* Создать новую базу данных и перейти в нее

Создание новой БД

**create database education engine = atomic**

* Создать таблицу для бизнес-кейса "Меню ресторана" с 5+ полями, наполнить ее данными. Обязательно указывать, где нужно, модификаторы Nullable, LowCardinality и пр. Добавить комментарии
  + Создание новой базы данных

**create database education engine = atomic**

* + Удаление таблицы в случае ее наличия

**drop table if exists item;**

* + Создание таблицы меню ресторана

**create table item**

**(**

**item\_key Int32 not null comment 'Уникальный, суррогатный ключ меню',**

**item\_name String not null comment 'Наименование блюда',**

**item\_category\_code LowCardinality(String) not null comment 'Код категории блюда (напитки, закузки, салаты, супы, основное блюдо)',**

**item\_price Decimal(10, 2) not null comment 'Цена блюда',**

**item\_description String not null comment 'Подробное описание блюда',**

**mt\_insert\_dt DateTime not null comment 'Дата-время добавления блюда в меню',**

**mt\_update\_dt DateTime not null comment 'Дата-время обновления блюда в меню'**

**)**

**engine = MergeTree()**

**order by (item\_category\_code, item\_key)**

**partition by toYYYYMM(mt\_insert\_dt)**

**comment 'Таблица меню ресторана'**

* **Протестировать CRUD на созданной таблице**
  + Добавление записи

**insert into item values**

**(1, 'Чесночный хлеб', 'Закузки', 'Свежеиспеченный хлеб с чесночным маслом', 500.99, '2024-10-19 11:00:00', '2023-10-19 12:00:00');**

* + Получение записи

**select \* from item where item\_key = 1;**

* + Обновление записи

**ALTER TABLE item**

**UPDATE**

**item\_price = 800.99,**

**mt\_update\_dt = now()**

**WHERE item\_key = 1;**

* + Удаление записи

**ALTER TABLE item**

**DELETE WHERE item\_key = 1;**

* **Добавить несколько новых полей, удалить пару старых**
  + Добавление нового поля

**ALTER TABLE item**

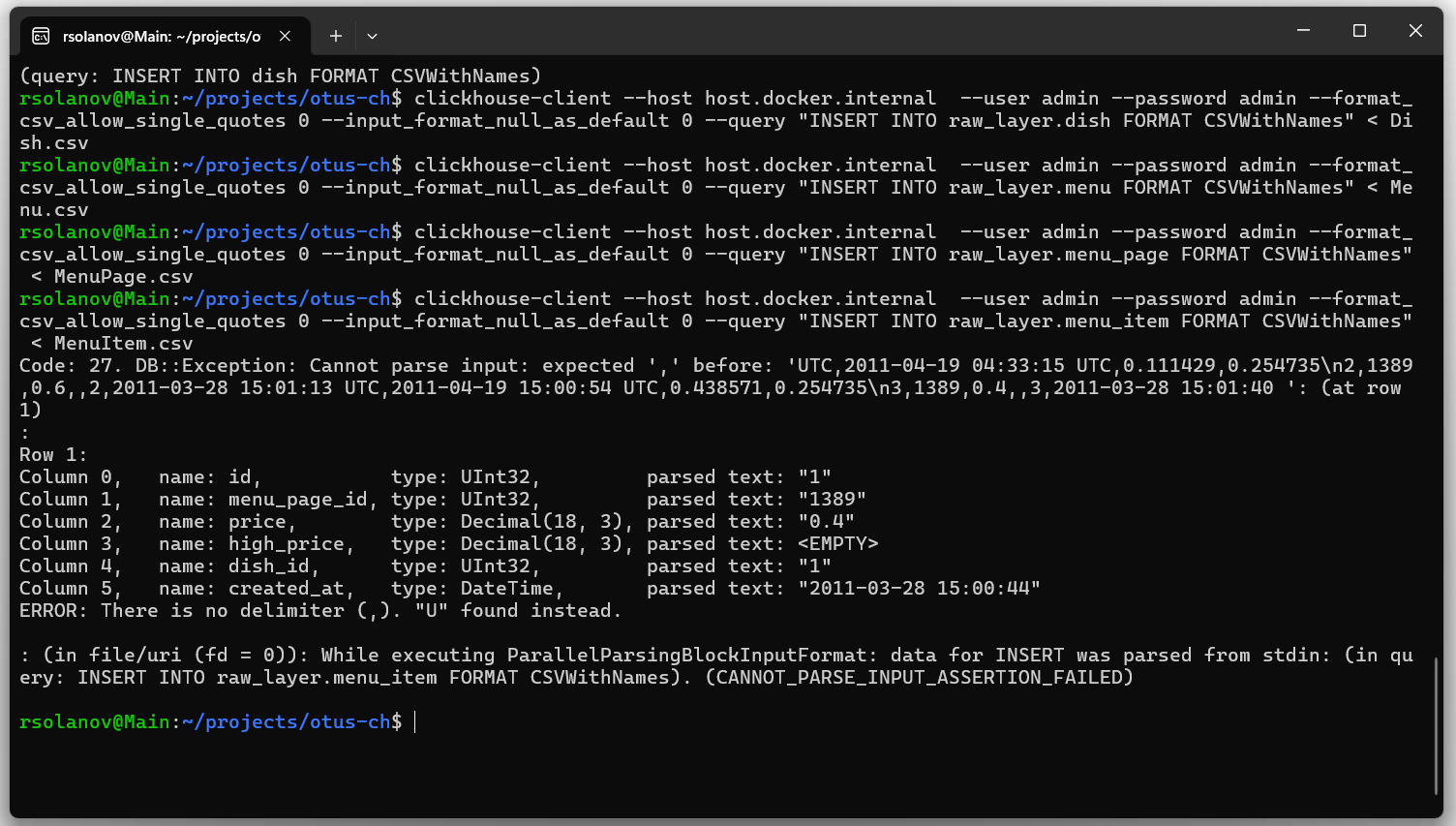
**ADD COLUMN calories UInt16;**

* + Удаление поля

**alter table item**

**drop column calories**

* **Заселектить таблицу (любую) из sample dataset -** [**https://clickhouse.com/docs/en/getting-started/example-datasets/menus**](https://clickhouse.com/docs/en/getting-started/example-datasets/menus)**.**
  + Добавление учебных данных:

****

* + Получение загруженных учебных данных:

**SELECT**

**id,**

**name,**

**description,**

**menus\_appeared,**

**times\_appeared,**

**first\_appeared,**

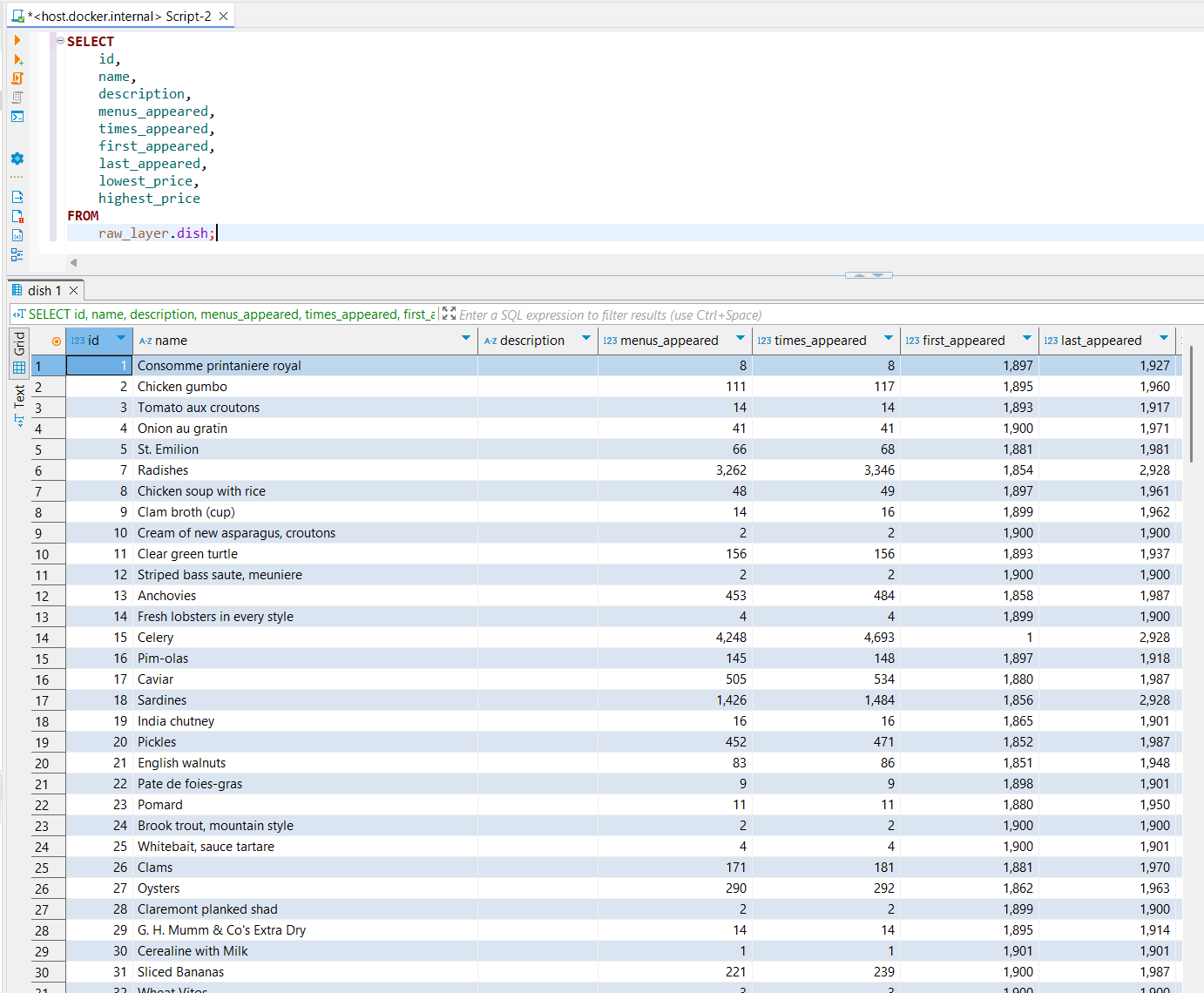
**last\_appeared,**

**lowest\_price,**

**highest\_price**

**FROM**

**raw\_layer.dish;**

****

* **Материализовать таблицу из п.5 (в виде таблицы)**

Запрос создания новой таблицы, которая будет содержать часть информации:

**create table dish\_export**

**engine = mergetree**

**order by id as**

**select**

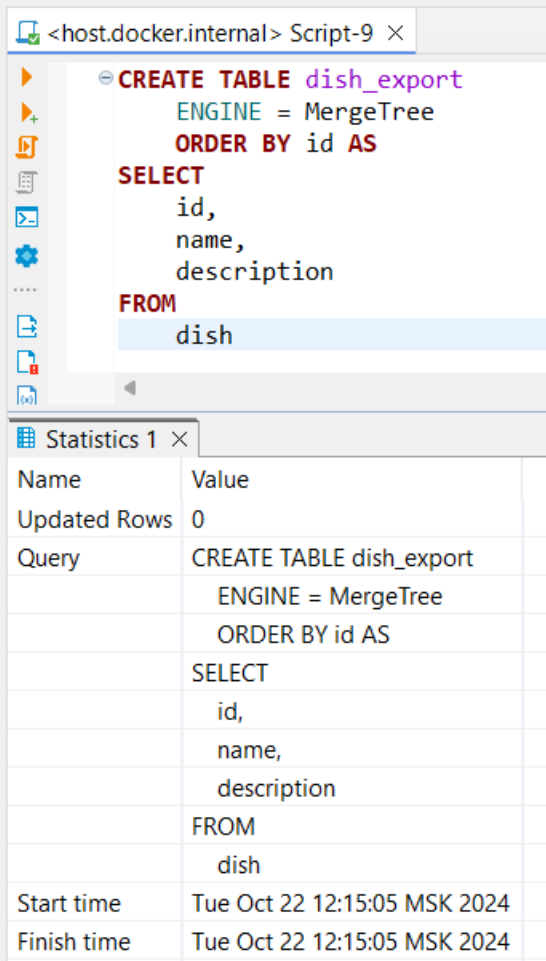
**id,**

**name,**

**description**

**from**

**dish;**

****

* **Поработать с партами. Сделать attach/detach/drop. Добавить данных в первоначально созданную таблицу**

**Алексей Железной в личном сообщении уточнил задание, сказав сделать следующее: «Сделать detach парты, найти ее в системной таблице с открепленными партами, потом вернуть эту парту на место»**

* + Добавление данных в изначально созданную таблицу:

**insert into dish\_export**

**select**

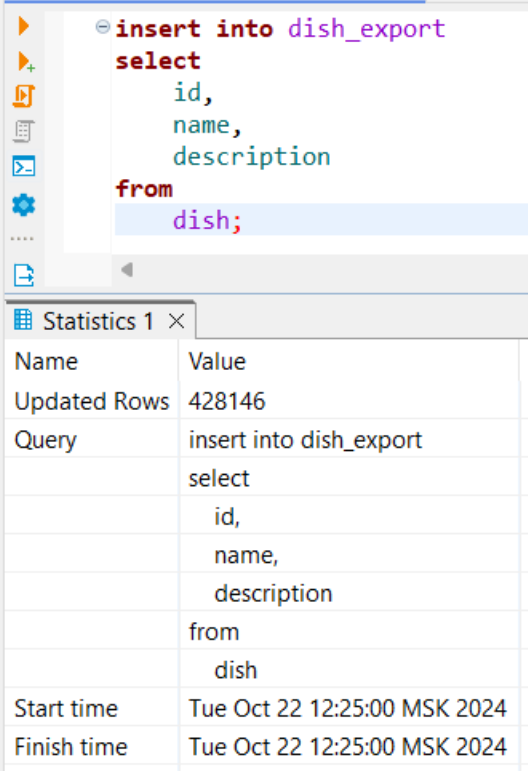
**id,**

**name,**

**description**

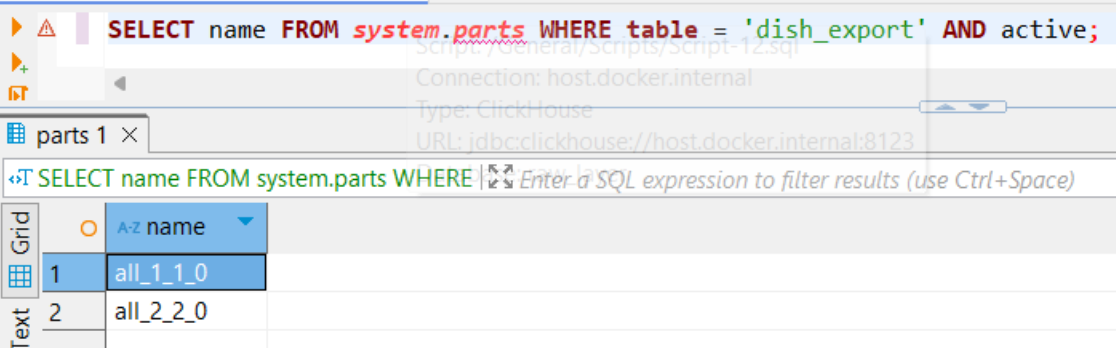
**from**

**dish;**



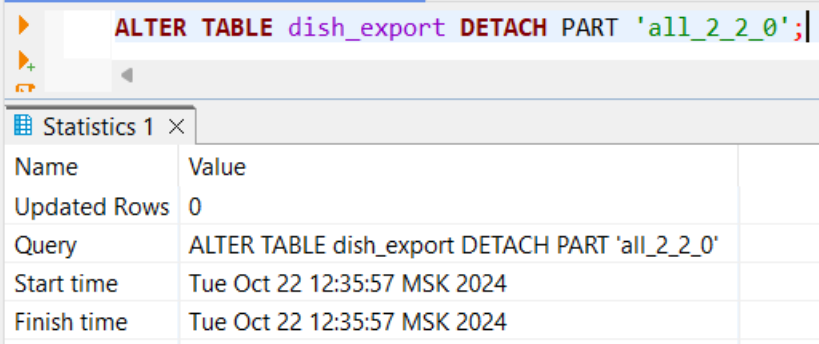
* + Получение имен существующих партов таблицы «dish\_export»:

**select name from system.parts where table = 'dish\_export' and active;**

****

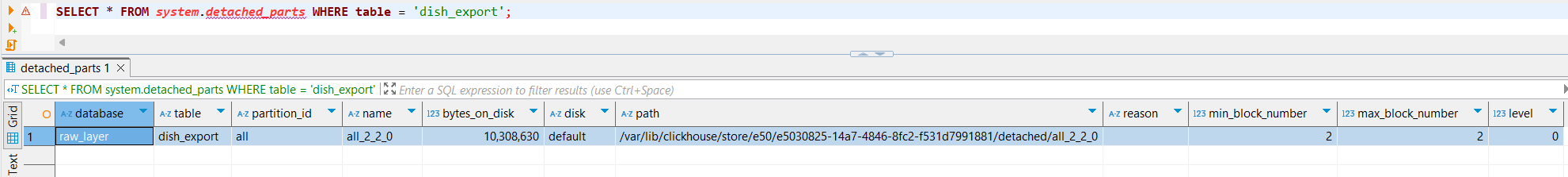
* + Открепление парты:

**alter table dish\_export detach part '** **all\_2\_2\_0';**

****

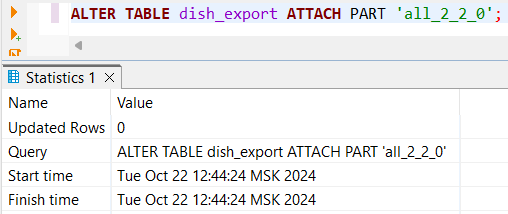
* + Поиск открепленной парты в системной таблице:

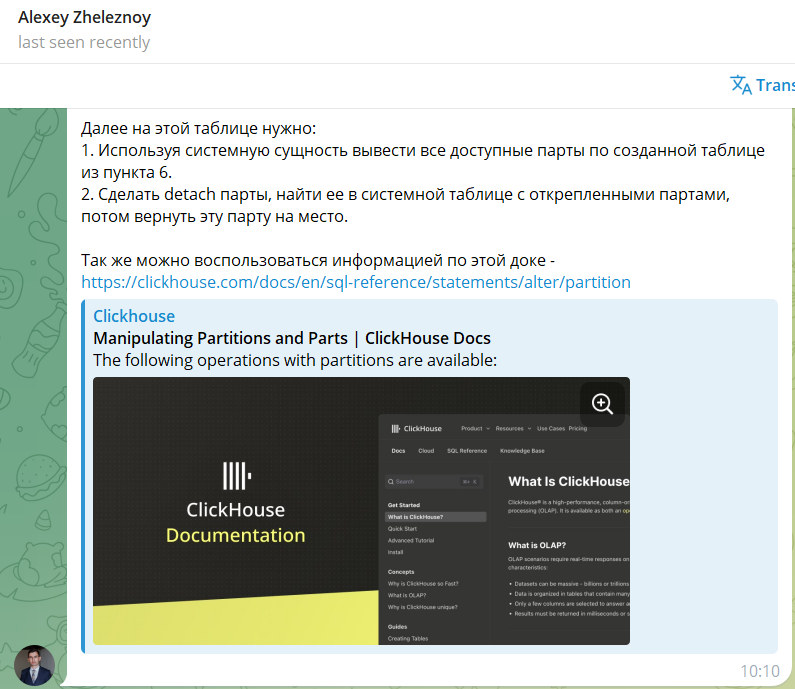
**select \* from system.detached\_parts where table = 'dish\_export';**

****

* + Прикрепление парты обратно:

**alter table dish\_export attach part 'all\_2\_2\_0';**

****

****