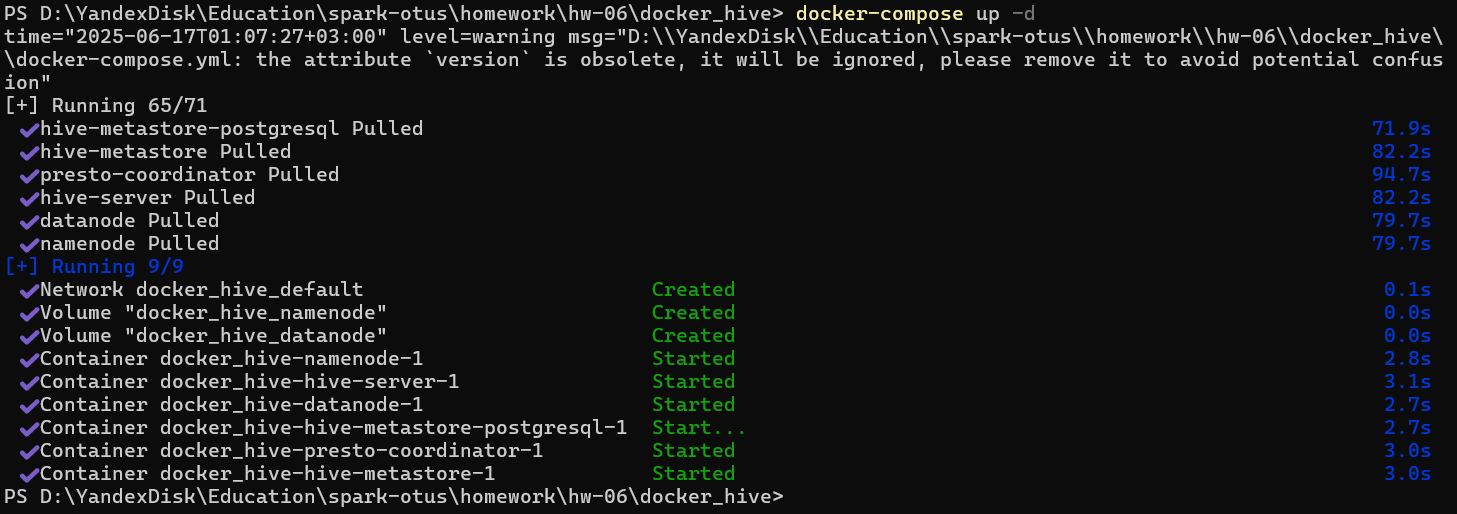
**Установка Hadoop/Hive и подготовка данных источника**

* Установка docker-container

docker-compose up -d

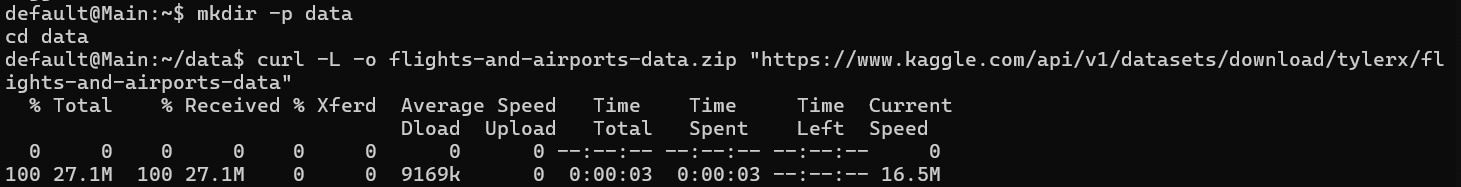


* Копирование данных с Kaggle

mkdir -p data

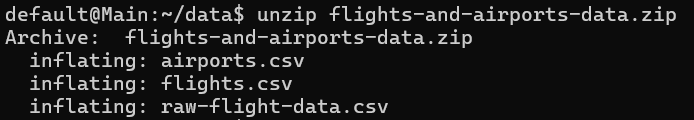
cd data

curl -L -o flights-and-airports-data.zip "https://www.kaggle.com/api/v1/datasets/download/tylerx/flights-and-airports-data"



* Распаковка архива

unzip flights-and-airports-data.zip

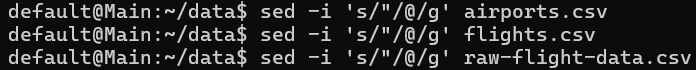


* Исправление кавычек в файлах

sed -i 's/"/@/g' airports.csv

sed -i 's/"/@/g' flights.csv

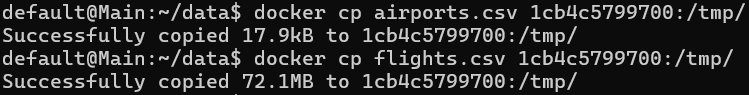
sed -i 's/"/@/g' raw-flight-data.csv



* Копирование файлов в контейнер namenode

docker cp airports.csv 1cb4c5799700:/tmp/

docker cp flights.csv 1cb4c5799700:/tmp/



* Создание директорий в HDFS

docker exec -it 1cb4c5799700 bash

hdfs dfs -mkdir -p /user/hive/warehouse/flight\_analysis.db/airports

hdfs dfs -mkdir -p /user/hive/warehouse/flight\_analysis.db/flights



* Загрузка данных в HDFS

hdfs dfs -put /tmp/airports.csv /user/hive/warehouse/flight\_analysis.db/airports/

hdfs dfs -put /tmp/flights.csv /user/hive/warehouse/flight\_analysis.db/flights/



* Создание БД

**create** **database** **if** **not** **exists** src;

**create** **database** **if** **not** **exists** ods;

**create** **database** **if** **not** **exists** dm;

**Создание таблиц слоя SRC (слой источника)**

* Создание таблицы «src.airports»

**CREATE** **EXTERNAL** **TABLE** src.airports (

airport\_id **INT**,

city **STRING**,

state **STRING**,

name **STRING**

)

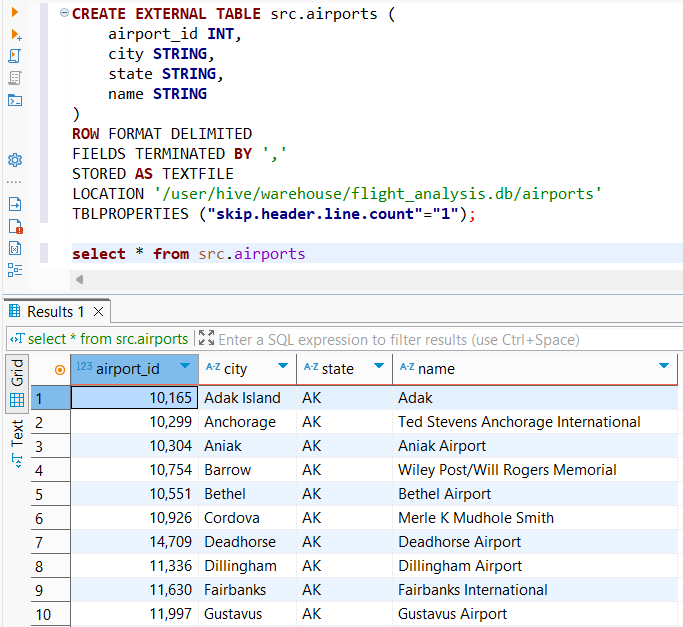
**ROW** FORMAT DELIMITED

FIELDS TERMINATED **BY** ','

STORED **AS** TEXTFILE

LOCATION '/user/hive/warehouse/flight\_analysis.db/airports'

TBLPROPERTIES (**"skip.header.line.count"**=**"1"**);



* Создание таблицы «src.flights»

**CREATE** **EXTERNAL** **TABLE** src.flights (

day\_of\_month **INT**,

day\_of\_week **INT**,

carrier **STRING**,

origin\_airport\_id **INT**,

dest\_airport\_id **INT**,

dep\_delay **INT**,

arr\_delay **INT**

)

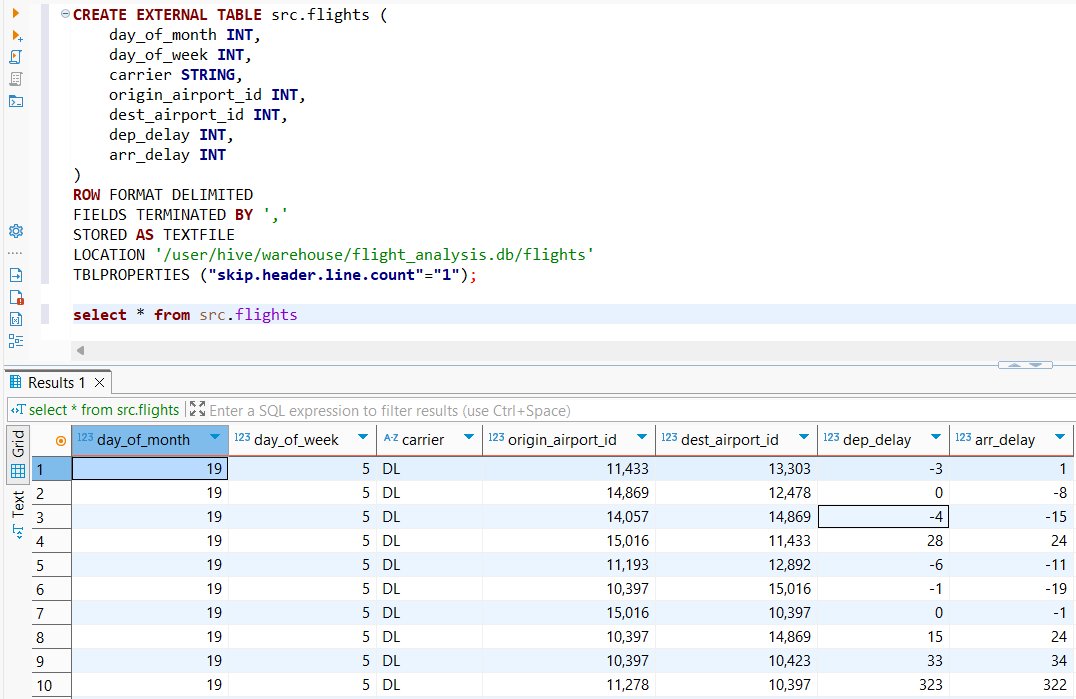
**ROW** FORMAT DELIMITED

FIELDS TERMINATED **BY** ','

STORED **AS** TEXTFILE

LOCATION '/user/hive/warehouse/flight\_analysis.db/flights'

TBLPROPERTIES (**"skip.header.line.count"**=**"1"**);



**Создание таблиц слоя ODS (слой сырых данных)**

* Создание таблицы «ods.airports»

**CREATE** **TABLE** ods.airports

(

airport\_id **INT**,

city **STRING**,

state **STRING**,

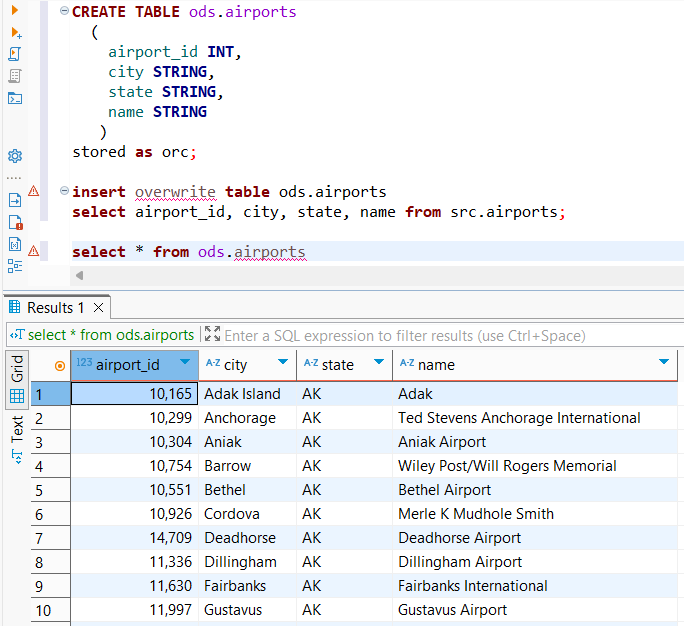
name **STRING**

)

stored **as** orc;

**insert** overwrite **table** ods.airports

**select** airport\_id, city, state, name **from** src.airports;



* Создание таблицы «ods.flights»

**CREATE** **TABLE** ods.flights (

day\_of\_month **INT**,

day\_of\_week **INT**,

carrier **STRING**,

origin\_airport\_id **INT**,

dest\_airport\_id **INT**,

dep\_delay **INT**,

arr\_delay **INT**

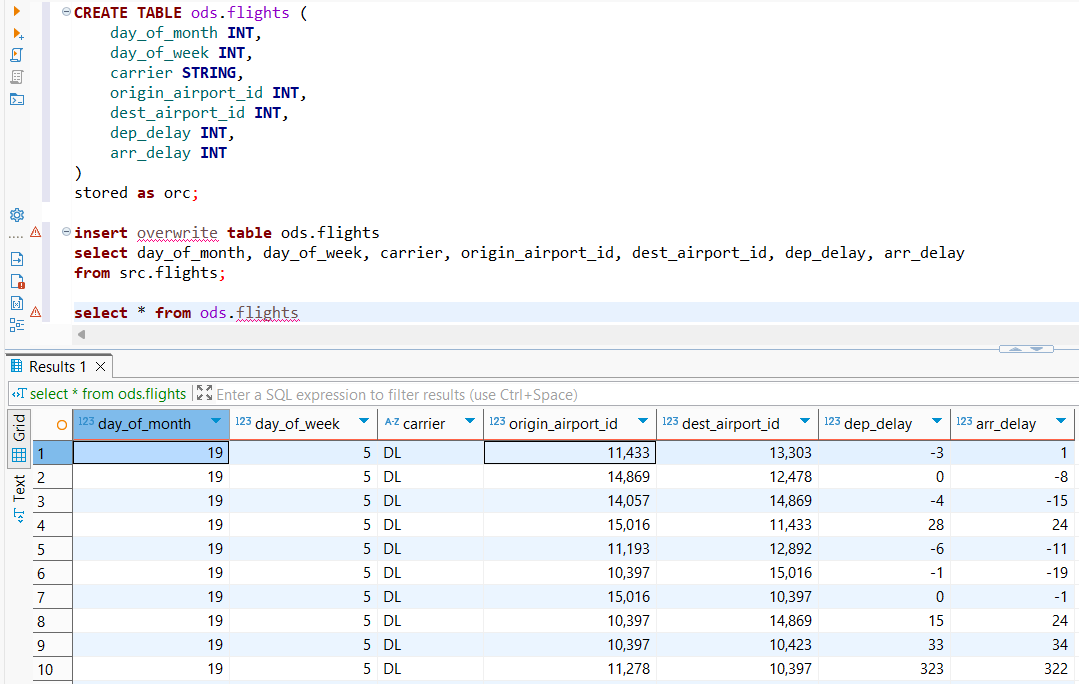
)

stored **as** orc;

**insert** overwrite **table** ods.flights

**select** day\_of\_month, day\_of\_week, carrier, origin\_airport\_id, dest\_airport\_id, dep\_delay, arr\_delay

**from** src.flights;



**Создание витринного слоя DM**

* Создание витрины с количеством рейсов по авиакомпаниям

**CREATE** **TABLE** dm.airline\_flight\_counts (

carrier **string**,

flight\_count **bigint**

)

stored **as** orc;

**insert** overwrite **table** dm.airline\_flight\_counts

**SELECT**

carrier,

**COUNT**(\*) **as** flight\_count

**FROM**

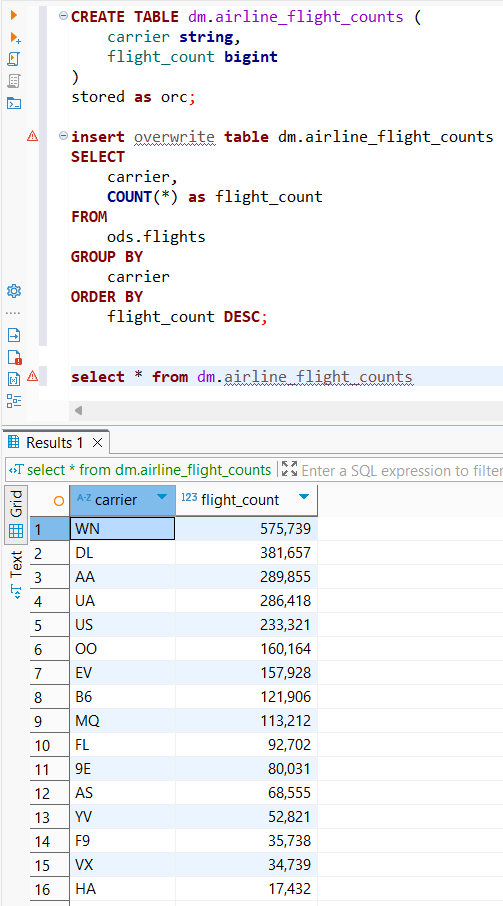
ods.flights

**GROUP** **BY**

carrier

**ORDER** **BY**

flight\_count **DESC**;



* Создание витрины со средней задержкой по аэропортам отправления

**CREATE** **TABLE** dm.avg\_departure\_delay\_by\_airport

(

airport\_id **int**,

airport\_name **string**,

city **string**,

avg\_dep\_delay **decimal**

)

stored **as** orc;

**insert** overwrite **table** dm.avg\_departure\_delay\_by\_airport

**SELECT**

a.airport\_id,

a.name **as** airport\_name,

a.city,

**AVG**(f.dep\_delay) **as** avg\_dep\_delay

**FROM**

ods.flights f

**INNER** **JOIN**

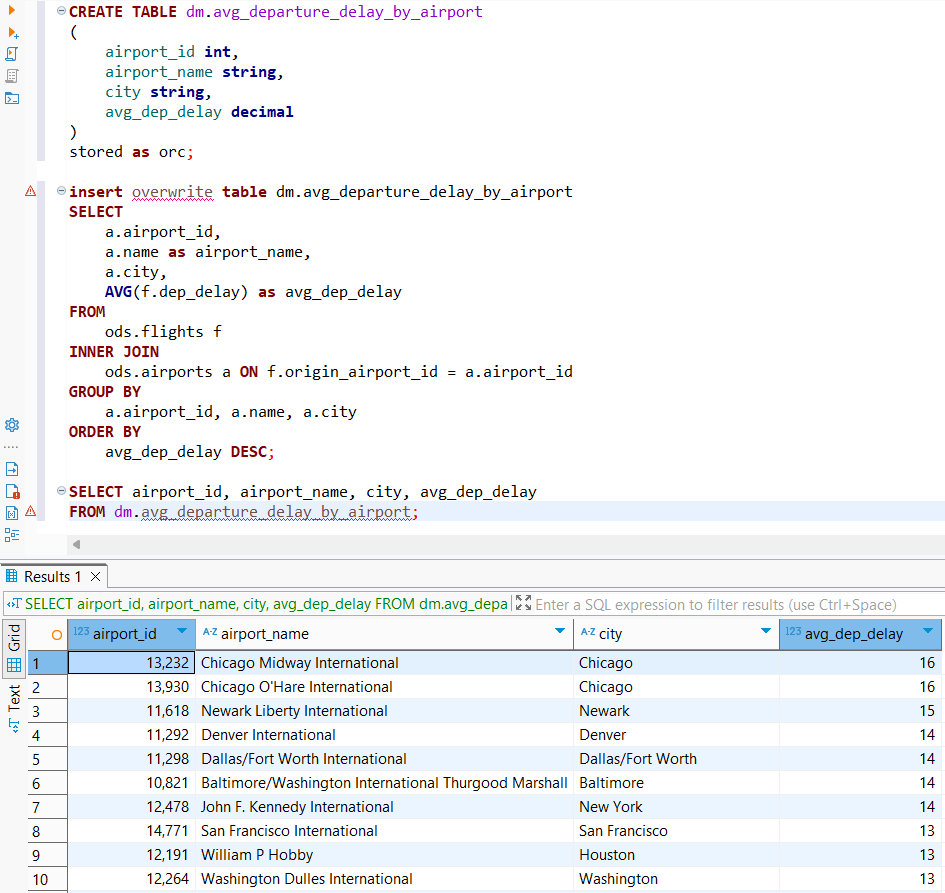
ods.airports a **ON** f.origin\_airport\_id = a.airport\_id

**GROUP** **BY**

a.airport\_id, a.name, a.city

**ORDER** **BY**

avg\_dep\_delay **DESC**;



* Создание витрины с 10-ю самых популярных маршрутов

**CREATE** **TABLE** dm.popular\_routes

(

origin\_airport **string**,

dest\_airport **string**,

flight\_count **bigint**

)

stored **as** orc;

**insert** overwrite **table** dm.popular\_routes

**SELECT**

a1.name **as** origin\_airport,

a2.name **as** dest\_airport,

**COUNT**(\*) **as** flight\_count

**FROM**

ods.flights f

**INNER** **JOIN**

ods.airports a1 **ON** f.origin\_airport\_id = a1.airport\_id

**INNER** **JOIN**

ods.airports a2 **ON** f.dest\_airport\_id = a2.airport\_id

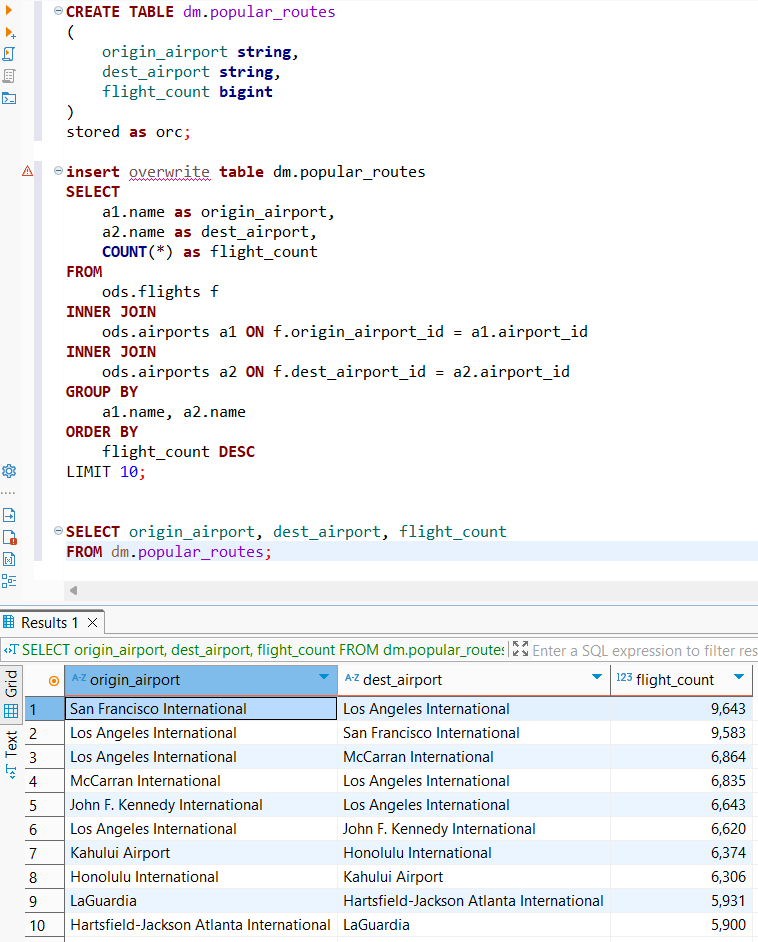
**GROUP** **BY**

a1.name, a2.name

**ORDER** **BY**

flight\_count **DESC**

LIMIT 10;



* Создание витрины с авиакомпаниями, к которых наибольшие задержки

**CREATE** **TABLE** dm.airlines\_with\_delays

(

carrier **string**,

avg\_dep\_delay **decimal**,

avg\_arr\_delay **decimal**,

total\_flights **bigint**

)

stored **as** orc;

**insert** overwrite **table** dm.airlines\_with\_delays

**SELECT**

carrier,

**AVG**(dep\_delay) **as** avg\_dep\_delay,

**AVG**(arr\_delay) **as** avg\_arr\_delay,

**COUNT**(\*) **as** total\_flights

**FROM**

ods.flights

**GROUP** **BY**

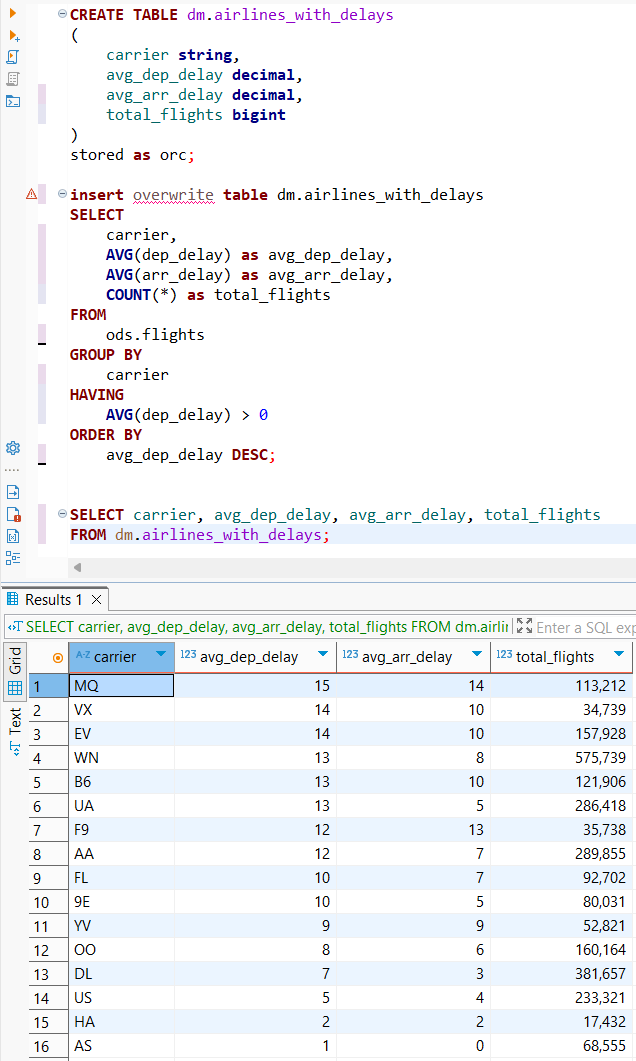
carrier

**HAVING**

**AVG**(dep\_delay) > 0

**ORDER** **BY**

avg\_dep\_delay **DESC**;



* Создание витрины с задержками по дням недели

**CREATE** **TABLE** dm.delays\_by\_weekday

(

day\_of\_week **int**,

avg\_dep\_delay **decimal**,

avg\_arr\_delay **decimal**,

total\_flights **bigint**

)

stored **as** orc;

**insert** overwrite **table** dm.delays\_by\_weekday

**SELECT**

day\_of\_week,

**AVG**(dep\_delay) **as** avg\_dep\_delay,

**AVG**(arr\_delay) **as** avg\_arr\_delay,

**COUNT**(\*) **as** total\_flights

**FROM**

ods.flights

**GROUP** **BY**

day\_of\_week

**ORDER** **BY**

day\_of\_week;

