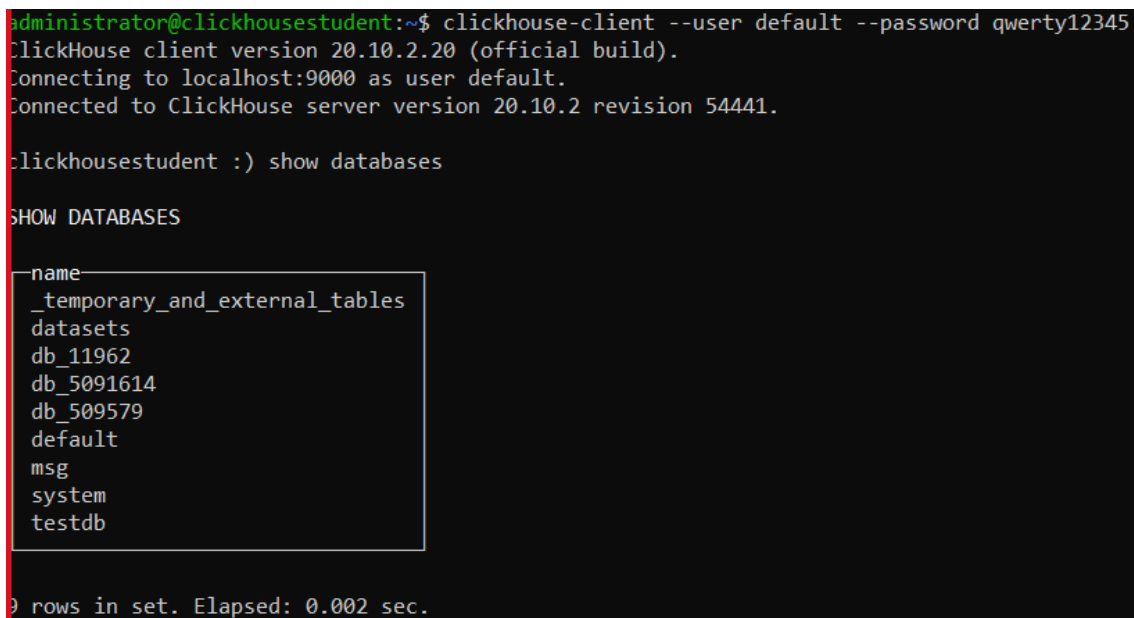


## Лабораторная работа №1 «Работа с ClickHouse»

Выполнили студенты группы ИТ-50916: Кураженкова О.С., Ставских А.Д., Тюленева Е.М.

### 1 Создадим базу данных db\_509579

```
clickhouse-client --user default --password qwerty12345 --query  
"create database db_509579"
```



```
administrator@clickhousestudent:~$ clickhouse-client --user default --password qwerty12345  
ClickHouse client version 20.10.2.20 (official build).  
Connecting to localhost:9000 as user default.  
Connected to ClickHouse server version 20.10.2 revision 54441.  
  
clickhousestudent :) show databases  
  
SHOW DATABASES  
  
+-----+  
| name                                     |  
+-----+  
| _temporary_and_external_tables         |  
| datasets                               |  
| db_11962                               |  
| db_5091614                             |  
| db_509579                              |  
| default                                 |  
| msg                                     |  
| system                                 |  
| testdb                                 |  
+-----+  
9 rows in set. Elapsed: 0.002 sec.
```

Создадим таблицу с необходимой структурой  
(<https://clickhouse.tech/docs/ru/getting-started/example-datasets/metrica/>)

```
clickhouse-client --user default --password qwerty12345 --query  
"CREATE TABLE db_509579.hits_v1 ( WatchID UInt64, JavaEnable  
UInt8, Title String, /*...*/ RequestNum UInt32, RequestTry  
UInt8) ENGINE = MergeTree() PARTITION BY toYYYYMM(EventDate)  
ORDER BY (CounterID, EventDate, intHash32(UserID)) SAMPLE BY  
intHash32(UserID) SETTINGS index_granularity = 8192"
```

### 2 Загрузим первые 10000 строк из файла hits\_v1.tsv в таблицу

```
head --lines=10000 hits_v1.tsv | clickhouse-client --user  
default --password qwerty12345 --query "INSERT INTO  
db_509579.hits_v1 FORMAT TSV" --max_insert_block_size=100000
```

Зайдём в ClickHouse и проверим число строк в таблице

```
clickhouse-client --user default --password qwerty12345  
select count(*) from db_509579.hits_v1
```

```

clickhousestudent :) select count(*) from db_509579.hits_v1

SELECT count(*)
FROM db_509579.hits_v1

count()
10000

1 rows in set. Elapsed: 0.006 sec.

```

Загрузим последние 10000 строк из файла hits\_v1.tsv в таблицу

```

tail --lines=10000 hits_v1.tsv | clickhouse-client --user
default --password qwerty12345 --query "INSERT INTO
db_509579.hits_v1 FORMAT TSV" --max_insert_block_size=100000

```

Проверим количество строк в таблице

```

select count(*) from db_509579.hits_v1

```

```

clickhousestudent :) select count(*) from db_509579.hits_v1

SELECT count(*)
FROM db_509579.hits_v1

count()
20000

1 rows in set. Elapsed: 0.003 sec.

clickhousestudent :)

```

3      Просмотрим структуру созданной таблицы – структура совпадает с SQL-запросом создания таблицы

```
clickhousestudent :) describe table db_509579.hits_v1
```

```
DESCRIBE TABLE db_509579.hits_v1
```

name	type	default_type	default_expression	comment	codec_expression	ttl_expression
WatchID	UInt64					
JavaEnable	UInt8					
Title	String					
GoodEvent	Int16					
EventTime	DateTime					
EventDate	Date					
CounterID	UInt32					
ClientIP	UInt32					
ClientIP6	FixedString(16)					
RegionID	UInt32					
UserID	UInt64					
CounterClass	Int8					
OS	UInt8					
UserAgent	UInt8					
URL	String					
Referer	String					
URLDomain	String					
RefererDomain	String					
Refresh	UInt8					
IsRobot	UInt8					
RefererCategories	Array(UInt16)					
URLCategories	Array(UInt16)					
URLRegions	Array(UInt32)					
RefererRegions	Array(UInt32)					
ResolutionWidth	UInt16					
ResolutionHeight	UInt16					
ResolutionDepth	UInt8					
FlashMajor	UInt8					
FlashMinor	UInt8					
FlashMinor2	String					
NetMajor	UInt8					
NetMinor	UInt8					
UserAgentMajor	UInt16					
UserAgentMinor	FixedString(2)					
CookieEnable	UInt8					
JavascriptEnable	UInt8					
IsMobile	UInt8					
MobilePhone	UInt8					
MobilePhoneModel	String					
Params	String					
IPNetworkID	UInt32					
TrafficSourceID	Int8					

IPNetworkID	UInt32
TrafficSourceID	Int8
SearchEngineID	UInt16
SearchPhrase	String
AdvEngineID	UInt8
IsArtificial	UInt8
WindowClientWidth	UInt16
WindowClientHeight	UInt16
ClientTimeZone	Int16
ClientEventTime	DateTime
SilverlightVersion1	UInt8
SilverlightVersion2	UInt8
SilverlightVersion3	UInt32
SilverlightVersion4	UInt16
PageCharset	String
CodeVersion	UInt32
IsLink	UInt8
IsDownload	UInt8
IsNotBounce	UInt8
FUniqID	UInt64
HID	UInt32
IsOldCounter	UInt8
IsEvent	UInt8
IsParameter	UInt8
DontCountHits	UInt8
WithHash	UInt8
HitColor	FixedString(1)
UTCEventTime	DateTime
Age	UInt8
Sex	UInt8
Income	UInt8
Interests	UInt16
Robotness	UInt8
GeneralInterests	Array(UInt16)
RemoteIP	UInt32
RemoteIP6	FixedString(16)
WindowName	Int32
OpenerName	Int32
HistoryLength	Int16
BrowserLanguage	FixedString(2)
BrowserCountry	FixedString(2)
SocialNetwork	String
SocialAction	String
HTTPError	UInt16
SendTiming	Int32
DNSTiming	Int32
ConnectTiming	Int32
ResponseStartTiming	Int32
ResponseEndTiming	Int32

ResponseStartTiming	Int32
ResponseEndTiming	Int32
FetchTiming	Int32
RedirectTiming	Int32
DOMInteractiveTiming	Int32
DOMContentLoadedTiming	Int32
DOMCompleteTiming	Int32
LoadEventStartTiming	Int32
LoadEventEndTiming	Int32
NSToDOMContentLoadedTiming	Int32
FirstPaintTiming	Int32
RedirectCount	Int8
SocialSourceNetworkID	UInt8
SocialSourcePage	String
ParamPrice	Int64
ParamOrderID	String
ParamCurrency	FixedString(3)
ParamCurrencyID	UInt16
GoalsReached	Array(UInt32)
OpenstatServiceName	String
OpenstatCampaignID	String
OpenstatAdID	String
OpenstatSourceID	String
UTMSource	String
UTMMedium	String
UTMCampaign	String
UTMContent	String
UTMTerm	String
FromTag	String
HasGCLID	UInt8
RefererHash	UInt64
URLHash	UInt64
CLID	UInt32
YCLID	UInt64
ShareService	String
ShareURL	String
ShareTitle	String
ParsedParams.Key1	Array(String)
ParsedParams.Key2	Array(String)
ParsedParams.Key3	Array(String)
ParsedParams.Key4	Array(String)
ParsedParams.Key5	Array(String)
ParsedParams.ValueDouble	Array(Float64)
IslandID	FixedString(16)
RequestNum	UInt32
RequestTry	UInt8

133 rows in set. Elapsed: 0.003 sec.

В SQL-запросе создавалась секция по EventDate. Также можно создать секции по RegionID, OS, Age. Это позволит просматривать строки базы, сгруппированные по Стране/ОС/Возрасту пользователя.

#### 4 Узнаем размер таблицы hits\_v1

```
SELECT formatReadableSize(sum(bytes)) AS size, sum(rows) AS rows
FROM system.parts WHERE database='db_509579' and table='hits_v1'
```

```
clickhousestudent :) SELECT formatReadableSize(sum(bytes)) AS size, sum(rows) AS rows FROM system.parts WHERE active and database='db_509579' and table='hits_v1'
SELECT
  formatReadableSize(sum(bytes)) AS size,
  sum(rows) AS rows
FROM system.parts
WHERE active AND (database = 'db_509579') AND (table = 'hits_v1')

```

size	rows
3.06 MiB	20000

```
1 rows in set. Elapsed: 0.006 sec.
clickhousestudent :)
```

Число строк соответствует числу загруженных строк

Загрузим ещё 10000 строк, пропустив первые 10000

```
head --lines=20000 hits_v1.tsv | tail --lines=10000 |
clickhouse-client --user default --password qwerty12345 --query
"INSERT INTO db_509579.hits_v1 FORMAT TSV" --
max_insert_block_size=100000
```

Посмотрим размер таблицы – объём данных увеличился на 1.19 MB

```
clickhousestudent :) SELECT formatReadableSize(sum(bytes)) AS size, sum(rows) AS rows FROM system.parts WHERE active and database='db_509579'
' and table='hits_v1'
SELECT
  formatReadableSize(sum(bytes)) AS size,
  sum(rows) AS rows
FROM system.parts
WHERE active AND (database = 'db_509579') AND (table = 'hits_v1')

```

size	rows
4.25 MiB	30000

```
1 rows in set. Elapsed: 0.008 sec.
```

Сохраним эти же строки в файл и проверим размер файла

```
head --lines=20000 hits_v1.tsv | tail --lines=10000 >
.509579.txt
ls -all --human-readable | grep 509579
```

```
administrator@clickhousestudent:~$ head --lines=20000 hits_v1.tsv | tail --lines=10000 > .509579.txt
administrator@clickhousestudent:~$ ls -all --human-readable | grep 509579
-rw-rw-r-- 1 administrator administrator 11M Nov 11 10:35 .509579.txt
```

Размер строк в файле больше размера строк в таблице в  $\approx 10$  раз. Это связано с тем, что числа в файле .tsv хранятся в строковом формате, и все значения строк разделены друг от друга символами табуляции.