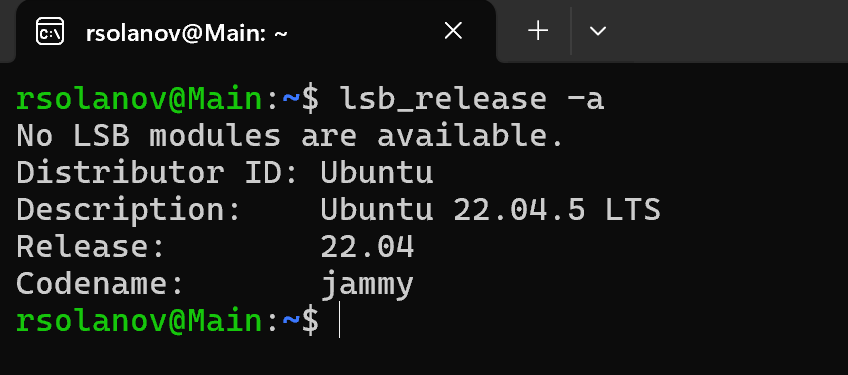
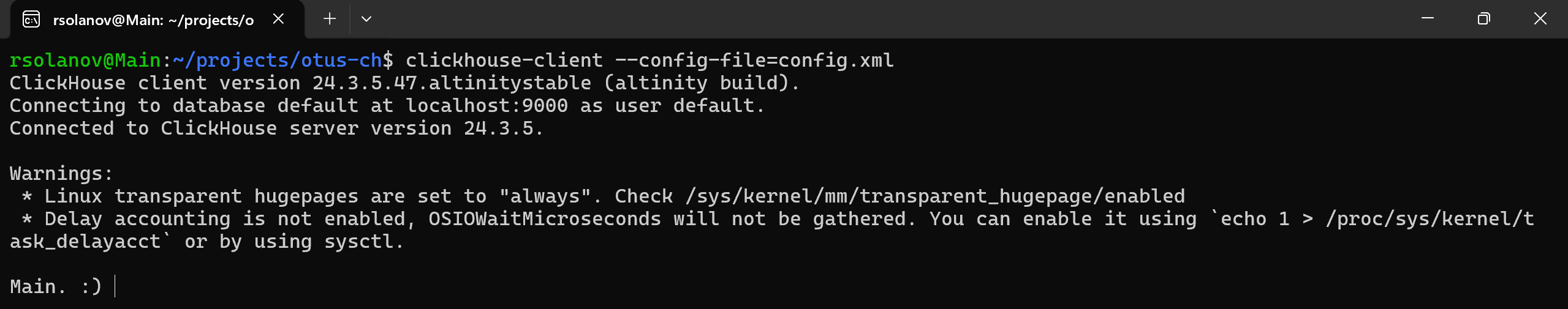
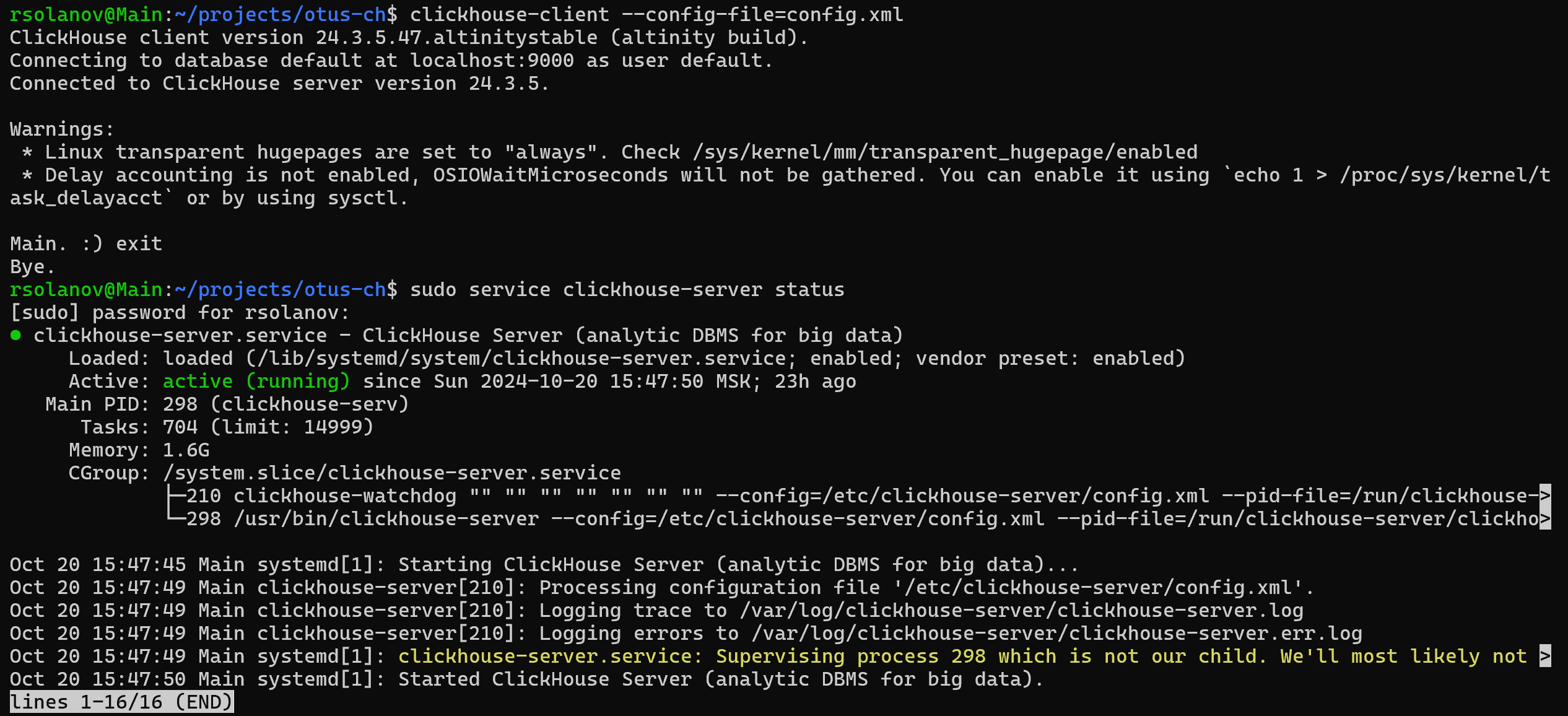
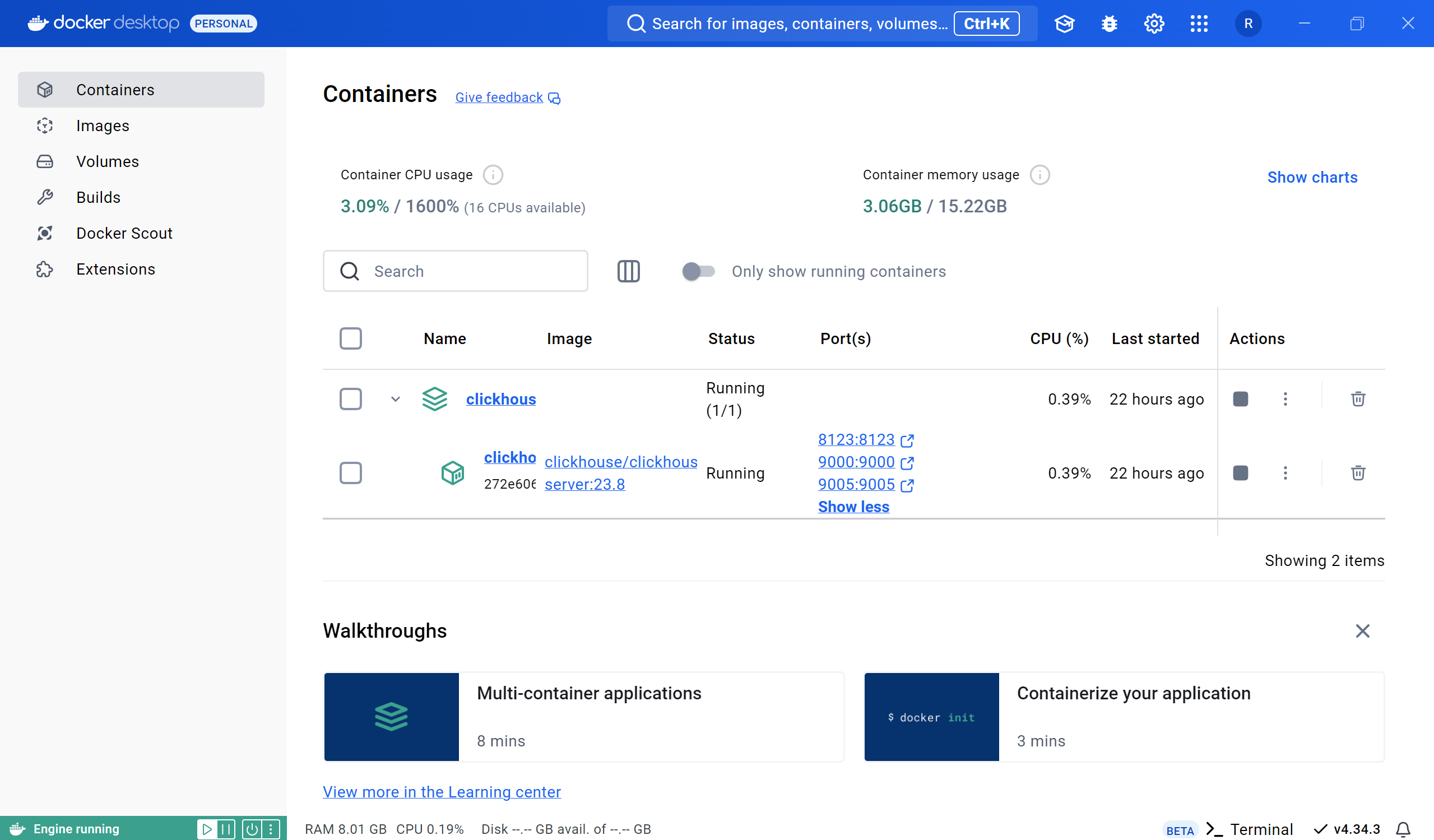
* Установить ClickHouse.
* Подгрузить датасет для примера и сделать селект из таблицы.
* Для проверки отправить скрины работающего инстанса ClickHouse, созданной виртуальной машины и результата запроса select count() from trips where payment\_type = 1.
  + Установка ClickHouse в WSL Ubuntu 22.04.5 LTS

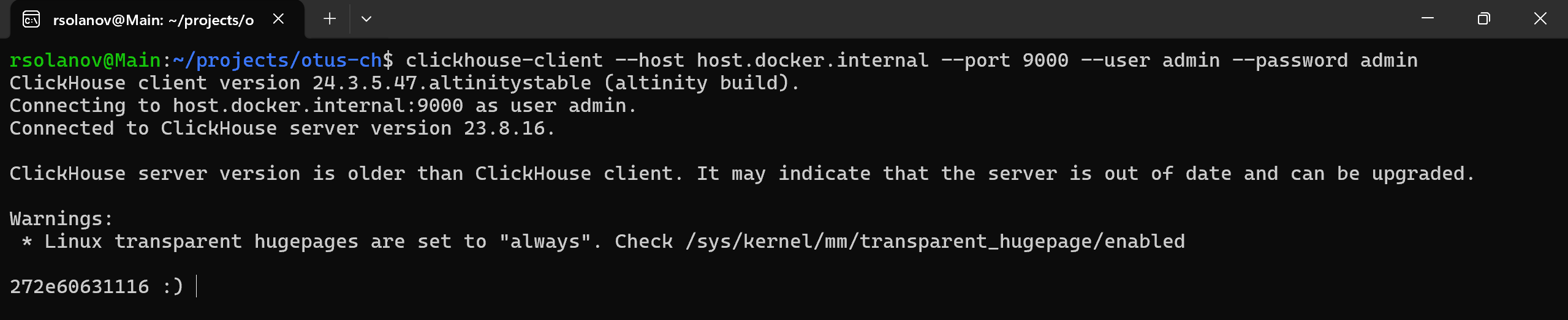


****

****

* + Установка ClickHouse из Docker’а

****

****

* + Загрузка датасета

**CREATE TABLE trips (**

**trip\_id UInt32,**

**pickup\_datetime DateTime,**

**dropoff\_datetime DateTime,**

**pickup\_longitude Nullable(Float64),**

**pickup\_latitude Nullable(Float64),**

**dropoff\_longitude Nullable(Float64),**

**dropoff\_latitude Nullable(Float64),**

**passenger\_count UInt8,**

**trip\_distance Float32,**

**fare\_amount Float32,**

**extra Float32,**

**tip\_amount Float32,**

**tolls\_amount Float32,**

**total\_amount Float32,**

**payment\_type Enum('CSH' = 1, 'CRE' = 2, 'NOC' = 3, 'DIS' = 4, 'UNK' = 5),**

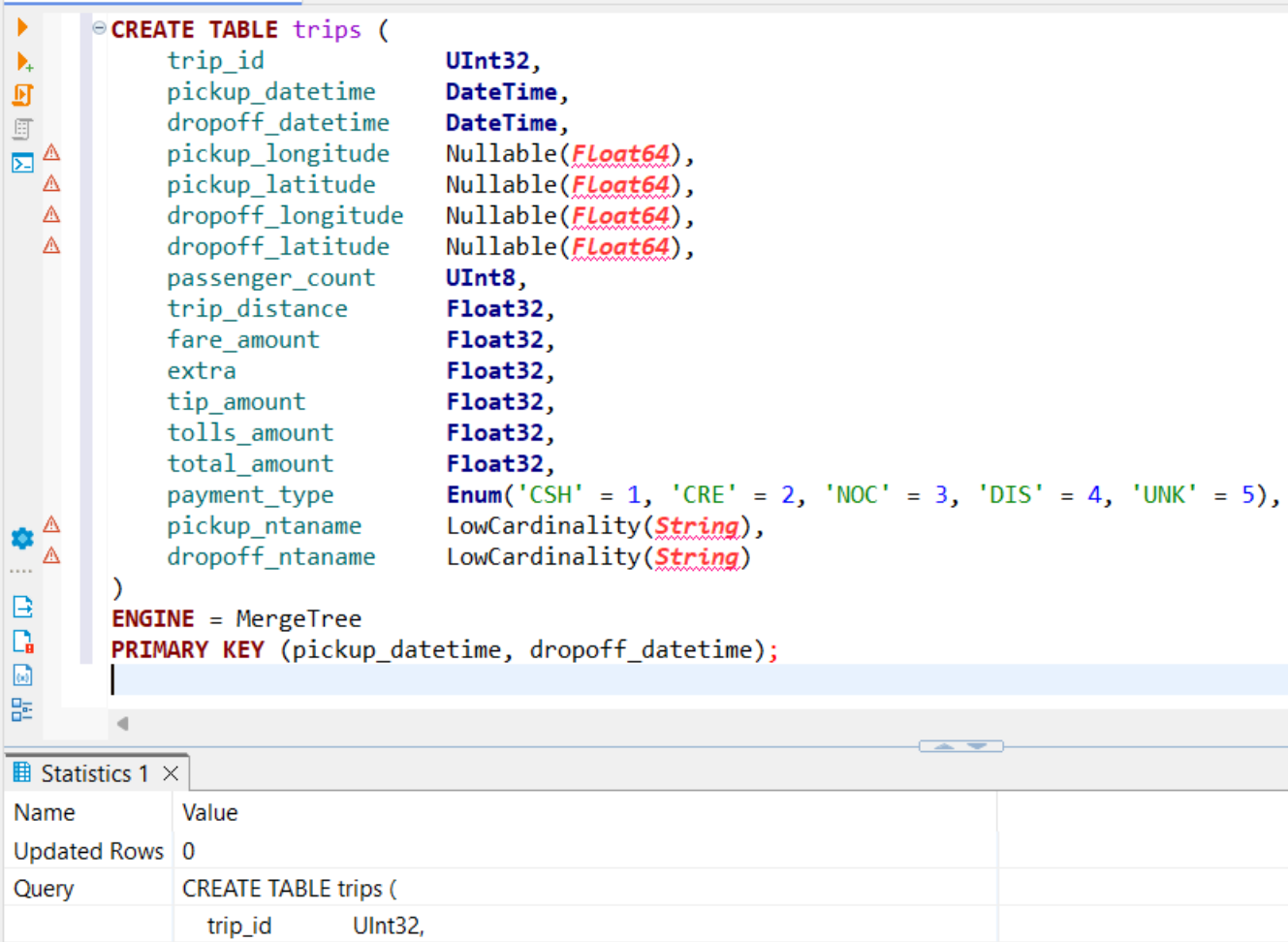
**pickup\_ntaname LowCardinality(String),**

**dropoff\_ntaname LowCardinality(String)**

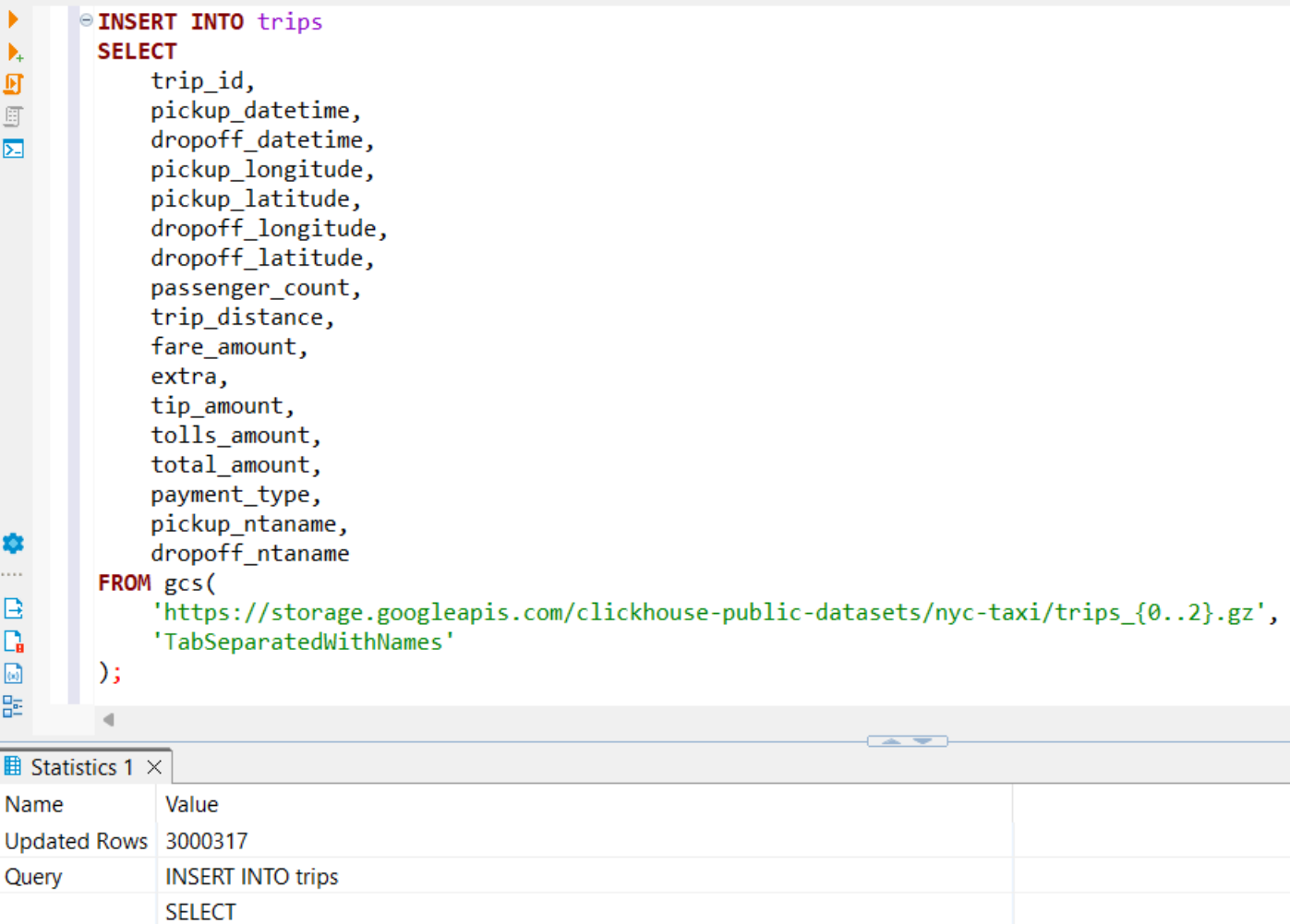
**)**

**ENGINE = MergeTree**

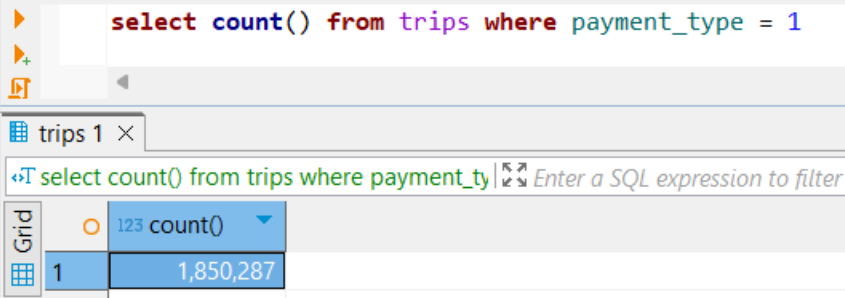
**PRIMARY KEY (pickup\_datetime, dropoff\_datetime);**

****

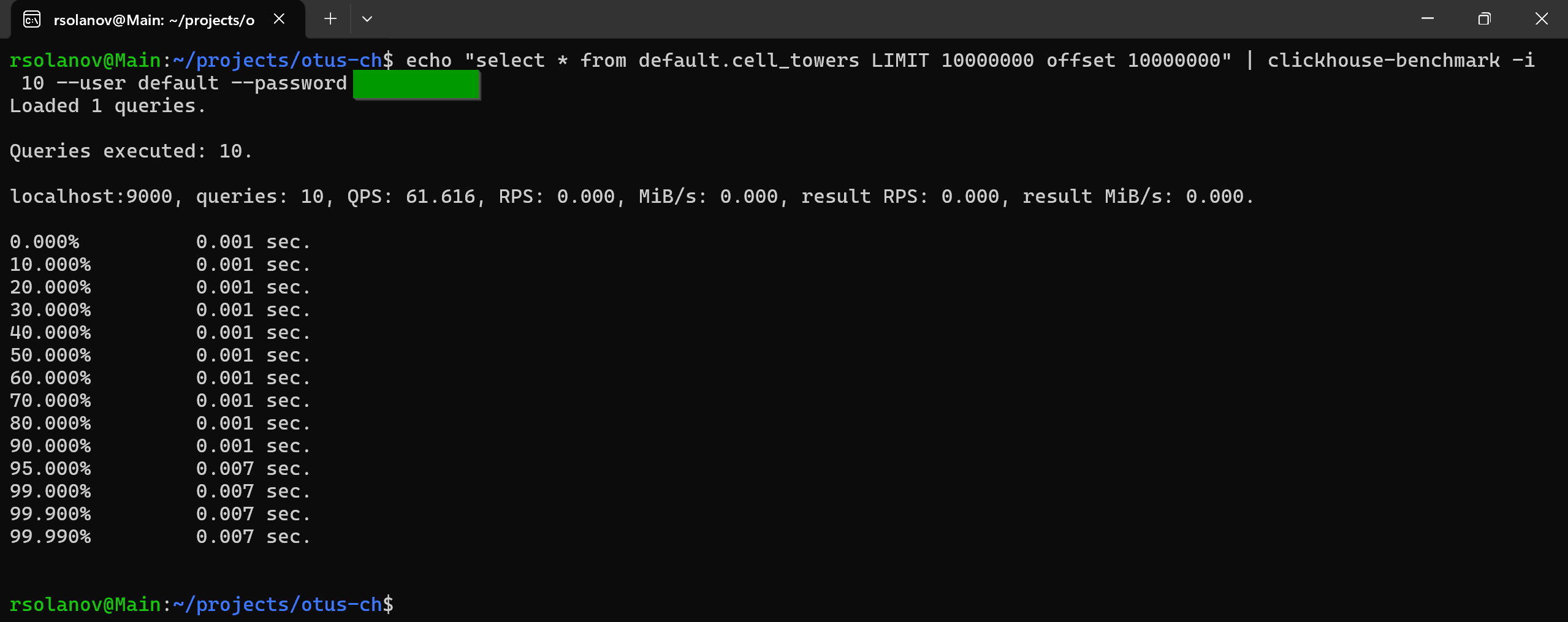
**INSERT INTO trips  
SELECT  
 trip\_id,  
 pickup\_datetime,  
 dropoff\_datetime,  
 pickup\_longitude,  
 pickup\_latitude,  
 dropoff\_longitude,  
 dropoff\_latitude,  
 passenger\_count,  
 trip\_distance,  
 fare\_amount,  
 extra,  
 tip\_amount,  
 tolls\_amount,  
 total\_amount,  
 payment\_type,  
 pickup\_ntaname,  
 dropoff\_ntaname  
FROM gcs(  
 'https://storage.googleapis.com/clickhouse-public-datasets/nyc-taxi/trips\_{0..2}.gz',  
 'TabSeparatedWithNames'  
);**



**select count() from trips where payment\_type = 1**

****

* Провести тестирование производительности и сохранить результаты

****

**Queries executed: 10.**

**localhost:9000, queries: 10, QPS: 61.616, RPS: 0.000, MiB/s: 0.000, result RPS: 0.000, result MiB/s: 0.000.**

**0.000% 0.001 sec.**

**10.000% 0.001 sec.**

**20.000% 0.001 sec.**

**30.000% 0.001 sec.**

**40.000% 0.001 sec.**

**50.000% 0.001 sec.**

**60.000% 0.001 sec.**

**70.000% 0.001 sec.**

**80.000% 0.001 sec.**

**90.000% 0.001 sec.**

**95.000% 0.007 sec.**

**99.000% 0.007 sec.**

**99.900% 0.007 sec.**

**99.990% 0.007 sec.**

**Внесенные изменения:**

* **Увеличение количества открытых файловых дескрипторов: увеличение значения параметра "ulimit -n" не менее чем до 100000;**
* **Увеличение объема разделяемой памяти: увеличение значения параметра "kernel.shmmax" не менее чем до 1 Гб**

**max\_memory\_usage = 8000000000;**

**max\_execution\_time = 1800;**

**force\_primary\_key = 1;**

**max\_threads = 16;**

**max\_insert\_threads = 16;**

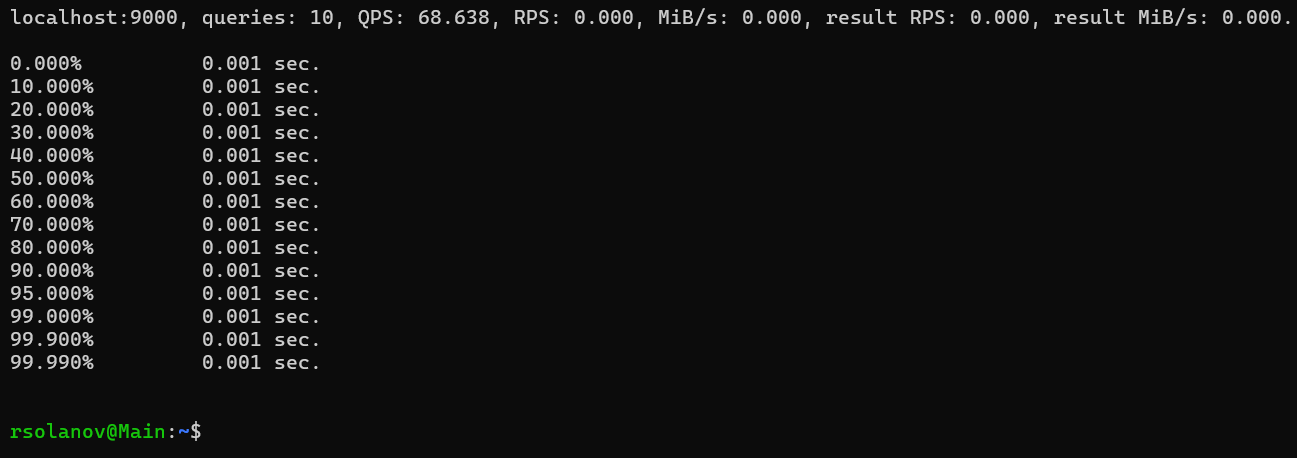
**max\_final\_threads = 16;**

**max\_insert\_block\_size = 268435456; --from 1MB to 256MB**

**max\_compress\_block\_size = 104857600; -- from 1MB to 100MB**

**min\_insert\_block\_size\_rows = 1048449;**

**min\_insert\_block\_size\_bytes = 268435456; -- 256MB**

****

**0.000% 0.001 sec.**

**10.000% 0.001 sec.**

**20.000% 0.001 sec.**

**30.000% 0.001 sec.**

**40.000% 0.001 sec.**

**50.000% 0.001 sec.**

**60.000% 0.001 sec.**

**70.000% 0.001 sec.**

**80.000% 0.001 sec.**

**90.000% 0.001 sec.**

**95.000% 0.001 sec.**

**99.000% 0.001 sec.**

**99.900% 0.001 sec.**

**99.990% 0.001 sec.**

**После применения изменения настроек видно, что увеличился параметр QPS c 61616 до 68638**