**13 ДЗ - Storage Policy и резервное копирование.**

**Собираем Dockerfile и docker-compose.**

- clickhouse-minio-25.2.1.Dockerfile – для бэкапа данных средствами clickhouse:  
  
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

COPY clickhouse/backup.xml /etc/clickhouse-server/config.d/backup.xml

EXPOSE 8123 9000 9363

ENTRYPOINT ["/entrypoint.sh"]

- clickhouse-backup-25.2.1.Dockerfile – для бэкапа данных через clickhouse-backup:  
  
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

RUN mkdir /tmp/clickhouse-backup \

&& cd /tmp/clickhouse-backup \

&& wget https://github.com/Altinity/clickhouse-backup/releases/download/v2.6.16/clickhouse-backup-linux-amd64.tar.gz \

&& tar -xf clickhouse-backup-linux-amd64.tar.gz \

&& install -o root -g root -m 0755 build/linux/amd64/clickhouse-backup /usr/local/bin \

&& mkdir /etc/clickhouse-backup

COPY clickhouse/clickhouse-backup/config.yml /etc/clickhouse-backup/config.yml

EXPOSE 8123 9000

ENTRYPOINT ["/entrypoint.sh"]

Собираем docker-compose.yml для запуска приложений:

docker-compose-backup.yml

version: "3.6"

services:

clickhouse-server-1:

container\_name: clickhouse-server-1

image: uoles/clickhouse-minio:25.2.1

hostname: clickhouse-server-1

build:

context: .

dockerfile: clickhouse-minio-25.2.1.Dockerfile

environment:

CLICKHOUSE\_DB: my\_database

CLICKHOUSE\_USER: username

CLICKHOUSE\_DEFAULT\_ACCESS\_MANAGEMENT: 1

CLICKHOUSE\_PASSWORD: password

ports:

- "18123:8123"

- "19000:9000"

ulimits:

nofile:

soft: 262144

hard: 262144

depends\_on:

minio:

condition: service\_healthy

createbuckets:

condition: service\_started

links:

- minio

networks:

- default

clickhouse-server-2:

container\_name: clickhouse-server-2

image: uoles/clickhouse:25.2.1

hostname: clickhouse-server-2

build:

context: .

dockerfile: clickhouse-backup-25.2.1.Dockerfile

environment:

CLICKHOUSE\_DB: my\_database

CLICKHOUSE\_USER: username

CLICKHOUSE\_DEFAULT\_ACCESS\_MANAGEMENT: 1

CLICKHOUSE\_PASSWORD: password

ports:

- "28123:8123"

- "29000:9000"

ulimits:

nofile:

soft: 262144

hard: 262144

depends\_on:

minio:

condition: service\_healthy

createbuckets:

condition: service\_started

links:

- minio

networks:

- default

minio:

image: quay.io/minio/minio

container\_name: minio

hostname: minio

command: server --address 0.0.0.0:10000 --console-address 0.0.0.0:10001 /data

ports:

- "10000:10000"

- "10001:10001"

environment:

- MINIO\_ROOT\_USER=minio-root-user

- MINIO\_ROOT\_PASSWORD=minio-root-password

networks:

- default

healthcheck:

test: [ "CMD", "curl", "-f", "http://localhost:10001" ]

interval: 5s

timeout: 10s

retries: 5

createbuckets:

image: minio/mc

entrypoint: >

/bin/sh -c "

/usr/bin/mc alias set myminio http://minio:10000 minio-root-user minio-root-password;

/usr/bin/mc admin info myminio;

/usr/bin/mc mb myminio/clickhouse;

/usr/bin/mc policy set public myminio/clickhouse;

exit 0;

"

depends\_on:

minio:

condition: service\_healthy

links:

- minio

networks:

- default

networks:

default:

ipam:

driver: default

config:

- subnet: 172.28.0.0/16

clickhouse-server-1 – для тестирования бэкапа данных средствами clickhouse.

clickhouse-server-2 – для тестирования бэкапа данных через clickhouse-backup.

createbuckets – создает бакет в minio.

Настройки storage policy:

/etc/clickhouse-server/config.d/backup.xml

<clickhouse>

<storage\_configuration>

<disks>

<s3\_plain>

<type>s3\_plain</type>

<endpoint>http://minio:10000/clickhouse/</endpoint>

<access\_key\_id>minio-root-user</access\_key\_id>

<secret\_access\_key>minio-root-password</secret\_access\_key>

</s3\_plain>

</disks>

<policies>

<s3>

<volumes>

<main>

<disk>s3\_plain</disk>

</main>

</volumes>

</s3>

</policies>

</storage\_configuration>

<backups>

<allowed\_disk>s3\_plain</allowed\_disk>

</backups>

</clickhouse>

Настройки для clickhouse-backup:

/etc/clickhouse-backup/config.yml

general:

remote\_storage: s3

disable\_progress\_bar: false

backups\_to\_keep\_local: 0

backups\_to\_keep\_remote: 0

log\_level: info

allow\_empty\_backups: false

clickhouse:

username: username

password: password

skip\_tables:

- system.\*

- INFORMATION\_SCHEMA.\*

s3:

access\_key: minio-root-user

secret\_key: minio-root-password

bucket: ""

endpoint: http://minio:10000/clickhouse/

region: us-west-000

acl: ""

force\_path\_style: false

path: clickhouse-backup

disable\_ssl: false

part\_size: 536870912

compression\_level: 1

compression\_format: tar

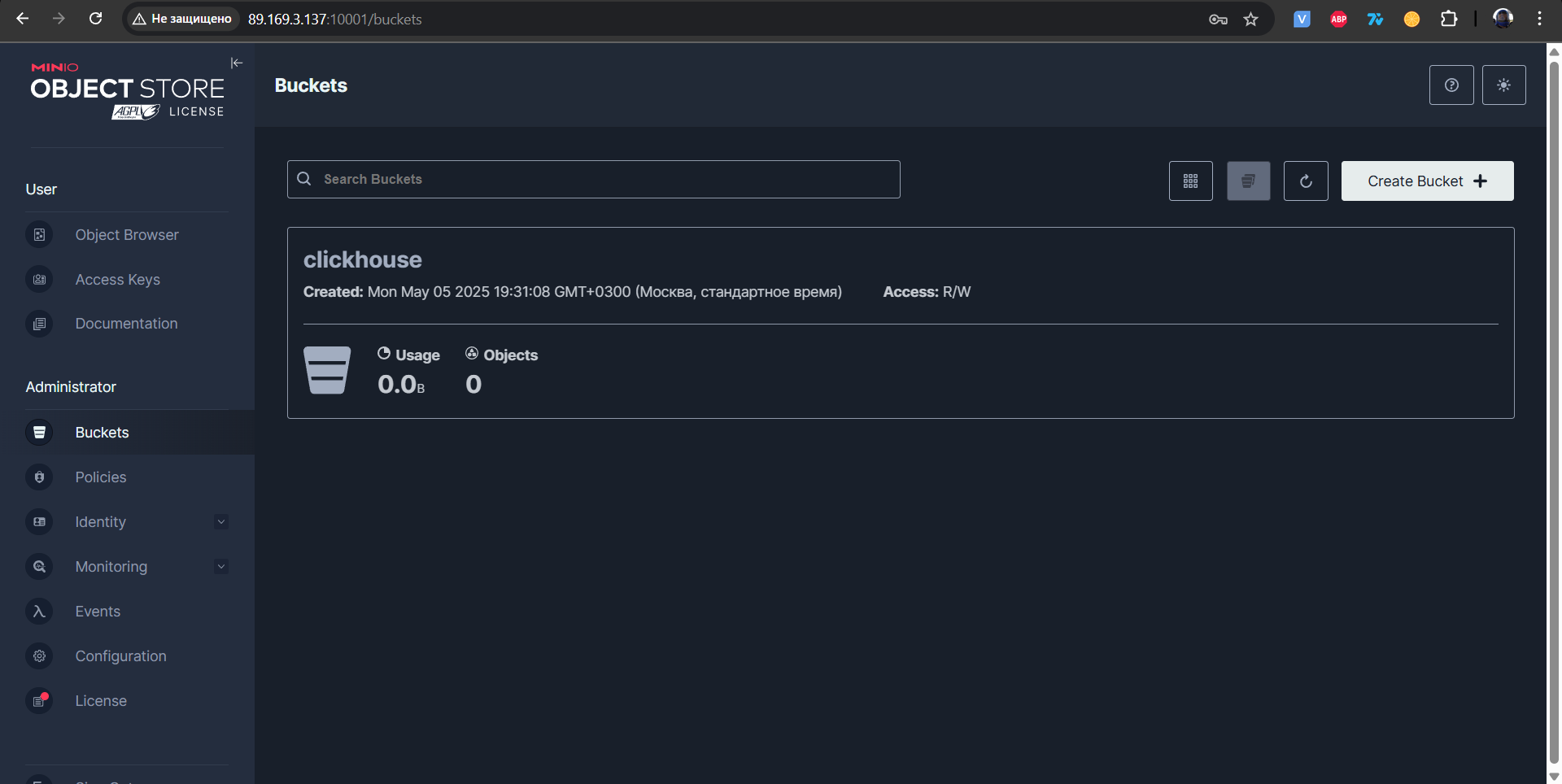
sse: ""

disable\_cert\_verification: false

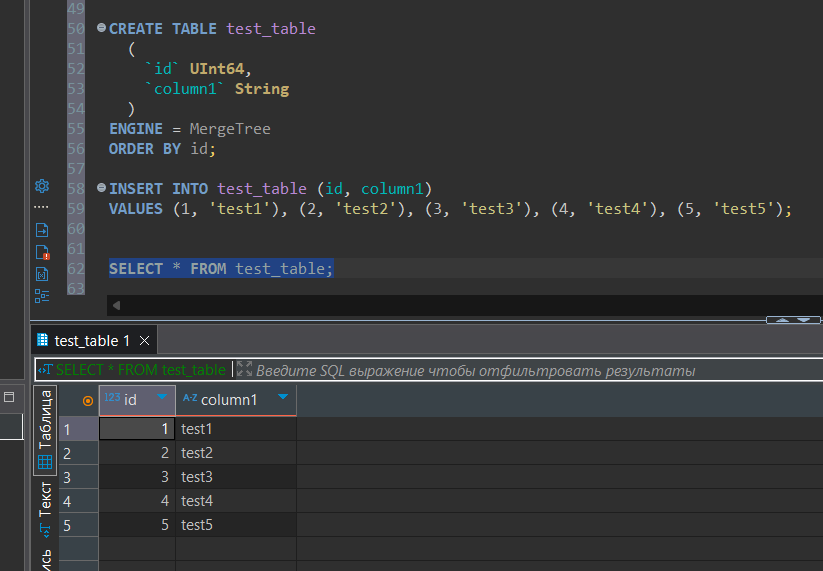
storage\_class: STANDARD

**Создание бэкапа средствами clickhouse.**

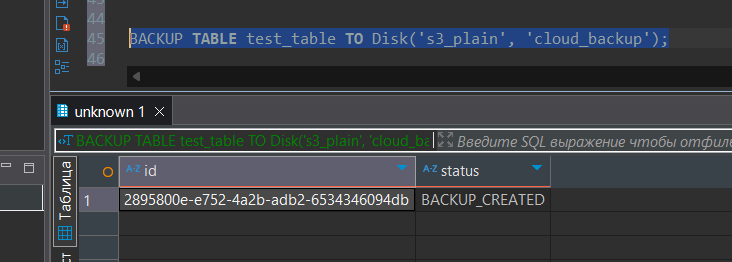
Проверяем наличие созданного бакета в minio:



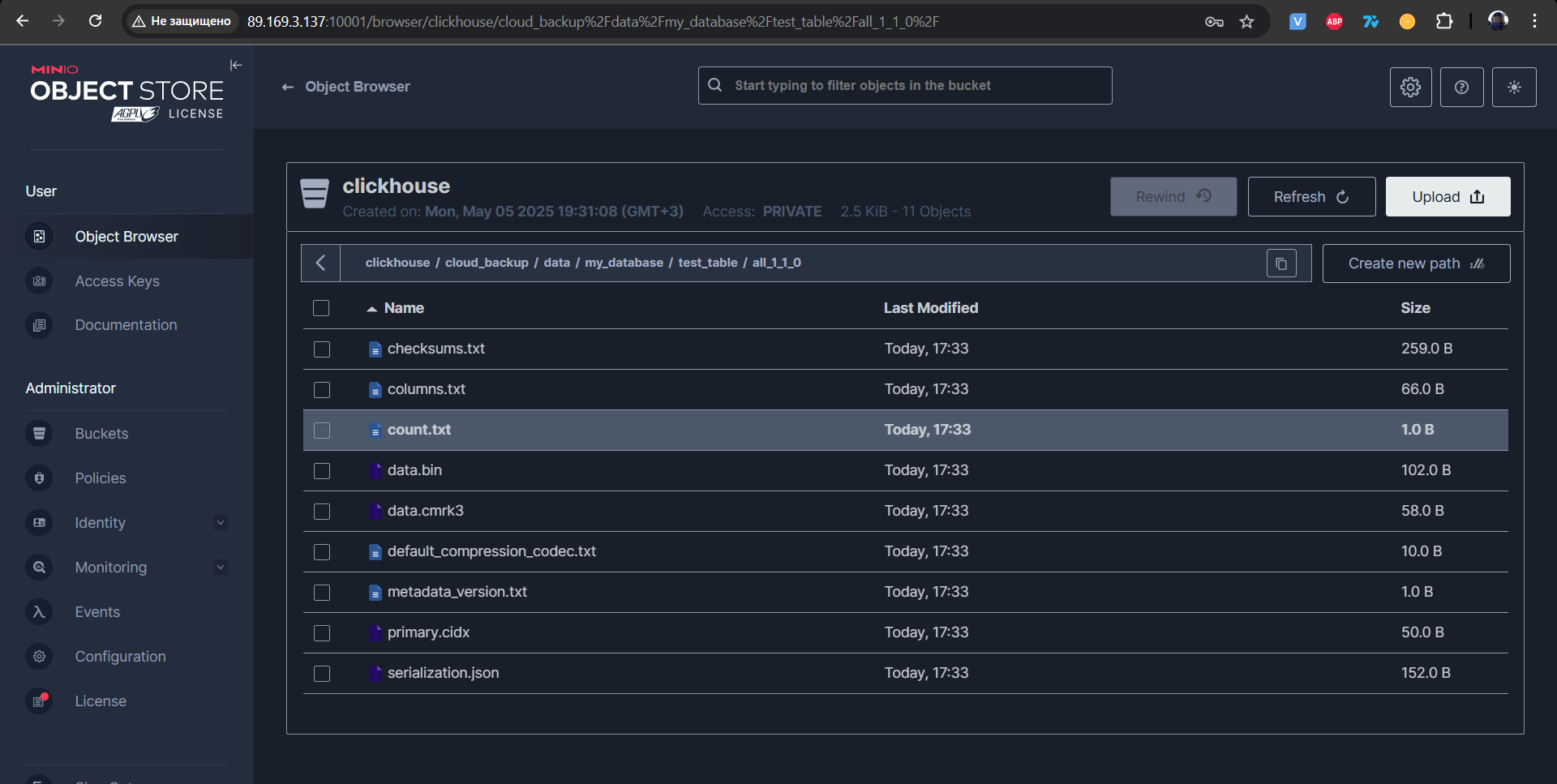
Подключаемся к clickhouse-server-1 и создаем таблицу с данными:



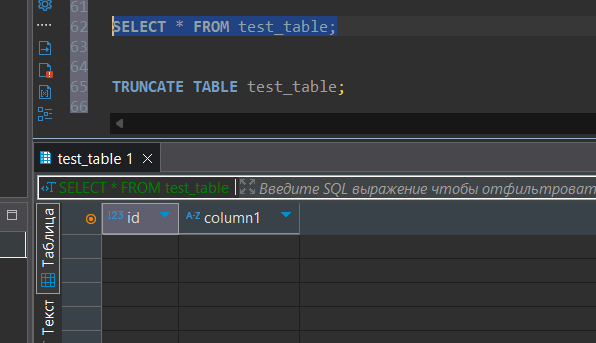
Делаем бэкап данных:



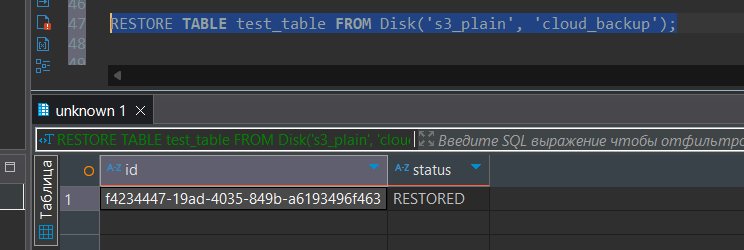
Проверяем данные в minio:

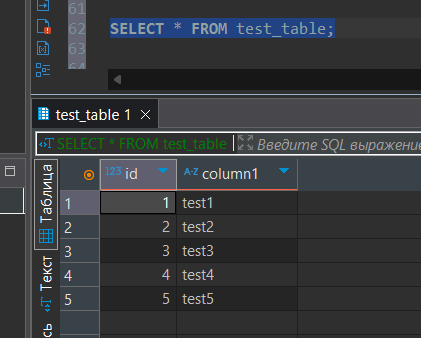


Очищаем таблицу и проверяем:



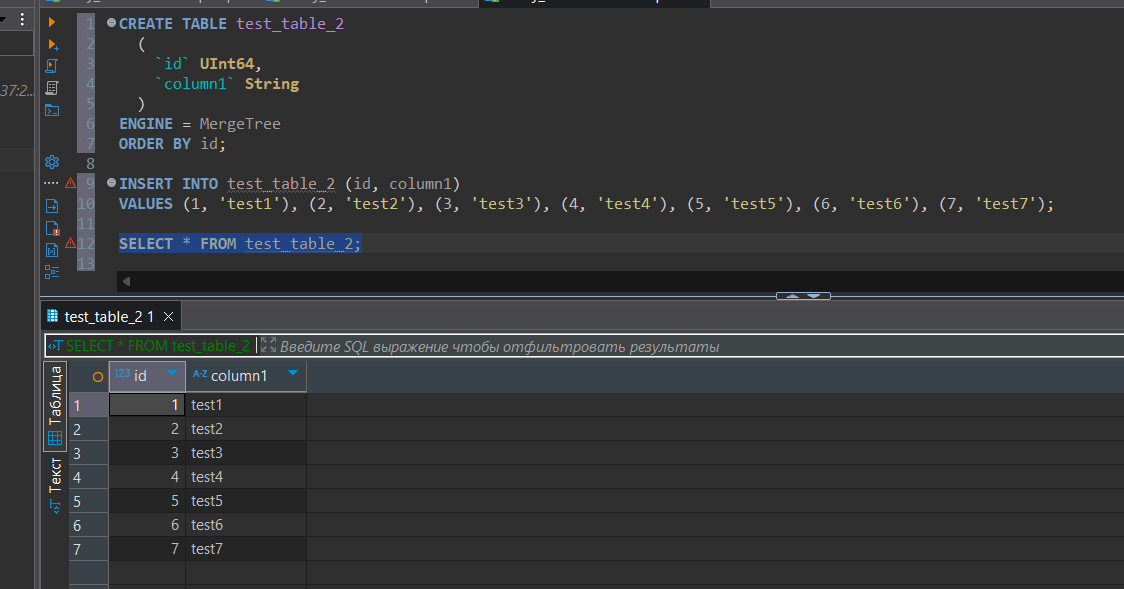
Восстанавливаем данные и проверяем:





**Делаем бэкап данных через cklickhouse-backup.**

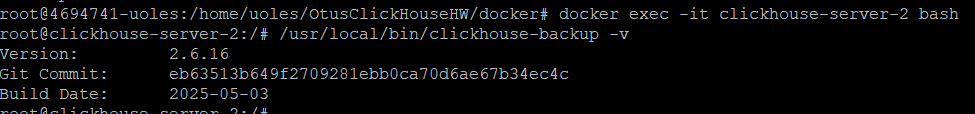
Подключаемся к clickhouse-server-2 и создаем таблицу с данными:



Заходим в контейнер clickhouse-server-2 командой:

>> docker exec -it clickhouse-server-2 bash

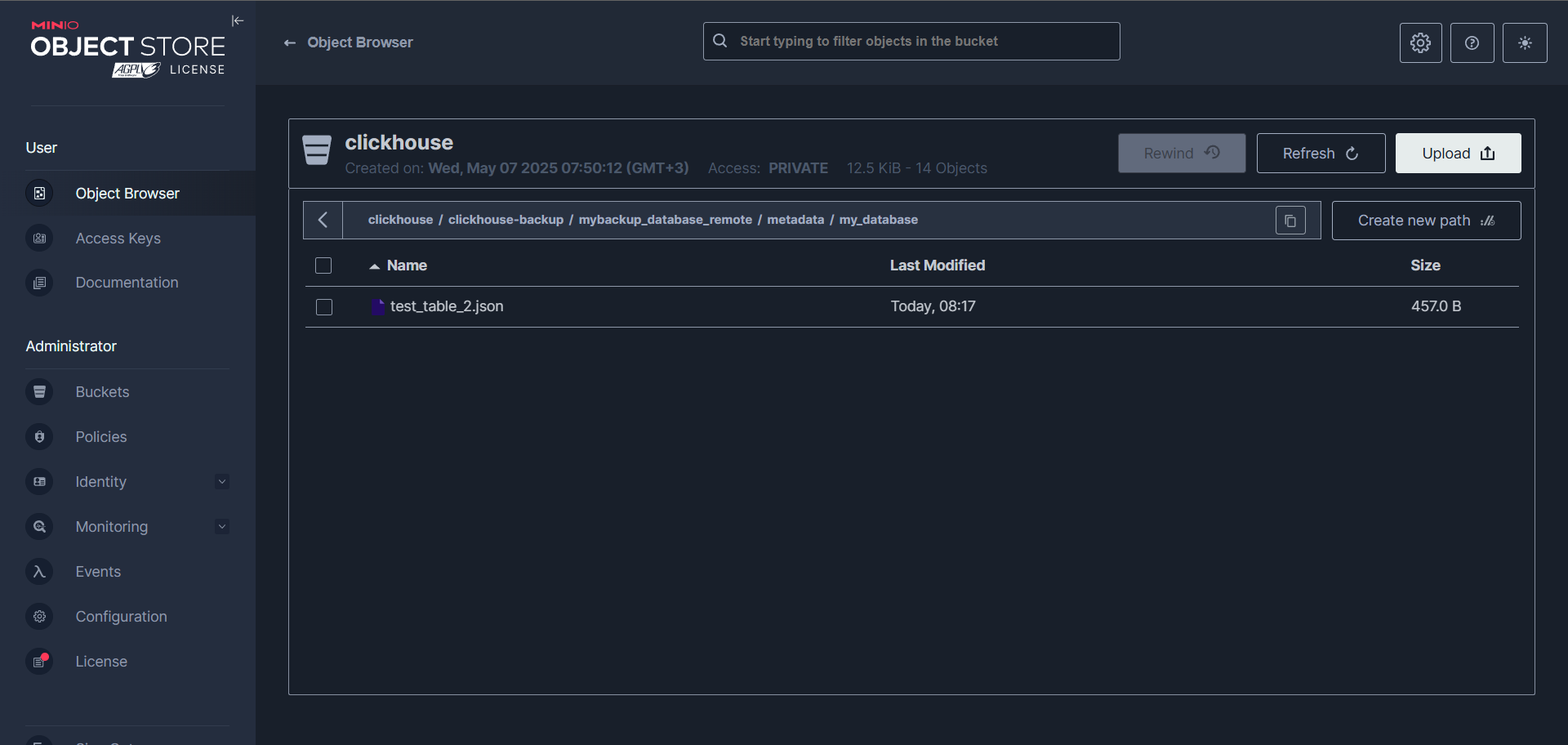
и проверяем наличие clickhouse-backup:



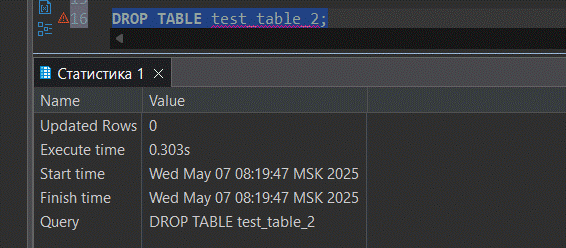
Делаем бэкап базы данных «my\_database» командой:

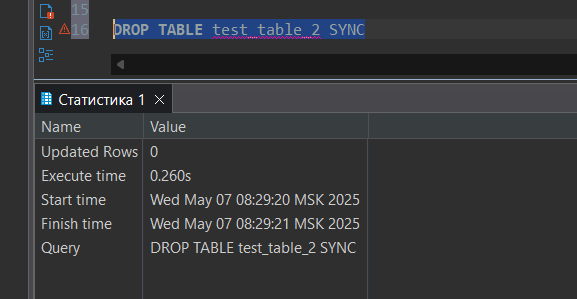
>> clickhouse-backup create\_remote mybackup\_database\_remote -t 'my\_database.\*'

Проверяем наличие бэкапа в minio:



Удаляем таблицу в clickhouse:





Восстанавливаем схему из бекапа командой:

>> clickhouse-backup restore\_remote mybackup\_database\_remote -t 'my\_database.\*'

и проверяем таблицу:

