**HIVE MINI PROJECT 2**

Objective - The assignment is meant for you to apply learnings of the module on Hive on a real-life dataset. One of the major objectives of this assignment is gaining familiarity with how an analysis works in Hive and how you can gain insights from large datasets.

Problem Statement - New York City is a thriving metropolis and just like most other cities of similar size, one of the biggest problems its residents face is parking. The classic combination of a huge number of cars and a cramped geography is the exact recipe that leads to a large number of parking tickets.

In an attempt to scientifically analyse this phenomenon, the NYC Police Department regularly collects data related to parking tickets. This data is made available by NYC Open Data portal. We will try and perform some analysis on this data.

Download Dataset - https://data.cityofnewyork.us/browse?q=parking+tickets

Note: Consider only the year 2017 for analysis and not the Fiscal year.

create table parking

(

summons\_number bigint,

plate\_id string,

registration\_state string,

plate\_type string,

issue\_date string,

violation\_code int,

vehicle\_body\_type string,

vehicle\_make string,

issuing\_agency string,

street\_code1 int,

street\_code2 int,

street\_code3 int,

vehicle\_expiration\_date int,

violation\_location int,

violation\_precinct int,

issuer\_precinct int,

issuer\_code int,

issuer\_command string,

issuer\_squad string,

violation\_time string,

time\_first\_observed string,

violation\_county string,

violation\_front\_opposite string,

house\_number string,

street\_name string,

intersecting\_street string,

date\_first\_observed int,

law\_section int,

sub\_division string,

violation\_legal\_code string,

days\_parking\_in\_effect string,

from\_hours\_in\_effect string,

to\_hours\_in\_effect string,

vehicle\_color string,

unregistered\_vehicle int,

vehicle\_year int,

meter\_number string,

feet\_from\_curb int,

violation\_post\_code string,

violation\_description string,

no\_standing\_or\_stopping\_violation string,

hydrant\_violation string,

double\_parking\_vio string

)

row format delimited

fields terminated by ','

location '/project2/data'

tblproperties("skip.header.line.count"="1");

create table parking\_violation

(

Summons\_Number bigint,

Plate\_ID string,

Registration\_State string,

Plate\_Type string,

Issue\_Date date,

Violation\_Code int,

Vehicle\_Body\_Type string,

Vehicle\_Make string,

Issuing\_Agency string,

Street\_Code1 int,

Street\_Code2 int,

Street\_Code3 int,

Vehicle\_Expiration string,

Violation\_Location int,

Violation\_Precinct int,

Issuer\_Precinct int,

Issuer\_Code int,

Issuer\_Command string,

Issuer\_Squad string,

Violation\_Time string,

Time\_First\_Observed string,

Violation\_In\_Front\_of\_or\_Opposite string,

House\_Number string,

Street\_Name string,

Intersecting\_Street string,

Date\_First\_Observed int,

Law\_Section int,

Sub\_Division string,

Violation\_Legal\_Code string,

Days\_Parking\_In\_Effect string,

From\_Hours\_In\_Effect string,

To\_Hours\_In\_Effect string,

Vehicle\_Color string,

Unregistered\_Vehicle int,

Vehicle\_Year string,

Meter\_Number string,

Feet\_From\_Curb int,

Violation\_Post\_Code string,

Violation\_Description string,

No\_Standing\_or\_Stopping\_Violation string,

Hydrant\_Violation string,

Double\_Parking\_Violation string)

partitioned by (Violation\_County string)

CLUSTERED BY (Violation\_Code) sorted by (Violation\_Code) INTO 8 BUCKETS

row format delimited

fields terminated by ','

tblproperties ("skip.header.line.count" = "1");

set hive.exec.dynamic.partition=true;

set hive.exec.dynamic.partition.mode=nonstrict;

set hive.enforce.bucketing = true;

insert into parking\_violation

partition(Violation\_County) Select

Summons\_Number,

Plate\_ID,

Registration\_State,

Plate\_Type,

Issue\_Date,

Violation\_Code,

Vehicle\_Body\_Type,

Vehicle\_Make,

Issuing\_Agency,

Street\_Code1,

Street\_Code2,

Street\_Code3,

Vehicle\_Expiration\_date,

Violation\_Location,

Violation\_Precinct,

Issuer\_Precinct,

Issuer\_Code,

Issuer\_Command,

Issuer\_Squad,

Violation\_Time,

Time\_First\_Observed,

Violation\_front\_Opposite,

House\_Number,

Street\_Name,

Intersecting\_Street,

Date\_First\_Observed,

Law\_Section,

Sub\_Division,

Violation\_Legal\_Code,

Days\_Parking\_In\_Effect,

From\_Hours\_In\_Effect,

To\_Hours\_In\_Effect,

Vehicle\_Color,

Unregistered\_Vehicle,

Vehicle\_Year,

Meter\_Number,

Feet\_From\_Curb,

Violation\_Post\_Code,

Violation\_Description,

No\_Standing\_or\_Stopping\_Violation,

Hydrant\_Violation,

Double\_Parking\_Vio,

Violation\_County

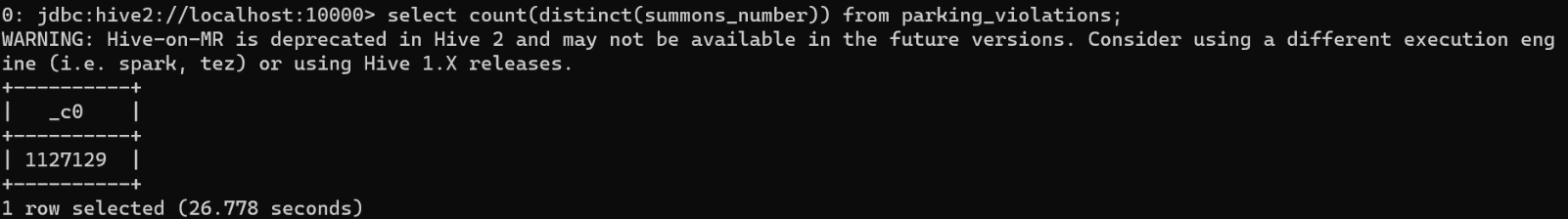
from parking ;

The analysis can be divided into two parts:

Part-I: Examine the data

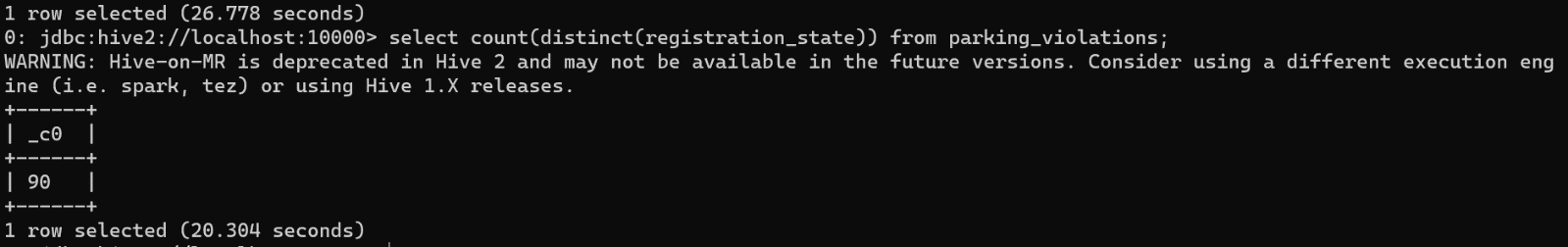
1. Find the total number of tickets for the year.

select count(distinct(summons\_number)) from parking\_violations;



1. Find out how many unique states the cars which got parking tickets came from.

select count(distinct(registration\_state)) from parking\_violations;

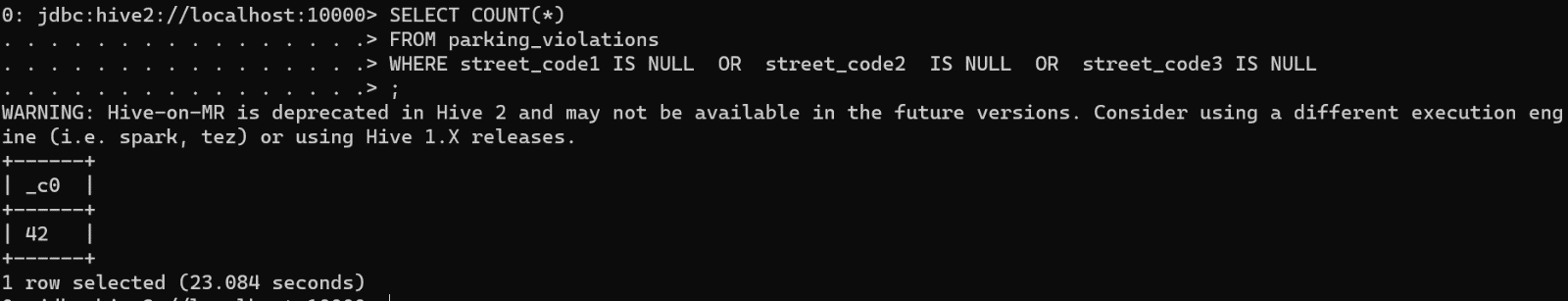


1. Some parking tickets don’t have addresses on them, which is cause for concern. Find out how many such tickets there are(i.e. tickets where either "Street Code 1" or "Street Code 2" or "Street Code 3" is empty )

SELECT COUNT(\*)

FROM parking\_violations

WHERE street\_code1 IS NULL OR street\_code2 IS NULL OR street\_code3 IS NULL;



Part-II: Aggregation tasks

1. How often does each violation code occur? (frequency of violation codes - find the top 5)

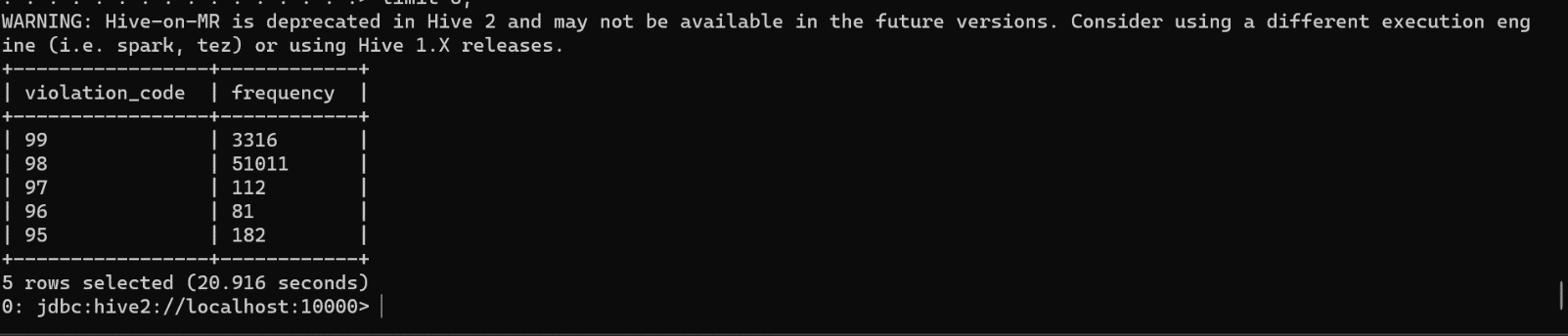
select violation\_code, count(violation\_code) as frequency

from parking\_violations

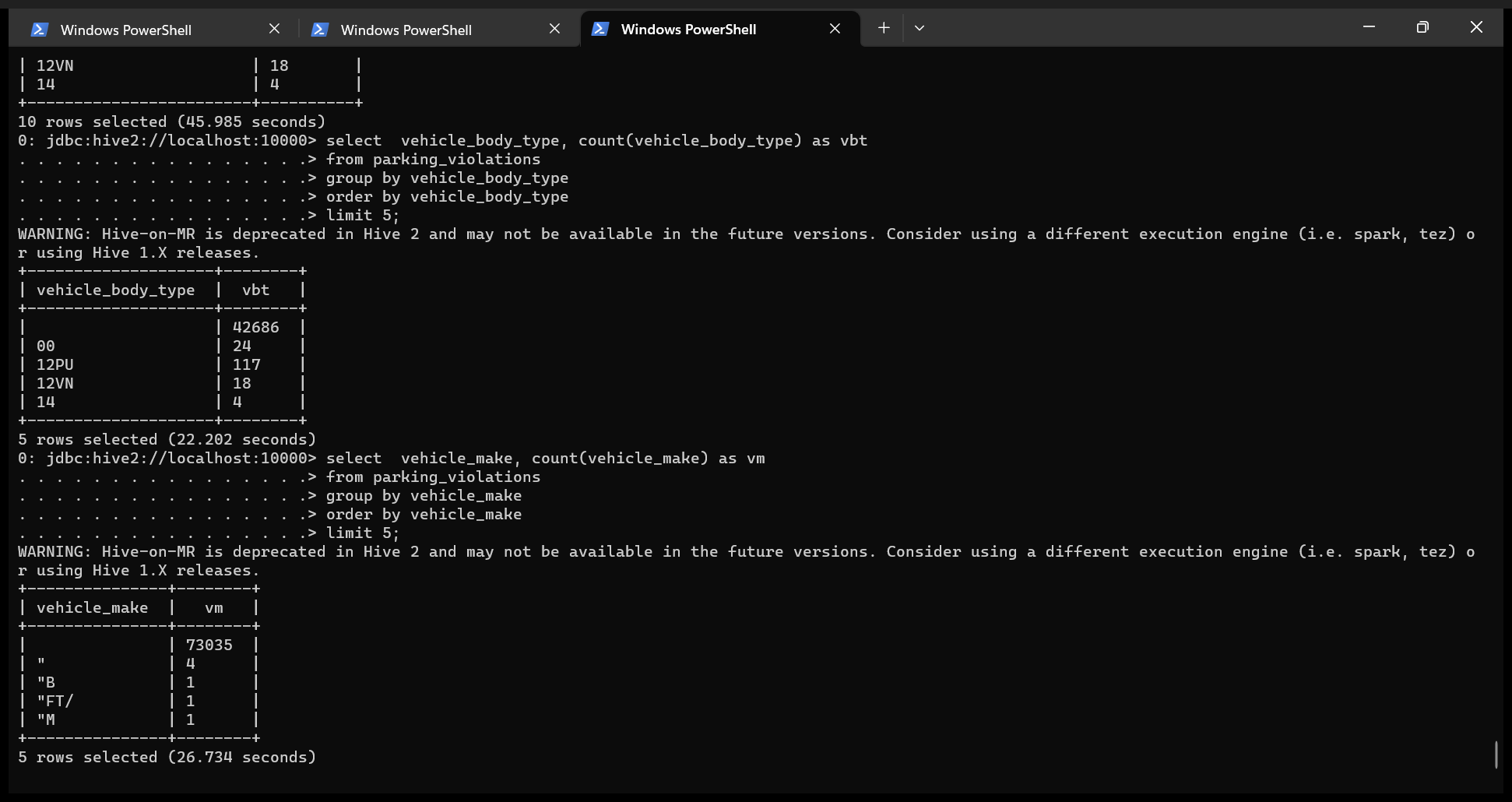
group by violation\_code

order by violation\_code desc

limit 5;



1. How often does each vehicle body type get a parking ticket? How about the vehicle make? (find the top 5 for both)



3.) A precinct is a police station that has a certain zone of the city under its command. Find the (5 highest) frequencies of:

a.) Violating Precincts (this is the precinct of the zone where the violation occurred)

b.) Issuer Precincts (this is the precinct that issued the ticket)

SELECT s.Violation\_Precinct, s.vps

FROM (

SELECT Violation\_Precinct, COUNT(Violation\_Precinct) AS vps

FROM parking\_violations

GROUP BY Violation\_Precinct

) s

ORDER BY s.vps DESC

LIMIT 5;

SELECT s.Issuer\_Precinct, s.vp

FROM (

SELECT Issuer\_Precinct, COUNT(Issuer\_Precinct) AS vp

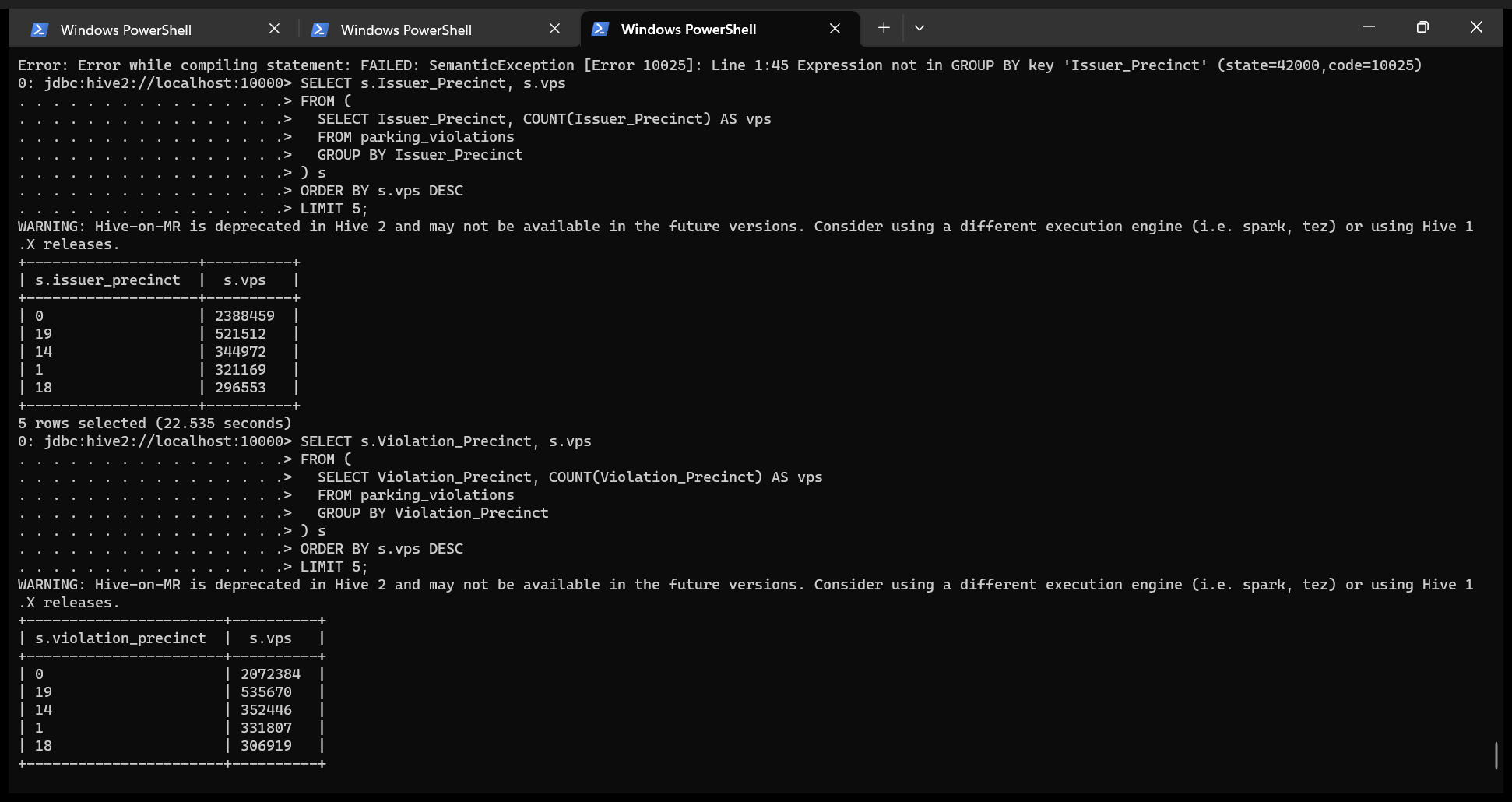
FROM parking\_violations

GROUP BY Issuer\_Precinct

) s

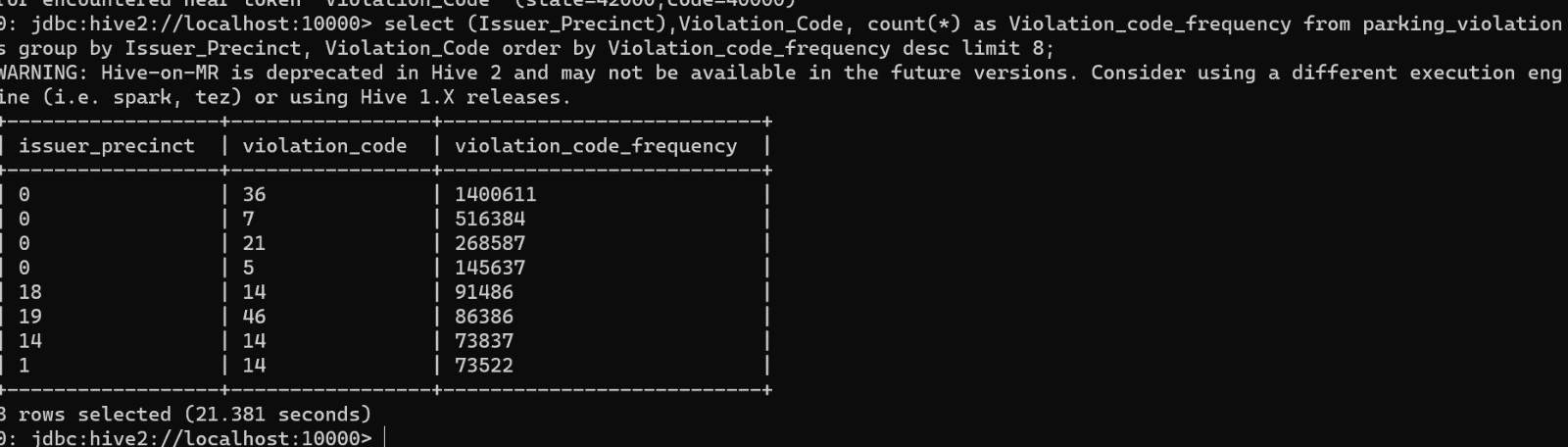
ORDER BY s.vp DESC

LIMIT 5;



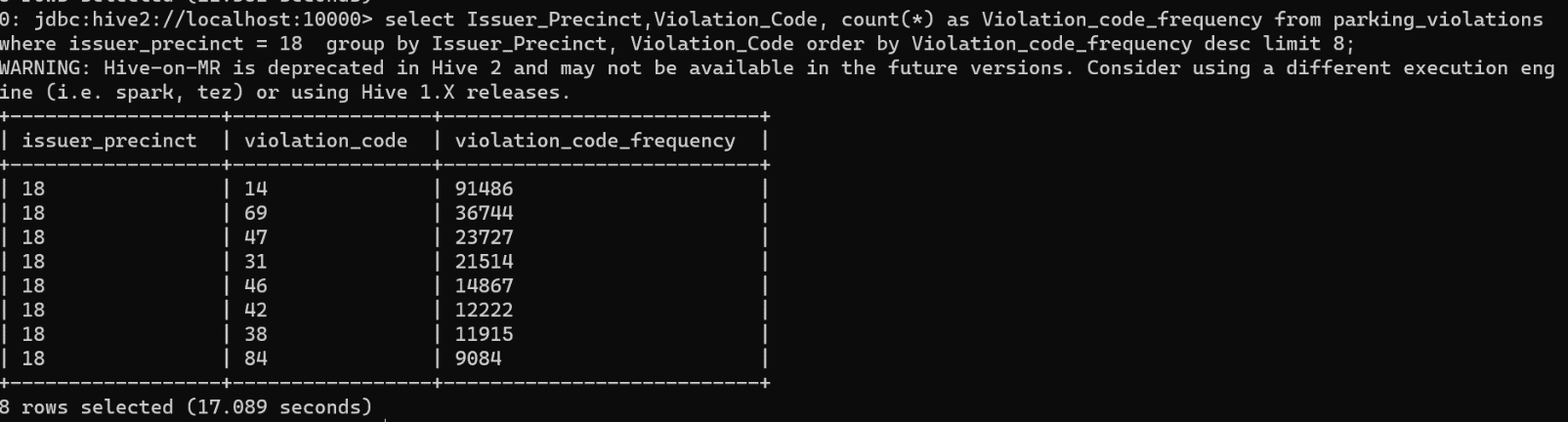
4.) Find the violation code frequency across 3 precincts which have issued the most number of tickets - do these precinct zones have an exceptionally high frequency of certain violation codes?

select Issuer\_Precinct,Violation\_Code, count(\*) as Violation\_code\_frequency from parking\_violations group by Issuer\_Precinct, Violation\_Code order by Violation\_code\_frequency desc limit 8;

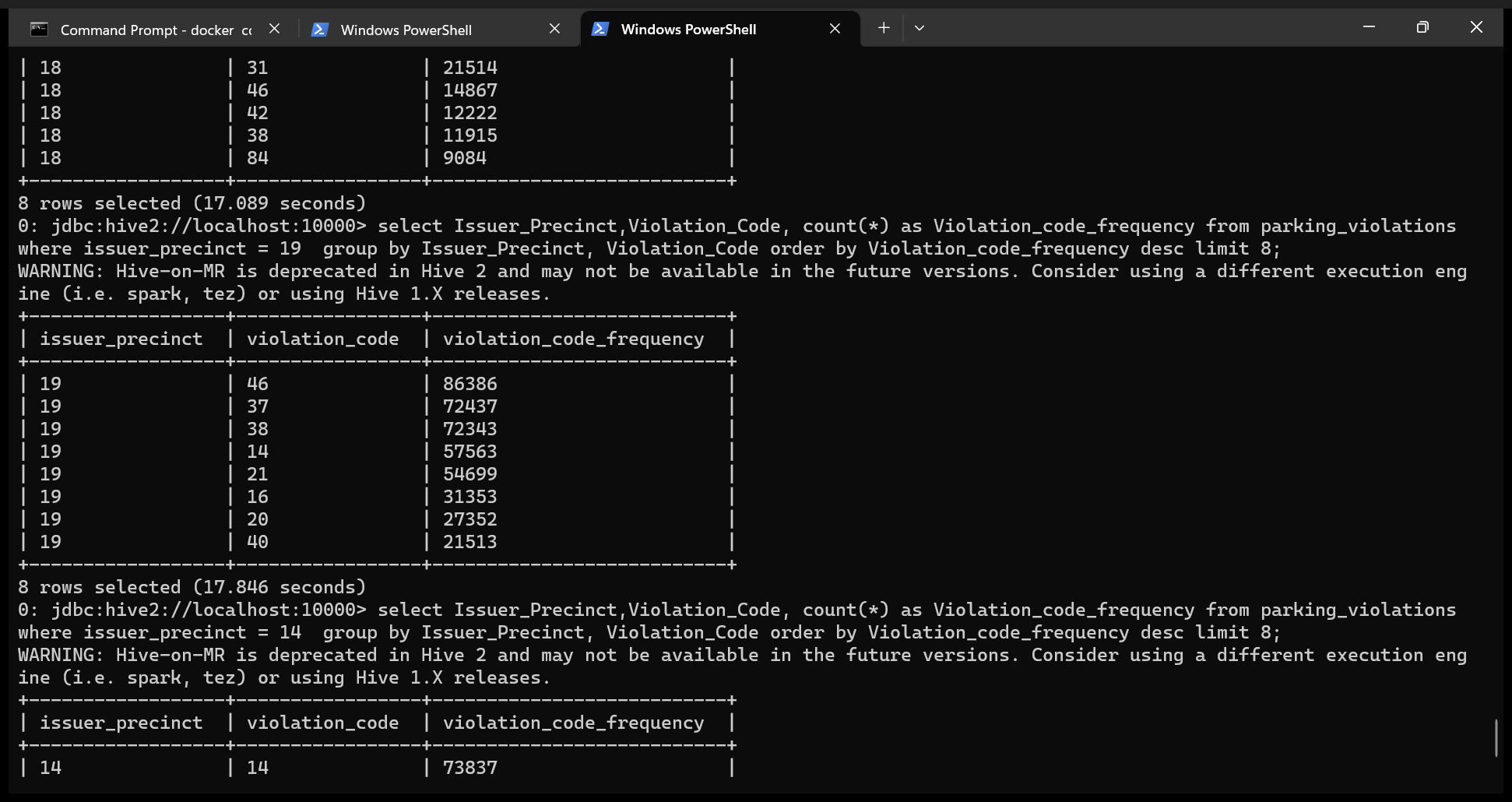


We will not be considering 0. Therefore 18,19,14 are the three issuer precincts which have the maximum number of violations. Lets dive into the Issuer Precincts one by one.

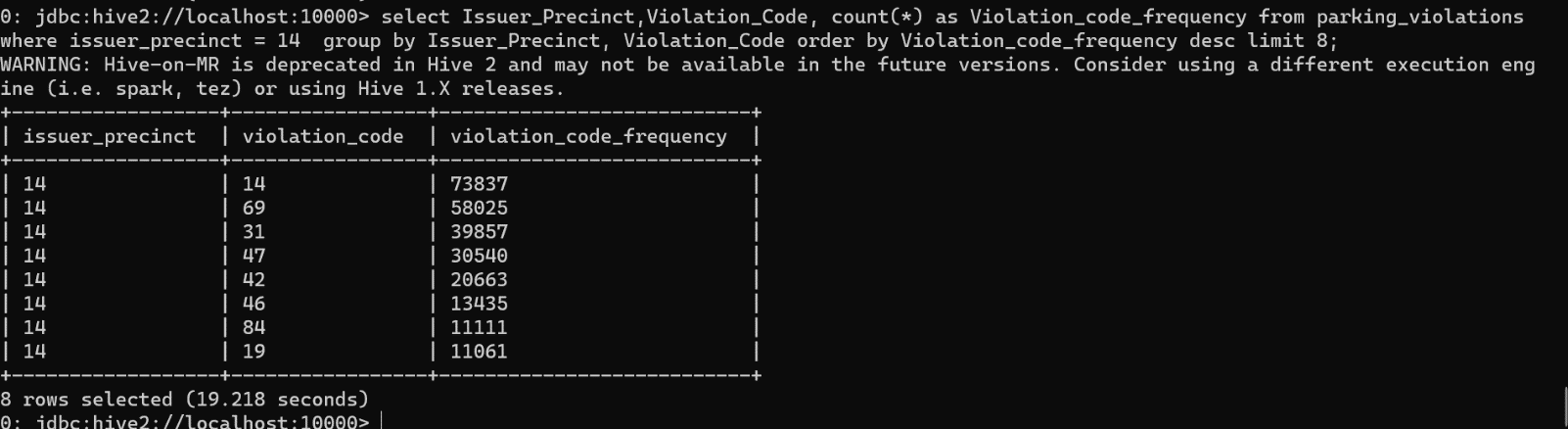
select Issuer\_Precinct,Violation\_Code, count(\*) as Violation\_code\_frequency from parking\_violations where issuer\_precinct = 18 group by Issuer\_Precinct, Violation\_Code order by Violation\_code\_frequency desc limit 8;



select Issuer\_Precinct,Violation\_Code, count(\*) as Violation\_code\_frequency from parking\_violations where issuer\_precinct = 19 group by Issuer\_Precinct, Violation\_Code order by Violation\_code\_frequency desc limit 8;

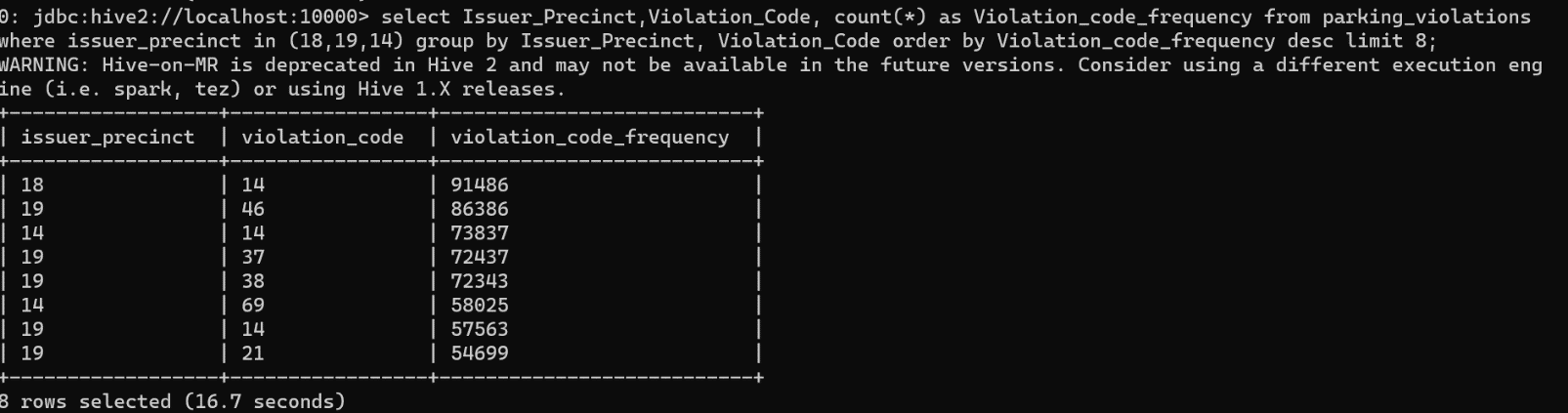


select Issuer\_Precinct,Violation\_Code, count(\*) as Violation\_code\_frequency from parking\_violations where issuer\_precinct = 14 group by Issuer\_Precinct, Violation\_Code order by Violation\_code\_frequency desc limit 8;

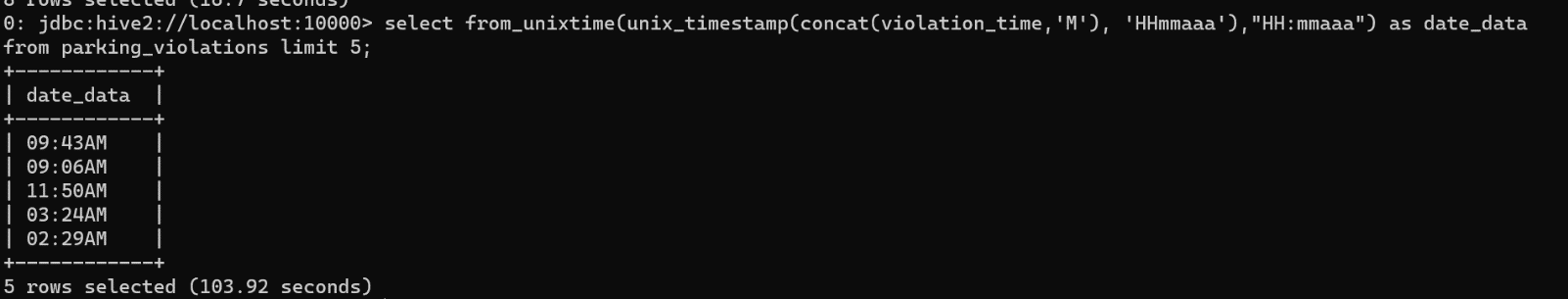


**🡺Common codes accross precincts**

select Issuer\_Precinct,Violation\_Code, count(\*) as Violation\_code\_frequency from parking\_violations where issuer\_precinct in (18,19,14) group by Issuer\_Precinct, Violation\_Code order by Violation\_code\_frequency desc limit 8;



5.) Find out the properties of parking violations across different times of the day: The Violation Time field is specified in a strange format. Find a way to make this into a time attribute that you can use to divide into groups.



6.) Divide 24 hours into 6 equal discrete bins of time. The intervals you choose are at your discretion. For each of these groups, find the 3 most commonly occurring violations

create view vw\_parking\_violations\_partitoned\_2017

partitioned on (Violation\_Code)

as

SELECT

Summons\_Number,

Violation\_Time,

Issuer\_Precinct,

case

when substring(Violation\_Time,1,2) in ('00','01','02','03','12') and upper(substring(Violation\_Time,-1))='A' then 1

when substring(Violation\_Time,1,2) in ('04','05','06','07') and upper(substring(Violation\_Time,-1))='A' then 2

when substring(Violation\_Time,1,2) in ('08','09','10','11') and upper(substring(Violation\_Time,-1))='A' then 3

when substring(Violation\_Time,1,2) in ('12','00','01','02','03') and upper(substring(Violation\_Time,-1))='P' then 4

when substring(Violation\_Time,1,2) in ('04','05','06','07') and upper(substring(Violation\_Time,-1))='P' then 5

when substring(Violation\_Time,1,2) in ('08','09','10','11') and upper(substring(Violation\_Time,-1))='P'then 6

else null

end as Violation\_Time\_bin,

Violation\_Code

from parking\_violations

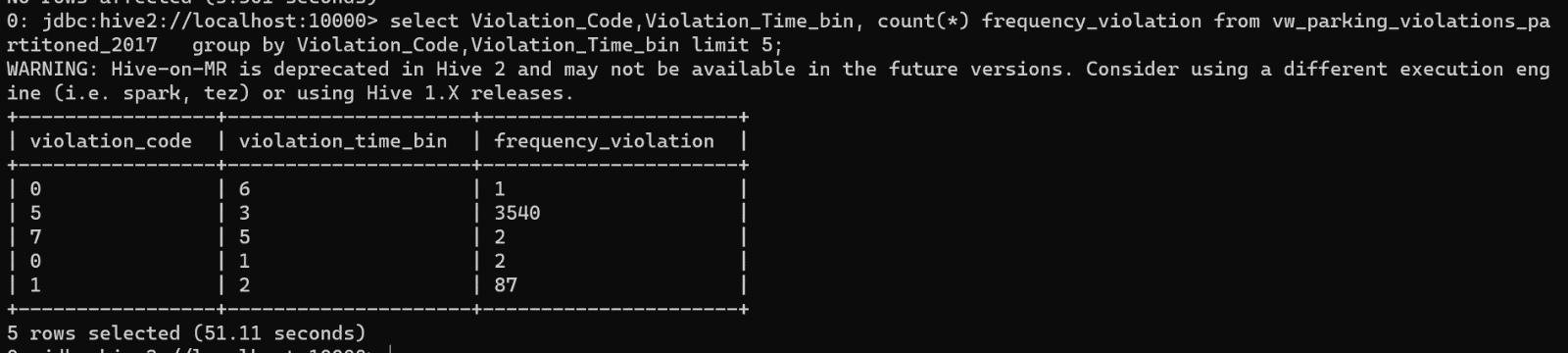
where

Violation\_Time is not null

or (length(Violation\_Time)=5 and upper(substring(Violation\_Time,-1))in ('A','P')

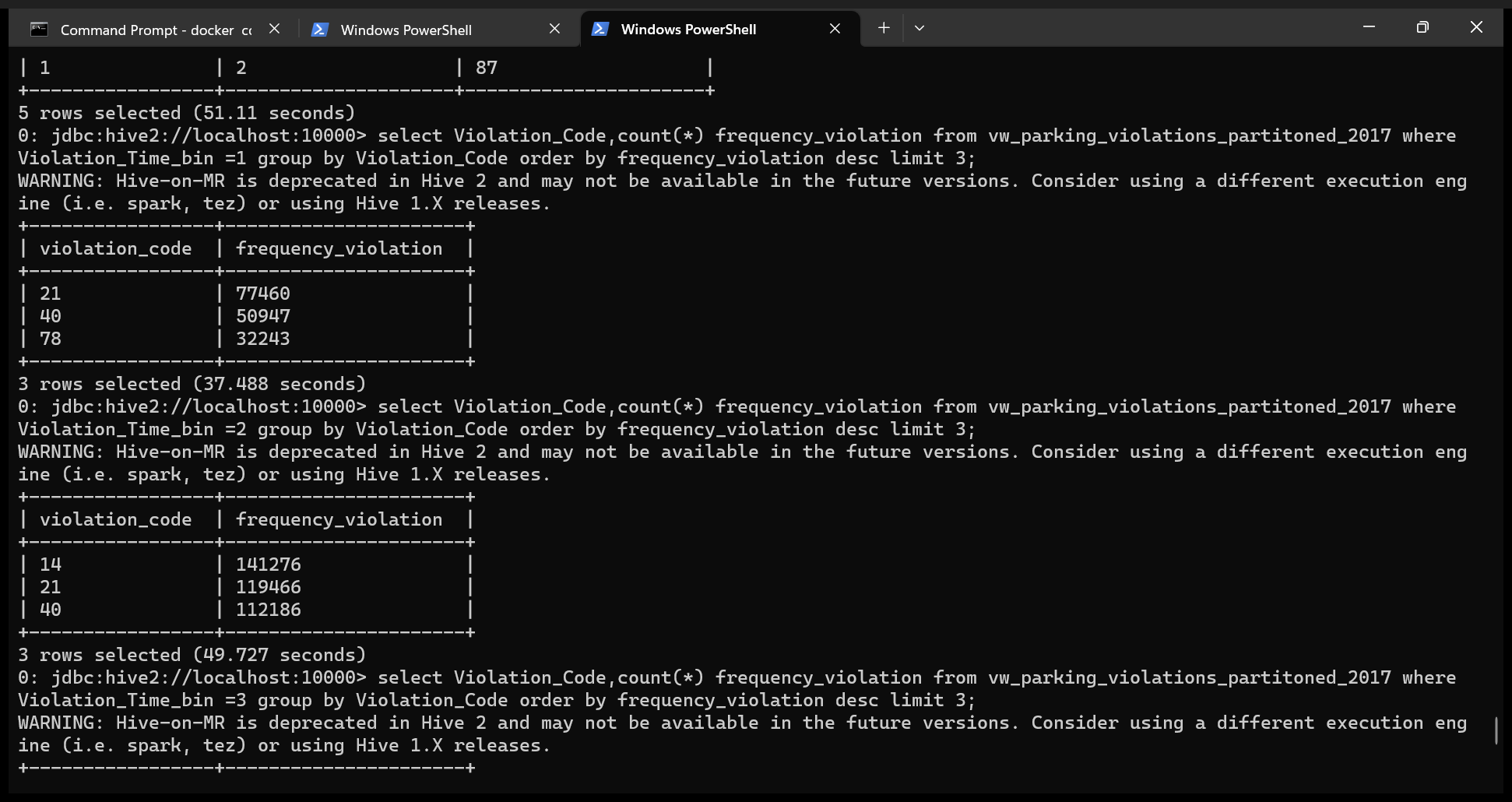
and substring(Violation\_Time,1,2) in ('00','01','02','03','04','05','06','07', '08','09','10','11','12'));

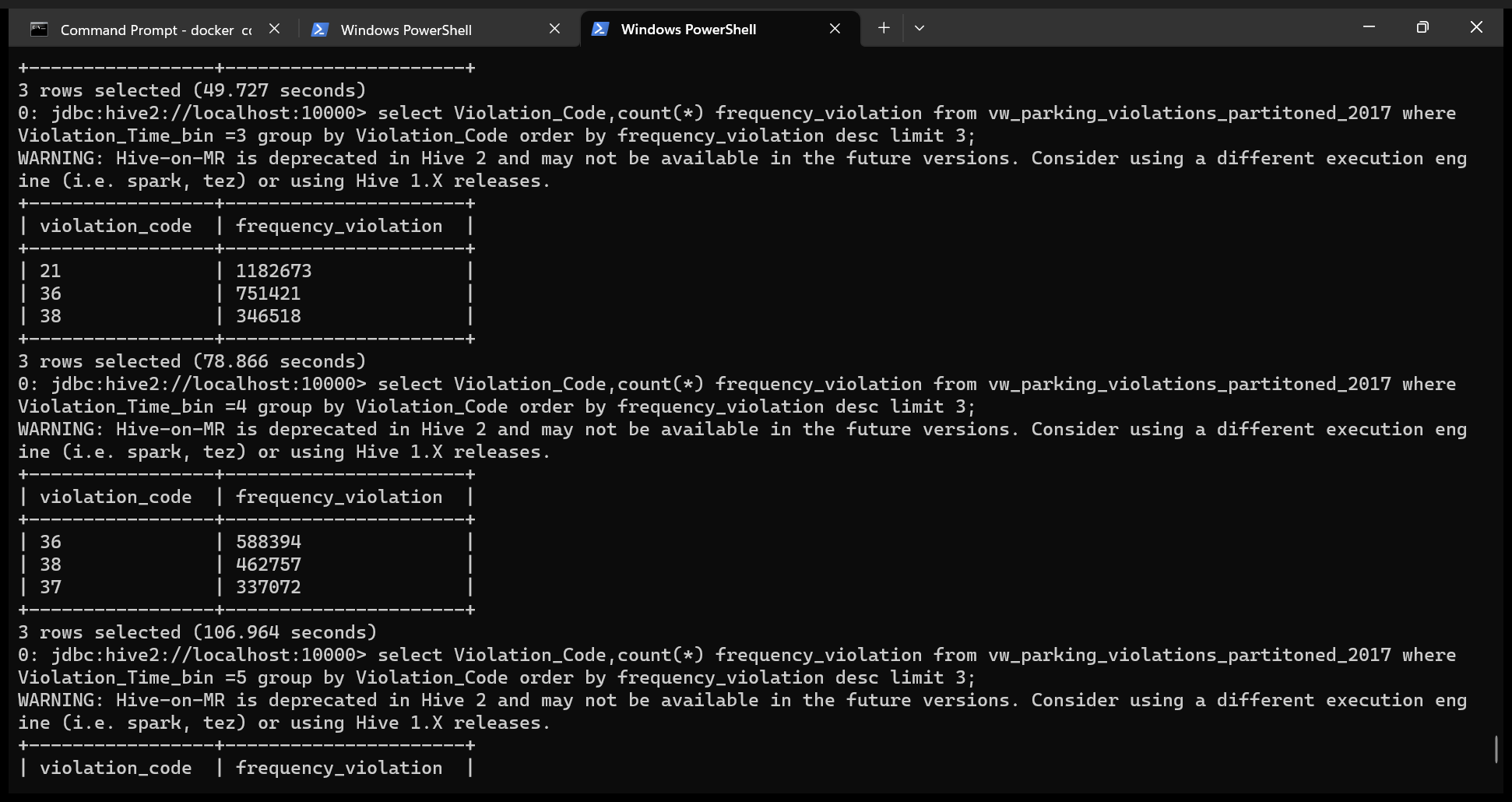
select Violation\_Code,Violation\_Time\_bin, count(\*) frequency\_violation from vw\_parking\_violations\_partitoned\_2017 group by Violation\_Code,Violation\_Time\_bin limit 5;

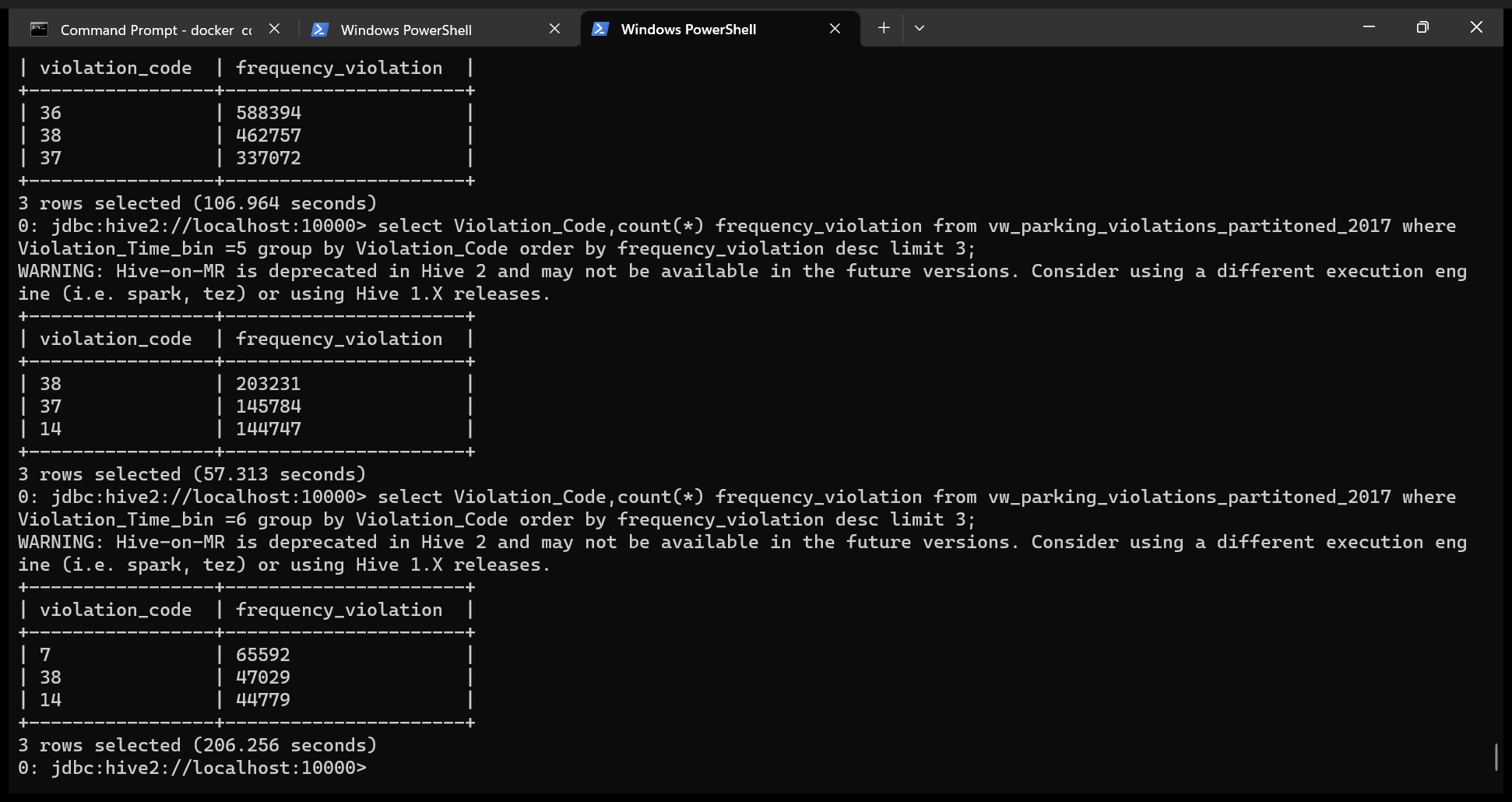


select Violation\_Code,count(\*) frequency\_violation from vw\_parking\_violations\_partitoned\_2017 where Violation\_Time\_bin =1 group by Violation\_Code order by frequency\_violation desc limit 3;

similar way applying on all bins







7.) Now, try another direction. For the 3 most commonly occurring violation codes, find the most common times of day (in terms of the bins from the previous part)

