

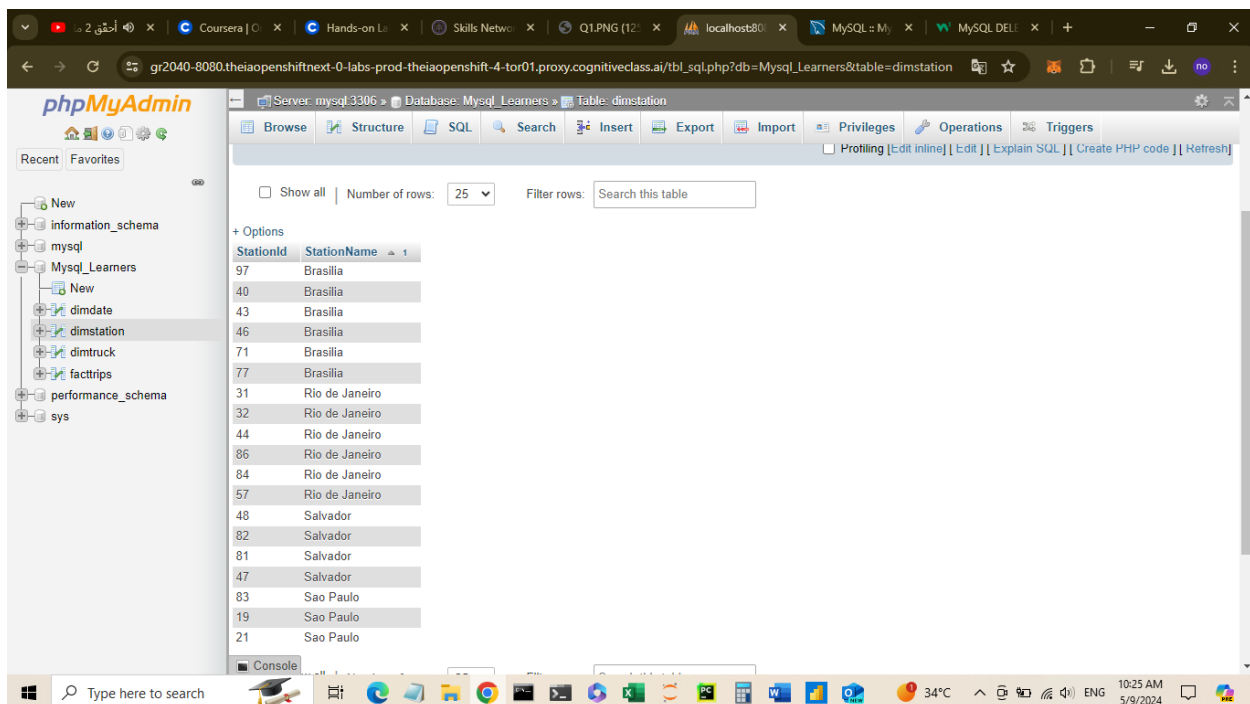
Execute SQL Queries

Exercise 1: List all stations in an alphabetical order. Output should contain StationId, StationName.

Select StationId, StationName

From dimstation

Order by StationName asc



Exercise 2: List all trips that collected waste > 40. Output should contain TripId, Waste.

Select TripId, Waste

From facttrips

Where Waste > 40

TripId	Waste
23480	40.33
23481	41.92
23483	43.56
23486	41.89
23488	43.25
23498	43.11
23499	43
23504	41.67
23508	44.95
23509	44.53
23510	43.42
23512	43.2
23514	43.57
23516	43.23
23528	41.02
23529	43.95
23532	44.6
23542	44.83
23545	40.59
23546	40.06
23551	44.11
23552	41.06
23557	42.04
23563	43.04
23565	40.71

Exercise 3: List average waste collected for each date. Output should contain Dateld, average waste.

Select Dateld, AVG(Waste) as avg_Waste

From facttrips

Group by Dateld

The screenshot shows the phpMyAdmin interface. On the left, the database structure is visible, including the 'MySQL_Learners' database and the 'facttrips' table. The main panel displays the 'facttrips' table data, which consists of two columns: 'DateId' and 'avg_Waste'.

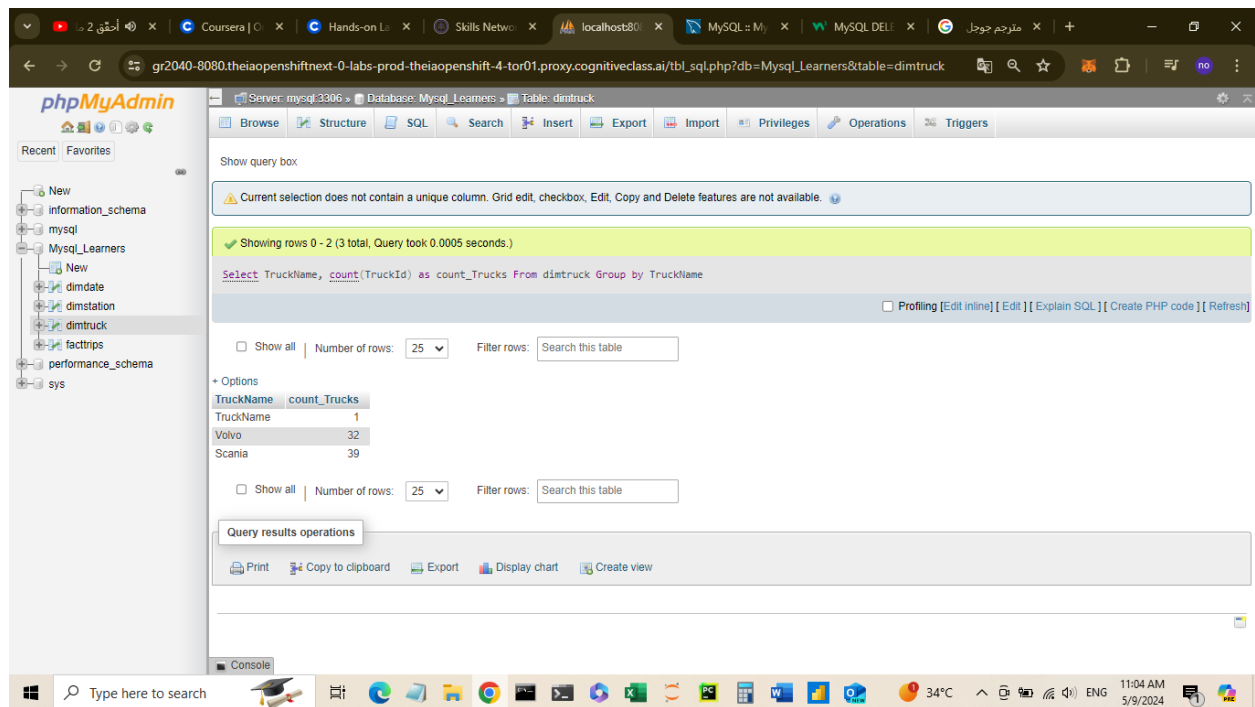
DateId	avg_Waste
1	37.44900263157898
2	37.13526315789475
3	37.71575657894735
4	36.99509868421052
5	37.344046052631576
6	37.44200657894739
7	37.61592105263156
8	37.2935852631578
9	37.62095394736841
10	37.6403289473684
11	37.57217105263157
12	37.510953947368435
13	37.116381578947404
14	37.13440789473685
15	37.588190789473664
16	37.21138157894736
17	37.05243421052632
18	37.64039473684212
19	37.51713815789474
20	37.27898026315791
21	37.86092105263159
22	37.29388157894736
23	37.344407894736825
24	37.50049342105262
25	37.237499999999976

Exercise 4: List truck Names with their count. Output should contain TruckName, count

Select TruckName, count(TruckId) as count_Trucks

From dimtruck

Group by TruckName



Exercise 5: List City with total waste collected. Output should contain CityName, total_Waste

Select st.StationName as CityName, sum(tr.Waste) as total_Waste

From dimstation st

Left outer join facttrips tr

On st.StationId = tr.StationId

Group by st.StationName

The screenshot shows the phpMyAdmin interface with the following details:

- Database:** Mysq_Learners
- Table:** dimstation
- Query:**

```

Select st.StationName as CityName, sum(tr.Waste) as total_Waste From dimstation st Left outer join facttrips tr On st.StationId = tr.StationId Group by st.StationName

```
- Results:**

CityName	total_Waste
Sao Paulo	488028.8499999973
Rio de Janeiro	487200.53999999934
Brasilia	487831.42999999999
Salvador	487085.43999999965

Exercise 6: List minimum waste collected per quarter in 2019.

Output should contain QuarterName, minimum waste.

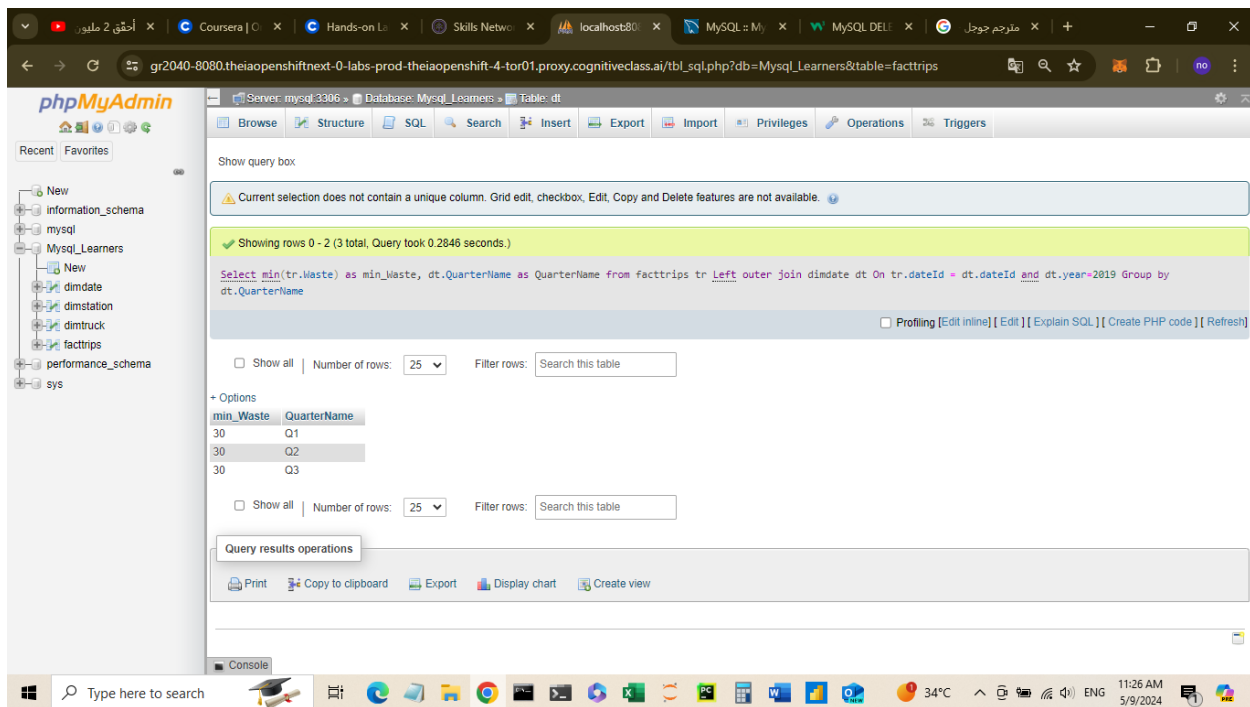
Select min(tr.Waste) as min_Waste, dt.QuarterName as QuarterName

from facttrips tr

Left outer join dimdate dt

On tr.dateId = dt.dateId and dt.year=2019

Group by dt.QuarterName



Exercise 7: List maximum waste collected in Q1 in Sao Paulo.

Output should contain QuarterName, City, maximum Waste.

Select dt.QuarterName, st.StationName, max(tr.Waste) as max_Waste

From facttrips tr

Left outer join dimstation st

On tr.StationId = st.StationId

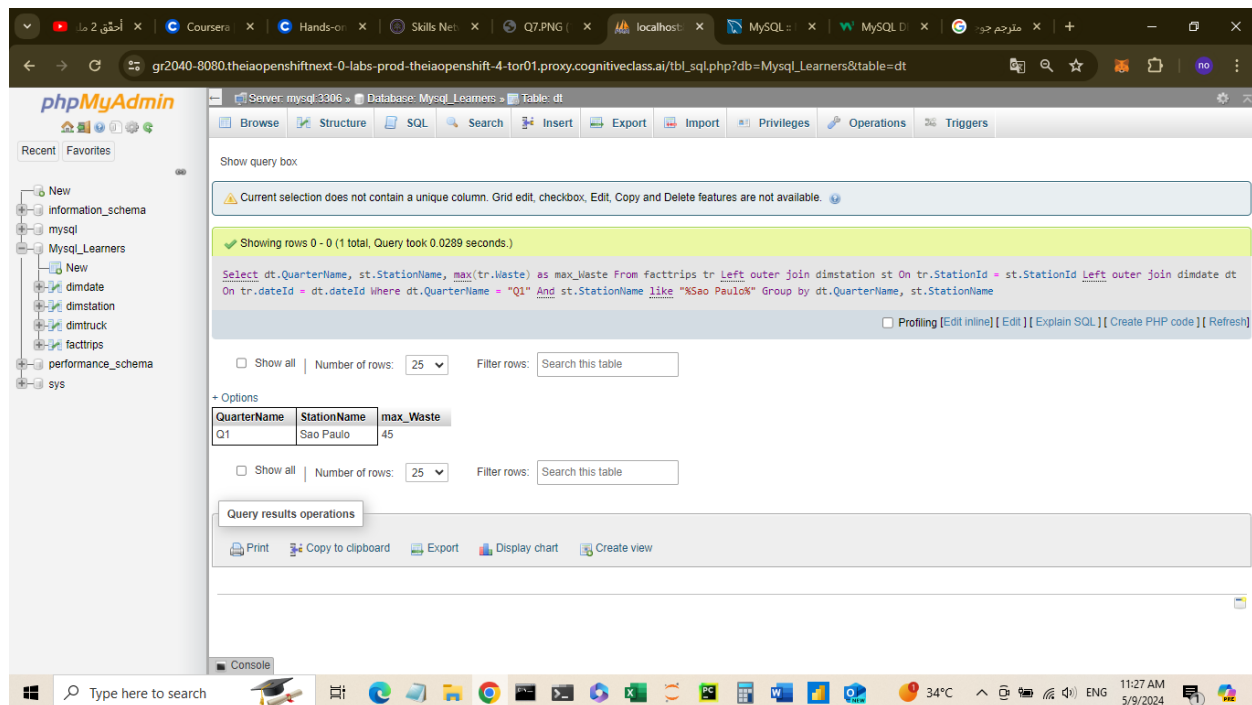
Left outer join dimdate dt

On tr.dateId = dt.dateId

Where dt.QuarterName = "Q1"

And st.StationName like "%Sao Paulo%"

Group by dt.QuarterName, st.StationName



Exercise 8: List the days of the week results in the highest average waste collected by Volvo trucks. Output should contain WeekDayName, TruckName, avg_Waste.

Select dt.WeekDayName, tru.truckName, AVG(tr.Waste) as avg_Waste

From facttrips tr

Left outer join dimtruck tru

On tr.TruckId = tru.TruckId

Left outer join dimdate dt

On tr.dateId = dt.dateId

Where tru.TruckName like "%Volvo%"

Group by dt.WeekDayName, tru.TruckName

ORDER by AVG(tr.Waste)

DESC Limit 7

The screenshot shows the phpMyAdmin interface with the following details:

- Database:** Mysql_Learners
- Table:** dt
- Query:**

```

Select dt.WeekDayName, tru.truckName, AVG(tr.Waste) as avg_Waste From facttrips tr Left outer join dimtruck tru On tr.TruckId = tru.TruckId Left outer join dimdate dt On tr.dateId = dt.dateId Where tru.TruckName like "%Volvo%" Group by dt.WeekDayName, tru.TruckName ORDER by AVG(tr.Waste) DESC Limit 7

```
- Results:**

WeekDayName	truckName	avg_Waste
Saturday	Volvo	37.5637151880563
Thursday	Volvo	37.563407853253054
Monday	Volvo	37.54559559834932
Friday	Volvo	37.527673210161794
Wednesday	Volvo	37.50538816738819
Tuesday	Volvo	37.498568688995716
Sunday	Volvo	37.47507111597375

Exercise 9: List the dates when each city collected its maximum Waste. Output should contain city, date, maximum Waste.

Select st.StationName as city, dt.date, a.waste

From (

Select StationId, DateId, Waste, rank() over (partition by StationId order by Waste desc) as rnk

From facttrips) a

Left outer join dimdate dt

On a.DateId = dt.DateId

Left outer join dimstation st

On a.StationId = st.StationId

Where a.rnk = 1

gr2040-8080.theiaopenshiftnext-0-labs-prod-theiaopenshift-4-tor01.proxy.cognitiveclass.ai/tbl_sql.php?db=MySQL_Learners&table=dt

phpMyAdmin

Server: mysql:3306 Database: MySQL_Learners Table: dt

Options

city	date	waste
Sao Paulo	08-07-2019	45
Sao Paulo	21-05-2019	45
Sao Paulo	11-04-2019	45
Sao Paulo	05-05-2019	45
Sao Paulo	17-07-2019	45
Sao Paulo	09-05-2019	45
Rio de Janeiro	02-05-2019	45
Rio de Janeiro	02-08-2019	45
Brasilia	21-07-2019	45
Brasilia	22-08-2019	45
Brasilia	13-03-2019	44.99
Brasilia	27-05-2019	44.99
Brasilia	21-06-2019	44.99
Rio de Janeiro	02-07-2019	45
Brasilia	01-05-2019	45
Brasilia	08-05-2019	45
Brasilia	04-07-2019	45
Salvador	05-05-2019	45
Salvador	21-08-2019	45
Salvador	13-03-2019	45
Salvador	07-05-2019	45
Salvador	04-04-2019	45
Rio de Janeiro	18-06-2019	45
Rio de Janeiro	26-08-2019	45
Brasilia	13-05-2019	45

Console

Show all Number of rows: 25 Filter rows: Search this table

Type here to search

34°C 11:30 AM 5/9/2024