация репозитория в соответствии с моделью git-flow

4. Проработка примеров из лабораторной работы:

Задание №1(7). Выполнение команд. Что вернула команда .schema?

```
D create table customer(name string);
D select * from customer;

name
varchar

0 rows

D .schema customer
CREATE TABLE customer("name" VARCHAR);;
```

Рисунок 4. Выполнение первого задания

Задание №2(8). Решите задачу: с помощью команды .help найдите в песочнице команду, которая отвечает за вывод времени выполнения запроса. Если ее включить, в результатах запроса добавится строчка:

```

D .timer on
D select count(*) from read_csv_auto('./city.csv');

```

count_star()
int64
1117

```

Run Time (s): real 0.078 user 0.046875 sys 0.000000

```

Рисунок 5. Выполнение второго задания

Задание №3(9). Решите задачу: загрузите файл city.csv. Затем выполните такой запрос: `select max(length(city)) from city;`

```

D select max(length(city)) from read_csv_auto('./city.csv');

```

max(length(city))
int64
25

Рисунок 6. Выполнение третьего задания

Задание №4(10). Решите задачу: загрузите файл city.csv с помощью команды `.import`, но без использования опции `--csv`. Эта опция появилась только в недавней версии SQLite (3.32, май 2020), так что полезно знать способ, подходящий для старых версий.

```

D create table city as select * from read_csv_auto('./city.csv');
D

```

Рисунок 7. Выполнение четвертого задания

Задание №5(11). Решите задачу: напишите запрос, который посчитает количество городов для каждого часового пояса в Сибирском и Приволжском федеральных округах. Выведите столбцы `timezone` и `city_count`, отсортируйте по значению часового пояса.

```

D select timezone, count(city) as city_count from read_csv_auto('./city.csv')
> where federal_district in ('Приволжский', 'Сибирский')
> group by timezone
> order by timezone ASC;

```

timezone varchar	city_count int64
UTC+3	101
UTC+4	41
UTC+5	58
UTC+6	6
UTC+7	86
UTC+8	22

Рисунок 8. Выполнение пятого задания

Задание №6(12). Решите задачу: напишите запрос, который посчитает количество городов в каждом часовом поясе. Отсортируйте по количеству городов по убыванию.

```

0 create table city as select * from read_csv_auto('./city.csv')
0 select
> timezone, count(city) as city_count from city
> group by timezone
> order by city_count DESC;

```

timezone varchar	city_count int64
UTC+3	656
UTC+5	173
UTC+7	86
UTC+4	66
UTC+9	31
UTC+8	28
UTC+10	22
UTC+2	22
UTC+11	17
UTC+12	6
UTC+6	6
11 rows	2 columns

Рисунок 9. Выполнение шестого задания

5. **Индивидуальное задание.** Загрузите в DuckDB выбранный Вами датасет в формате CSV (датасет можно найти на сайте Kaggle). Сформируйте более пяти запросов к таблицам БД. Выгрузите результат выполнения запросов в форматы CSV и JSON.

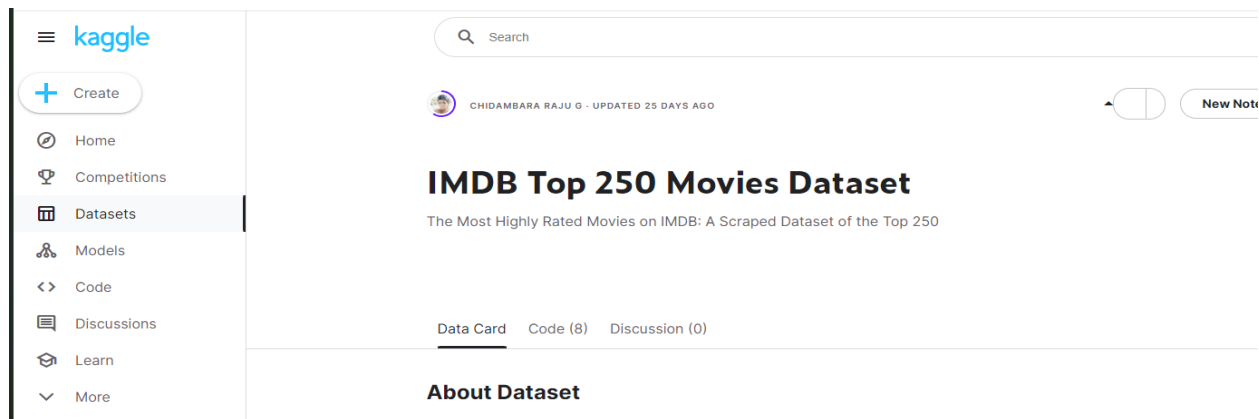


Рисунок 12. Выбранный датасет

Запрос 1. Выбрать фильмы с рейтингом больше 8.9. Сортировка по году выпуска в порядке убывания.

```

D select rank, name
> from imdb
> where rating > 8.9
> ORDER BY year DESC;

```

rank int64	name varchar
3	The Dark Knight
7	The Lord of the Rings: The Return of the King
1	The Shawshank Redemption
6	Schindler's List
4	The Godfather Part II
2	The Godfather
5	12 Angry Men

Рисунок 13. Первый запрос

Запрос 2. Выбрать фильмы с рангом от 50 до 65.

```

D select rank, name tagline from imdb where rank between 50 and 65;

```

rank int64	tagline varchar
50	Cinema Paradiso
51	Alien
52	City Lights
53	Apocalypse Now
54	Memento
55	Django Unchained
56	Indiana Jones and the Raiders of the Lost Ark
57	WALL·E
58	The Lives of Others
59	Sunset Blvd.
60	Paths of Glory
61	The Shining
62	The Great Dictator
63	Avengers: Infinity War
64	Witness for the Prosecution
65	Aliens

16 rows 2 columns

```

D copy(
> select rank, name tagline from imdb where rank between 50 and 65) to 'request2.json';
D copy(
> select rank, name tagline from imdb where rank between 50 and 65) to 'request2.csv' (header, delimiter ',');

```

Рисунок 14. Второй запрос

Запрос 3. Выбрать фильмы с кастом «Леонардо ди Каприо» и режиссером «Мартин Скорсезе».

```
0 select name, rank, year from imdb where casts like '%Leonardo DiCaprio%' and directors like '%Martin Scorsese%';
```

name varchar	rank int64	year int64
The Departed	39	2006
The Wolf of Wall Street	132	2013
Shutter Island	142	2010

```
0 copy(  
> select name, rank, year from imdb where casts like '%Leonardo DiCaprio%' and directors like '%Martin Scorsese%') to  
'request3.json';  
0 copy(  
> select name, rank, year from imdb where casts like '%Leonardo DiCaprio%' and directors like '%Martin Scorsese%') to  
'request3.csv' (header, delimiter',');
```

Рисунок 15. Третий запрос

Запрос 4. Вывести количество фильмов в каждом жанре.

```

D select genre, COUNT(*) AS count
> from imdb
> GROUP BY genre;

```

genre varchar	count int64
Drama	19
Crime,Drama	14
Action,Crime,Drama	5
Biography,Drama,History	9
Action,Adventure,Drama	5
Adventure,Western	1
Drama,Romance	6
Action,Adventure,Sci-Fi	5
Action,Adventure,Fantasy	5
Action,Sci-Fi	3
Biography,Crime,Drama	4
Crime,Drama,Mystery	9
Action,Drama	3
Drama,Family,Fantasy	1
Crime,Drama,Thriller	5
Drama,War	8
Adventure,Drama,Sci-Fi	1
Comedy,Drama,Romance	4
Crime,Drama,Fantasy	1
Adventure,Comedy,Sci-Fi	1
.	.
.	.
.	.
Adventure,Comedy,Drama	1
Adventure,Family,Fantasy	2
Adventure,Drama	2
Adventure,Comedy,Crime	1
Action,Adventure,Comedy	1
Film-Noir,Mystery,Thriller	1
Adventure,Drama,Thriller	1
Action,Comedy,Romance	1
Comedy,Drama,Thriller	1
Drama,Fantasy	1
Animation,Comedy,Drama	1
Adventure,Thriller	1
Comedy,Drama,Fantasy	1
Horror	1
Comedy,Romance,War	1
Biography,Drama,Family	2
Drama,Film-Noir,Mystery	1
Drama,Romance,Thriller	1
Comedy	1
Drama,Sport	2
104 rows (40 shown)	2 columns

```

D copy(
> select genre, COUNT(*) AS count
> from imdb
> GROUP BY genre) to 'request4.json';
D copy(
> select genre, COUNT(*) AS count
> from imdb
> GROUP BY genre) to 'request4.csv' (header, delimiter ',');

```

Рисунок 16. Четвертый запрос

Запрос 5. Вывести название и слоган фильмов режиссера «Quentin Tarantino».

```
D select name, rank, tagline from imdb where directors like '%Quentin Tarantino%';
```

name varchar	rank int64	tagline varchar
Pulp Fiction	8	Girls like me don't make invitations like this to just anyone!
Django Unchained	55	Life, liberty and the pursuit of vengeance.
Inglourious Basterds	71	Once upon a time in Nazi occupied France...
Reservoir Dogs	92	Seven Total Strangers Team Up For The Perfect Crime. They Don't Know Each Other's:
Kill Bill: Vol. 1	151	In the year 2003, Uma Thurman will kill Bill

```
D copy(
> select name, rank, tagline from imdb where directors like '%Quentin Tarantino%') to 'request5.json';
D copy(
> select name, rank, tagline from imdb where directors like '%Quentin Tarantino%') to 'request5.csv' (header, delimiter ',');
```

Рисунок 17. Пятый запрос

Запрос 6. Вывести название, режиссера и слоган фильмов 2000 года с рейтингом большим или равным 8.

```
D select name, directors, tagline from imdb where year=2000 and rating >=8.0;
```

name varchar	directors varchar	tagline varchar
Gladiator	Ridley Scott	Father of a murdered son, husband to a murdered wife and I shall have my ve:
Memento	Christopher Nolan	Some memories are best forgotten
Requiem for a Dream	Darren Aronofsky	From the director of [Pi]
Snatch	Guy Ritchie	Stealin' Stones and Breakin' Bones
Amores Perros	Alejandro G. Inarr:	Love. Betrayal. Death.

```
D copy(
```

Рисунок 18. Шестой запрос

Вывод: в результате выполнения лабораторной работы были исследованы на практике базовые возможности системы управления базами данных DuckDB.