

Отчет по ДЗ Интеграция ClickHouse с PostgreSQL

1. Инициализируйте базу данных PostgreSQL любым удобным способом.
2. Загрузите тестовый датасет в PostgreSQL.
3. С помощью функции postgres в ClickHouse запросите данные из PostgreSQL.
4. На стороне ClickHouse создайте таблицу для интеграции с PostgreSQL через движок Postgres.
5. На стороне ClickHouse создайте базу данных для интеграции с PostgreSQL.

1. PostgreSQL был развернут в докере вместе с кликом:

```
root@NB-R911LJCT:~/kafka_clickhouse# docker compose up -d
WARN[000] /root/kafka_clickhouse/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 4/4
  ✓ Container zookeeper   Running
  ✓ Container postgres    Running
  ✓ Container clickhouse  Running
  ✓ Container kafka       Running
root@NB-R911LJCT:~/kafka_clickhouse#
root@NB-R911LJCT:~/kafka_clickhouse# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
bd99d0c6b6f4   bitnami/kafka:2.8.1                "/opt/bitnami/script.." 2 hours ago   Up 9 seconds  0.0.0.0:9092->9092/tcp, [::]:9092->9092/tcp
f9357d7fed5    bitnami/zookeeper:3.8              "/opt/bitnami/script.." 2 hours ago   Up 33 seconds 2888/tcp, 3888/tcp, 0.0.0.0:2181->2181/tcp, [::]:2181->2181/tcp, 8080/tcp
aea02d90a7dd   yandex/clickhouse-server:latest    "/entrypoint.sh"        21 hours ago  Up 9 seconds  0.0.0.0:8123->8123/tcp, [::]:8123->8123/tcp, 0.0.0.0:9000->9000/tcp, [::]:9000->9000/tcp, 9009/tcp
a71ec7dc6cb7   postgres:latest                    "docker-entrypoint.s.." 21 hours ago  Up 33 seconds  5432/tcp
```

2. Я решил что сам создам БД и элементарную тестовую таблицу в PostgreSQL

- 2.1. Сначала зашел в контейнер с PostgreSQL:

```
root@NB-R911LJCT:~/kafka_clickhouse# docker exec -it postgres psql -U test
psql (17.4 (Debian 17.4-1.pgdg120+2))
Type "help" for help.

test=#
```

- 2.2. Создал новую БД:

```
test=# CREATE DATABASE otus;
CREATE DATABASE
test=#
```

- 2.3. Перехожу в эту базу данных:

```
test=# \c otus
You are now connected to database "otus" as user "test".
otus=#
```

- 2.4. Делаю элементарную тестовую табличку:

```
otus=# CREATE TABLE mytable (id SERIAL PRIMARY KEY, name VARCHAR(100));
CREATE TABLE
otus=#
```

- 2.5. Заполняю ее одной строкой:

```
otus=# INSERT INTO mytable (name) VALUES ('Example');
INSERT 0 1
otus=#
```

2.6. Убеждаюсь что данные залетели:

```
otus=# SELECT * FROM mytable;
 id | name 
----+-----
  1 | Example
(1 row)
```

3. Захожу в клик и забираю данные из таблички через функцию postgres:

The screenshot shows a database interface with a query editor and a results pane. The query editor contains the following SQL query:

```
SELECT * FROM postgresql('postgres:5432', 'otus', 'mytable', 'test', 'test');
```

The results pane shows the following output:

id	name
1	Example

4. Создаю табличку, для интеграции Клика с PostgreSQL и читаю из нее данные:

The screenshot shows a database interface with a query editor and a results pane. The query editor contains the following SQL query:

```
CREATE TABLE otus_postgres (  
  id UInt32,  
  name String  
) ENGINE = PostgreSQL('postgres:5432', 'otus', 'mytable', 'test', 'test');
```

The results pane shows the following output:

id	name
1	Example

5. Создаю и настраиваю БД в клике интегрированную с БД PostgreSQL и через нее читаю данные из таблицы на PostgreSQL:

```
347 CREATE DATABASE postgres_otus
348 ENGINE = PostgreSQL('postgres:5432', 'otus', 'test', 'test');
349
350 ✓ SELECT * FROM postgres_otus.mytable;
351
```

Output postgres_otus.mytable

id	name
1	Example

Так же удалось выполнить вставку в PostgreSQL через функцию postgres:

```
352 INSERT INTO FUNCTION postgresql('postgres:5432', 'otus', 'mytable', 'test', 'test') VALUES
353 ( postgresql 2, 'John Doe'),
354 ( postgresql 3, 'Jane Smith'),
355 ( postgresql 4, 'Alice Johnson');
356
```

Output postgresql

id	name
1	Example
2	John Doe
3	Jane Smith
4	Alice Johnson

```
otus=# SELECT * FROM mytable;
 id |      name
----+-----
  1 | Example
  2 | John Doe
  3 | Jane Smith
  4 | Alice Johnson
(4 rows)

otus=#
```