

12 ДЗ - RBAC контроль доступа, квоты и ограничения.

Собираем Dockerfile.

clickhouse-25.2.1.Dockerfile:

```
FROM clickhouse/clickhouse-server:25.2.1
MAINTAINER Maksim Kulikov <max.uoles@rambler.ru>

RUN apt-get update -y --fix-missing
RUN DEBIAN_FRONTEND=noninteractive apt-get -yq upgrade
RUN apt-get install nano mc python3 pip kafkacat -y
RUN pip install clickhouse_driver

EXPOSE 8123 9000

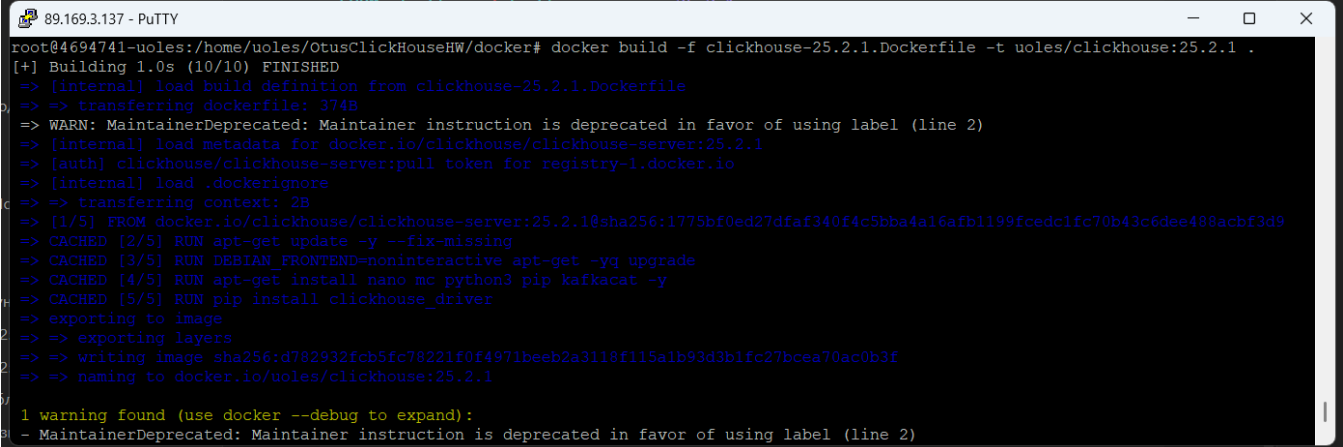
ENTRYPOINT ["/entrypoint.sh"]
```

Собираем образ командой:

```
docker build -f clickhouse-25.2.1.Dockerfile -t uoles/clickhouse:25.2.1 .
```

Запускаем:

```
docker run -d \
-p 18123:8123 \
-p 19000:9000 \
-e CLICKHOUSE_DB=my_database \
-e CLICKHOUSE_USER=username \
-e CLICKHOUSE_DEFAULT_ACCESS_MANAGEMENT=1 \
-e CLICKHOUSE_PASSWORD=password \
--name clickhouse-server \
--ulimit nofile=262144:262144 \
uoles/clickhouse:25.2.1
```



```
89.169.3.137 - PuTTY
root@4694741-uoles:/home/uoles/OtusClickHouseHW/docker# docker build -f clickhouse-25.2.1.Dockerfile -t uoles/clickhouse:25.2.1 .
[+] Building 1.0s (10/10) FINISHED
=> [internal] load build definition from clickhouse-25.2.1.Dockerfile
=> => transferring dockerfile: 374B
=> WARN: MaintainerDeprecated: Maintainer instruction is deprecated in favor of using label (line 2)
=> [internal] load metadata for docker.io/clickhouse/clickhouse-server:25.2.1
=> [auth] clickhouse/clickhouse-server:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/5] FROM docker.io/clickhouse/clickhouse-server:25.2.1@sha256:1775bf0ed27dfaf340f4c5bba4a16afb1199fcedc1fc70b43c6dee488acbf3d9
=> CACHED [2/5] RUN apt-get update -y --fix-missing
=> CACHED [3/5] RUN DEBIAN_FRONTEND=noninteractive apt-get -yq upgrade
=> CACHED [4/5] RUN apt-get install nano mc python3 pip kafkacat -y
=> CACHED [5/5] RUN pip install clickhouse_driver
=> exporting to image
=> => exporting layers
=> => writing image sha256:d782932fcb5fc78221f0f4971beeb2a3118f115a1b93d3b1fc27bcea70ac0b3f
=> => naming to docker.io/uoles/clickhouse:25.2.1

1 warning found (use docker --debug to expand):
- MaintainerDeprecated: Maintainer instruction is deprecated in favor of using label (line 2)
```

Создание пользователя и роли.

Выполняем запросы:

```
CREATE USER jhon IDENTIFIED WITH sha256_password BY 'qwerty'

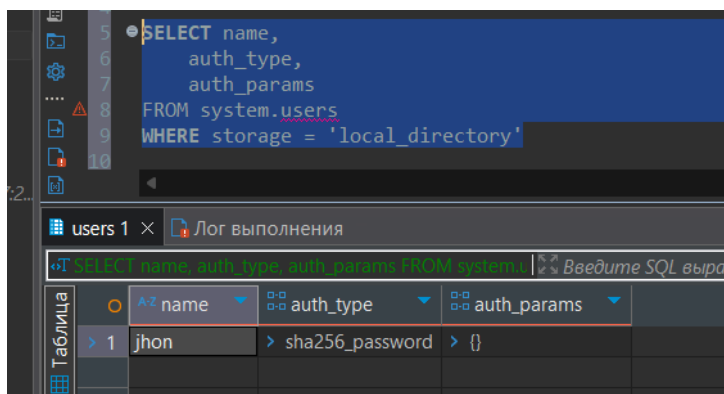
CREATE ROLE devs;

GRANT SELECT ON my_database.* TO devs;

GRANT devs TO jhon;
```

Проверяем наличие пользователя и роли:

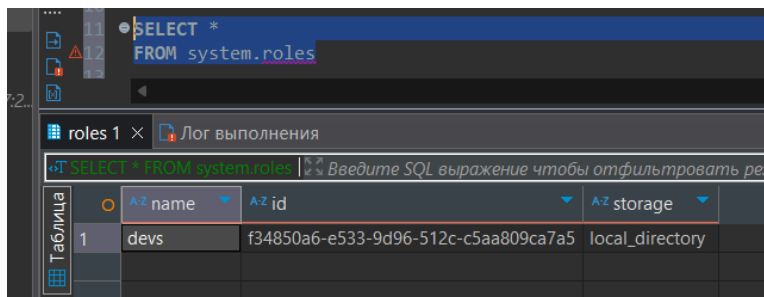
```
SELECT name,
       auth_type,
       auth_params
FROM system.users
WHERE storage = 'local_directory'
```



The screenshot shows a SQL query editor with the query: `SELECT name, auth_type, auth_params FROM system.users WHERE storage = 'local_directory'`. Below the editor, the results are displayed in a table with columns: name, auth_type, and auth_params. The first row shows the user 'jhon' with auth_type 'sha256_password' and an empty auth_params field.

	A-Z name	A-Z auth_type	A-Z auth_params
1	jhon	sha256_password	{}

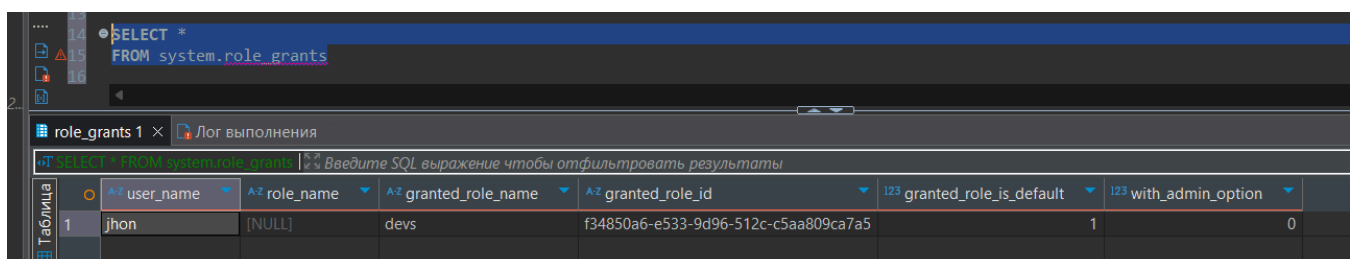
```
SELECT *
FROM system.roles
```



The screenshot shows a SQL query editor with the query: `SELECT * FROM system.roles`. Below the editor, the results are displayed in a table with columns: name, id, and storage. The first row shows the role 'devs' with id 'f34850a6-e533-9d96-512c-c5aa809ca7a5' and storage 'local_directory'.

	A-Z name	A-Z id	A-Z storage
1	devs	f34850a6-e533-9d96-512c-c5aa809ca7a5	local_directory

```
SELECT *
FROM system.role_grants
```



The screenshot shows a SQL query editor with the query: `SELECT * FROM system.role_grants`. Below the editor, the results are displayed in a table with columns: user_name, role_name, granted_role_name, granted_role_id, granted_role_is_default, and with_admin_option. The first row shows the user 'jhon' granted the role 'devs' with id 'f34850a6-e533-9d96-512c-c5aa809ca7a5', with granted_role_is_default set to 1 and with_admin_option set to 0.

	A-Z user_name	A-Z role_name	A-Z granted_role_name	A-Z granted_role_id	123 granted_role_is_default	123 with_admin_option
1	jhon	[NULL]	devs	f34850a6-e533-9d96-512c-c5aa809ca7a5	1	0

```
SELECT
grants.user_name,
```

```

grants.role_name,
users.name AS role_member,
grants.access_type,
grants.database,
grants.table
FROM system.grants
LEFT JOIN system.role_grants ON grants.role_name = role_grants.granted_role_name
LEFT JOIN system.users ON role_grants.user_name = users.name
WHERE users.name = 'jhon'

```

38
39 ● SELECT
40 grants.user_name,
41 grants.role_name,
42 users.name AS role_member,
43 grants.access_type,
44 grants.database,
45 grants.table
46 FROM system.grants
47 LEFT JOIN system.role_grants ON grants.role_name = role_grants.granted_role_name
48 LEFT JOIN system.users ON role_grants.user_name = users.name
49 WHERE users.name = 'jhon'
50

grants 1 × Лог выполнения

Введите SQL выражение чтобы отфильтровать результаты

	A-Z grants.user_name	A-Z grants.role_name	A-Z role_member	A-Z access_type	A-Z database	A-Z table
1	[NULL]	devs	jhon	SELECT	my_database	[NULL]