```
Use ".open FILENAME" to reopen on a persistent database.
D CREATE TABLE customer (name STRING);
D INSERT INTO customer VALUES ('John');
D INSERT INTO customer VALUES ('Alice');
D INSERT INTO customer VALUES ('Bob');
D
D SELECT * FROM customer;
┌─────────┐
│  name   │
│ varchar │
├─────────┤
│ John    │
│ Alice   │
│ Bob     │
└─────────┘
D .schema customer
CREATE TABLE customer("name" VARCHAR);;
D
```

Рисунок 1. Выполнение общего задания 1

```
D .timer on
D select count(*) from read_csv_auto('C:/Users/super/OneDrive/Рабочий стол/DI/ВУЗЬ/DuckDB/lab_20/city.csv');
┌──────────────┐
│ count_star() │
│    int64     │
├──────────────┤
│         1117 │
└──────────────┘
Run Time (s): real 0.035 user 0.046875 sys 0.000000
D
```

Рисунок 2. Выполнение общего задания 2

```
>     SELECT * FROM customer
>     UNION ALL
>     SELECT COUNT(*) FROM read_csv_auto('C:/Users/super/OneDrive/Рабочий стол/DI/ВУЗЬ/DuckDB/lab_20/city.csv')
> ) TO 'C:/Users/super/OneDrive/Рабочий стол/DI/ВУЗЬ/DuckDB/lab_20/rez1.csv' (FORMAT CSV, HEADER);
Run Time (s): real 0.054 user 0.062500 sys 0.015625
D
```

Рисунок 3. Выполнение общего задания 3

```
D select timezone, count(city) as city_count from read_csv_auto('./city.csv')
> where federal_district in ('Приволжский','Сибирский')
> group by timezone
> order by timezone ASC;
┌──────────┬────────────┐
│ timezone │ city_count │
│ varchar  │   int64    │
├──────────┼────────────┤
│ UTC+3    │        101 │
│ UTC+4    │         41 │
│ UTC+5    │         58 │
│ UTC+6    │          6 │
│ UTC+7    │         86 │
│ UTC+8    │         22 │
└──────────┴────────────┘
D
```

Рисунок 4. Выполнение общего задания 4

```
D create table city as select * from read_csv_auto('./city.csv');
D WITH geo_las AS (
>     SELECT geo_lat AS geo_las FROM city WHERE city = 'Самара'
> ),
> geo_los AS (
>     SELECT geo_lon AS geo_los FROM city WHERE city = 'Самара'
> ),
> geo_lam AS (
>     SELECT geo_lat AS geo_lam, city FROM city
> ),
> geo_lou AS (
>     SELECT geo_lon AS geo_lou FROM city
> )
> SELECT SQRT((POWER((geo_las.geo_las - geo_lam.geo_lam), 2) + POWER((geo_los.geo_los - geo_lou.geo_lou), 2))) AS
ce, geo_lam.city
> FROM geo_las, geo_los, geo_lam, geo_lou
> WHERE geo_lam.city != 'Самара'
> ORDER BY distance ASC
> LIMIT 3;
```

distance double	city varchar
0.0010529999999988604	Заречный
0.009484300000003998	Каменка
0.011993100000005086	Елизово

Рисунок 5. Выполнение общего задания 5

```
D select timezone,
> count(*) city_count
> from city
> group by timezone
> order by city_count desc;
```

timezone varchar	city_count int64
UTC+3	660
UTC+5	173
UTC+7	86
UTC+4	66
UTC+9	31
UTC+8	28
UTC+10	22
UTC+2	22
UTC+11	17
UTC+12	6
UTC+6	6
11 rows	2 columns

```
D
```

Рисунок 7. Выполнение общего задания 6