Practice №1. Distributed table in ClickHouse. Accessing files in S3 via ClickHouse.

***Ex. 1.*** ***Creating a distributed table in ClickHouse cluster.***

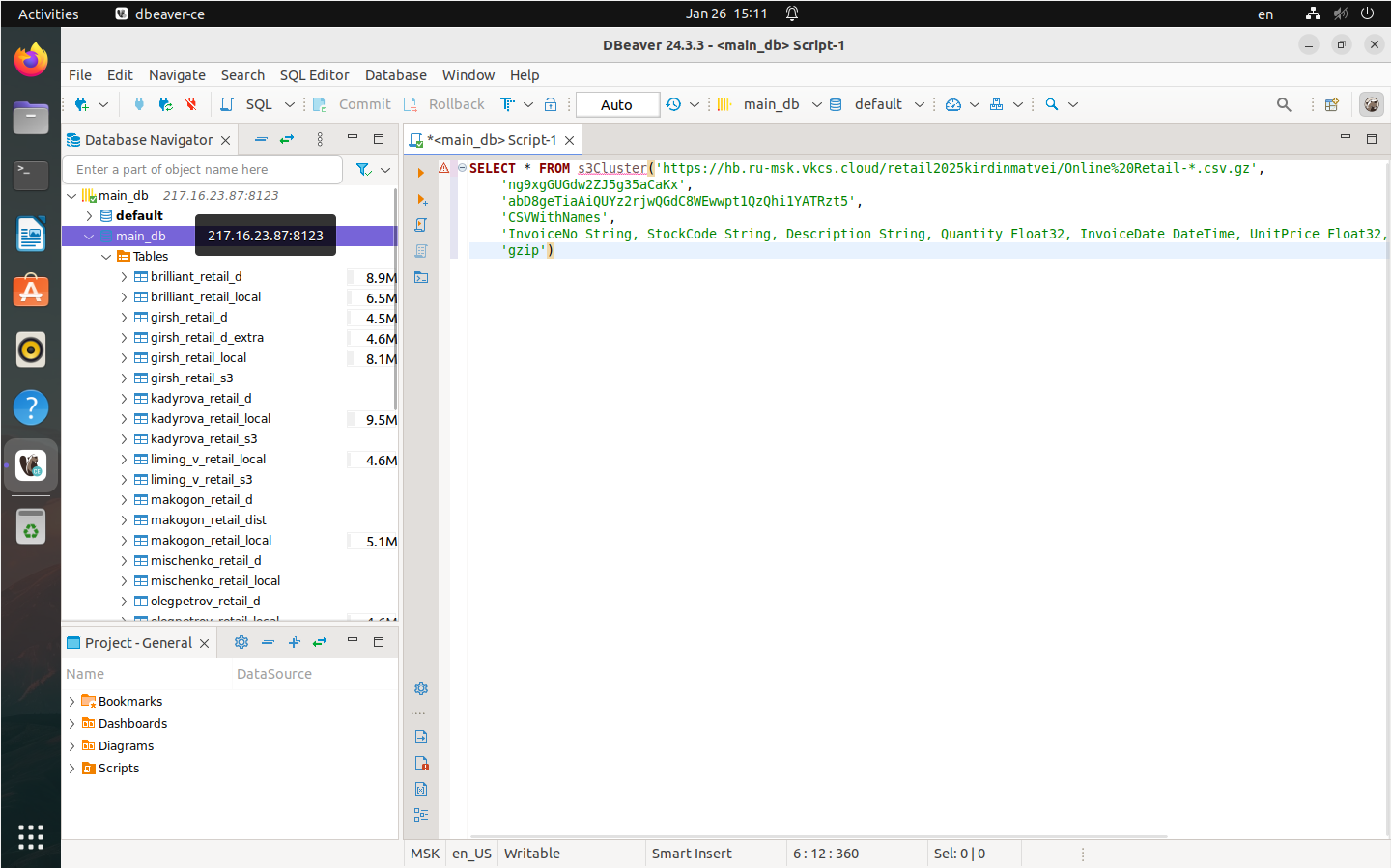
**For this, first we need to connect to the cluster. After connecting, we can see all the tables in its database, as seen on the figure 1.

Figure 1. Database Navigator in DBeaver after a successful connection to the cluster

After that, we use VK Cloud to store our data. This is done by creating a bucket and populating it with the tables provided for this task, as seen on figures 2 and 3.

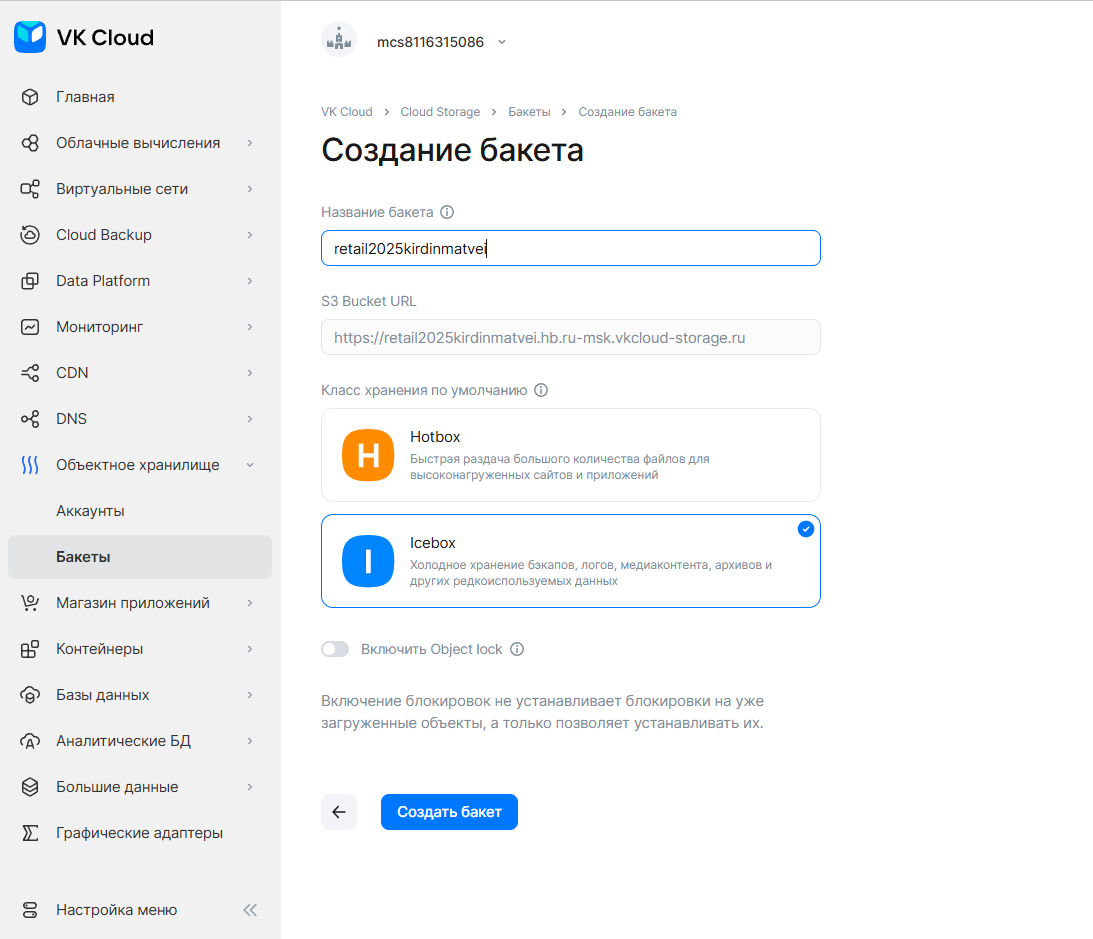
Figure 2. Bucket creation interface in VK Cloud

Figure 3. Bucket with the gzip archives of data uploaded

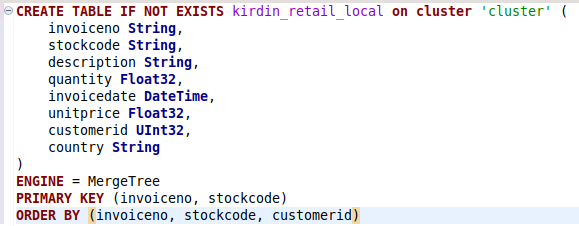
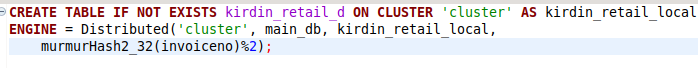
Now we can start creating distributed tables on the ClickHouse cluster. First, we construt two tables with local and distributed and mergre tree engines to serve as distributed and local tables respectively. For local table we run the SQL script given on figure 4.

Figure 4. Script that creates the local table with certain columns

After running this script, we also run the script from figure 5.

Figure 5. Script that creates distributed table, which will house the data

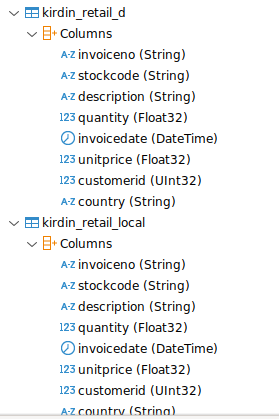
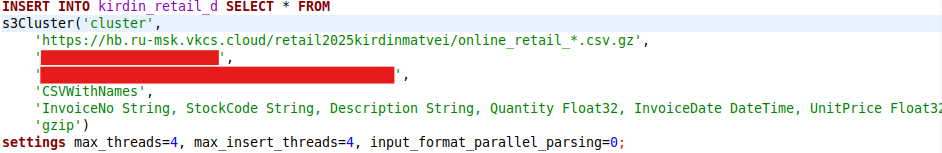
Results can be seen on figure 6. They indicate that both of those table were correctly created.

Figure 6. Database Navigator in DBeaver after successful creation of tables

 With tables in place, we can write the data from cluster to the distributed table by running a script from figure 7.

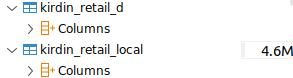
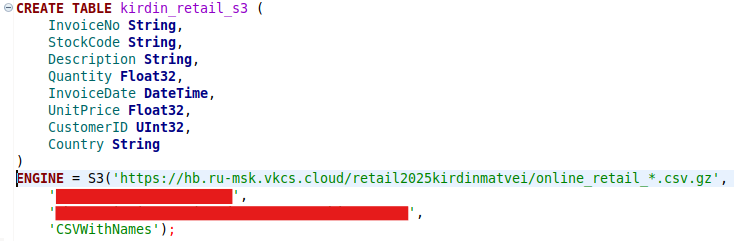
Figure 7. This query inserts all the data from VK Cloud bucket to our table

Figure 8. Changes to the tables we created

As indicated on figure 8, the data indeed did got copied to the local table.

***Ex. 2. Accessing S3 files via ClickHouse without copying data (Data Lake-style manipulations).***

First, we create a table with S3 engine which will not populate a table in DB. Instead, it will provide access to the files in tabular style. This can be done by running a script from figure 9. Its’ results can be seen on the figure 10.

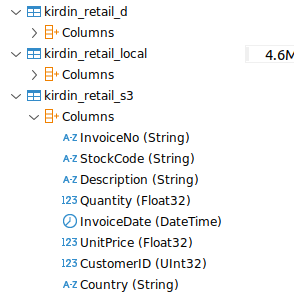
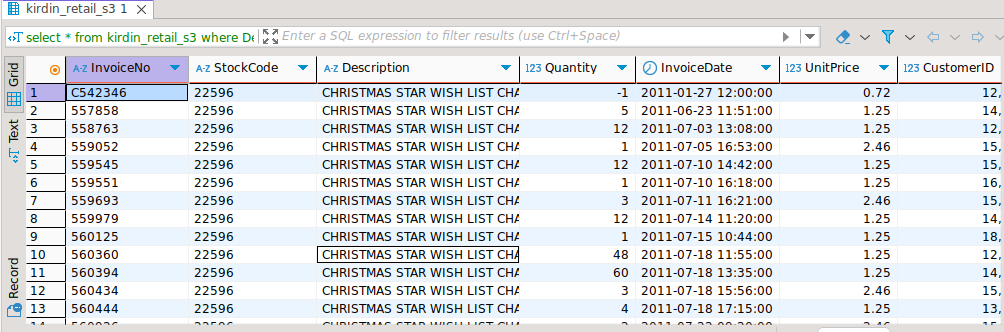
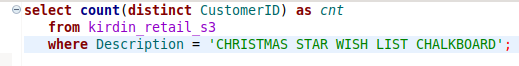
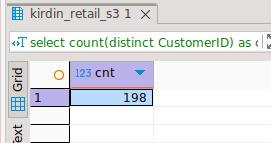
Figure 9. This script creates a table with a specific set of rows and an S3 engine

Figure 10. Database Navigator after a successful creation of S3 table

Now we can write queries to this table. For example, let us see the set of all customers who purchased “CHRISTMAS STAR WISH LIST CHALKBOARD” stocks and the number of them (figures 11 through 13). Let us also write a query for an average transaction value for each unique customer (figures 14 and 15).

Figure 11. A query that returns all the rows with “CHRISTMAS STAR WISH LIST CHALKBOARD” description

Figure 12. The result of query from figure 11

Figure 12. A query that returns the number of distinct customers that bought stocks with “CHRISTMAS STAR WISH LIST CHALKBOARD” description

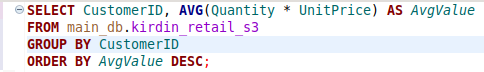
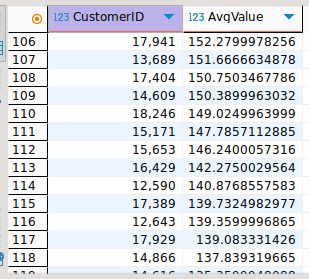
Figure 13. The result of query from figure 13

Figure 14. A query that returns the average transaction value for each customer from most to least.

Figure 15. The result of query from the figure 14