**Задание 9.**

**Миграция данных из PostgreSQL в ClickHouse в Minikube**

**Цель: Научиться развертывать PostgreSQL и ClickHouse в Kubernetes с использованием Minikube, и реализовать процесс миграции данных из PostgreSQL в ClickHouse.**

Описание задания:

* Запустите Minikube с достаточными ресурсами.
* Создайте файл postgres-deployment.yaml для развертывания PostgreSQL.
* Создайте файл clickhouse-deployment.yaml для развертывания ClickHouse.
* Подключитесь к PostgreSQL с помощью psql.
* Выполните следующие SQL-запросы для создания таблицы и добавления данных:

CREATE TABLE users (

    id SERIAL PRIMARY KEY,

    name VARCHAR(100),

    age INT,

    salary INT

);

INSERT INTO users (name, age, salary) VALUES

('Alice', 30, 70000),

('Bob', 25, 50000),

('Charlie', 35, 100000),

('David', 40, 120000),

('Eve', 28, 60000),

('Frank', 50, 150000);

* Подключитесь к ClickHouse с помощью clickhouse-client.
* Выполните следующий SQL-запрос для создания таблицы в ClickHouse:

CREATE TABLE users (

    id UInt32,

    name String,

    age UInt8,

    salary UInt32

) ENGINE = MergeTree()

ORDER BY id;

* В локальной среде (не в Minikube) создайте файл migration.py.
* Убедитесь, что PostgreSQL и ClickHouse доступны через портфорвардинг, и запустите скрипт, который перенесет данные из PG в ClickHouse.

В качестве решения, разместите в репозитории 2 yaml-файла развертывания БД, файл migration.py, а также сделайте скриншоты того, что в ClickHouse данные появились.

Результат задания — после выполнения задания у вас будет развернут Minikube-кластер с PostgreSQL и ClickHouse, настроена миграция данных между этими системами и успешно завершен процесс миграции.

**Шаг 1. Установка и запуск Minikube**

1. **Убедитесь, что Minikube установлен:**
   * Установите Minikube, если он не установлен.
2. **Запустите Minikube с достаточными ресурсами:**

minikube start --cpus=4 --memory=8192

1. **Проверьте статус кластера:**

kubectl cluster-info

**Шаг 2. Создание и развертывание PostgreSQL**

1. **Создайте файл postgres-deployment.yaml:**
2. apiVersion: apps/v1
3. kind: Deployment
4. metadata:
5. name: postgres
6. spec:
7. replicas: 1
8. selector:
9. matchLabels:
10. app: postgres
11. template:
12. metadata:
13. labels:
14. app: postgres
15. spec:
16. containers:
17. - name: postgres
18. image: postgres:15
19. ports:
20. - containerPort: 5432
21. env:
22. - name: POSTGRES\_USER
23. value: "admin"
24. - name: POSTGRES\_PASSWORD
25. value: "password"
26. - name: POSTGRES\_DB
27. value: "test\_db"
28. ---
29. apiVersion: v1
30. kind: Service
31. metadata:
32. name: postgres
33. spec:
34. ports:
35. - port: 5432
36. selector:
37. app: postgres
38. **Примените конфигурацию:**

kubectl apply -f postgres-deployment.yaml

1. **Проверьте статус:**

kubectl get pods

**Шаг 3. Создание и развертывание ClickHouse**

1. **Создайте файл clickhouse-deployment.yaml:**
2. apiVersion: apps/v1
3. kind: Deployment
4. metadata:
5. name: clickhouse
6. spec:
7. replicas: 1
8. selector:
9. matchLabels:
10. app: clickhouse
11. template:
12. metadata:
13. labels:
14. app: clickhouse
15. spec:
16. containers:
17. - name: clickhouse
18. image: yandex/clickhouse-server:latest
19. ports:
20. - containerPort: 9000
21. - containerPort: 8123
22. env:
23. - name: user
24. value: "admin"
25. - name: password
26. value: "password"
27. ---
28. apiVersion: v1
29. kind: Service
30. metadata:
31. name: clickhouse
32. spec:
33. ports:
34. - port: 9000
35. targetPort: 9000
36. protocol: TCP
37. name: http
38. selector:
39. app: clickhouse
40. **Примените конфигурацию:**

kubectl apply -f clickhouse-deployment.yaml

1. **Проверьте статус:**

kubectl get pods

**Шаг 4. Работа с PostgreSQL**

1. **Установите портфорвардинг для PostgreSQL:**

kubectl port-forward svc/postgres 5432:5432

1. **Подключитесь к PostgreSQL:**

psql -h localhost -U admin -d test\_db

1. **Создайте таблицу и добавьте данные:**

CREATE TABLE users (

    id SERIAL PRIMARY KEY,

    name VARCHAR(100),

    age INT,

    salary INT

);

INSERT INTO users (name, age, salary) VALUES

('Alice', 30, 70000),

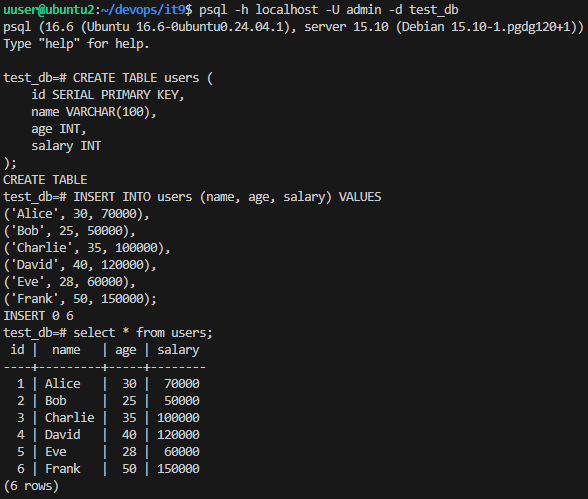
('Bob', 25, 50000),

('Charlie', 35, 100000),

('David', 40, 120000),

('Eve', 28, 60000),

('Frank', 50, 150000);



**Шаг 5. Работа с ClickHouse**

1. **Установите портфорвардинг для ClickHouse:**

kubectl port-forward svc/clickhouse 9000:9000

1. **Подключитесь к ClickHouse:**

clickhouse-client --host=localhost

1. **Создайте таблицу в ClickHouse:**

CREATE TABLE users (

    id UInt32,

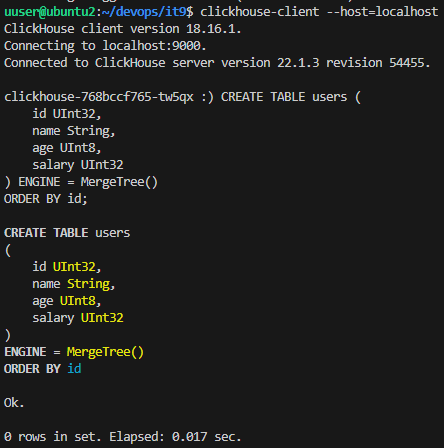
    name String,

    age UInt8,

    salary UInt32

) ENGINE = MergeTree()

ORDER BY id;



**Шаг 6. Миграция данных (Python-скрипт)**

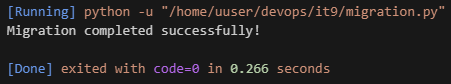
1. **Создайте файл migration.py:**
2. import psycopg2
3. from clickhouse\_driver import Client
4. # PostgreSQL connection
5. pg\_conn = psycopg2.connect(
6. dbname="test\_db",
7. user="admin",
8. password="password",
9. host="localhost",
10. port=5432
11. )
12. pg\_cursor = pg\_conn.cursor()
13. # Fetch data from PostgreSQL
14. pg\_cursor.execute("SELECT id, name, age FROM users;")
15. rows = pg\_cursor.fetchall()
16. # ClickHouse connection
17. ch\_client = Client(host='localhost', port=9000)
18. # Insert data into ClickHouse
19. for row in rows:
20. ch\_client.execute("INSERT INTO users (id, name, age) VALUES", [row])
21. print("Migration completed successfully!")
22. # Close connections
23. pg\_cursor.close()
24. pg\_conn.close()
25. **Убедитесь, что PostgreSQL и ClickHouse доступны через портфорвардинг.**
26. **Запустите скрипт:**

sudo apt install python-is-python3

sudo apt install python3-psycopg2

sudo apt install python3- clickhouse\_driver

python migration.py



**Шаг 7. Проверка миграции данных**

1. **Подключитесь к ClickHouse и проверьте данные:**

SELECT \* FROM users;

