

Executing SQL Queries



Hands-on Lab : Executing SQL Queries

Estimated time needed: 30 minutes

In this lab you will be using phpMyAdmin, which is a free tool embedded in this lab environment to work on MySQL.

Objectives

After completing this lab you will be able to:

- Create a Database
- Create and load tables using csv files
- Execute SQL queries

Software used in this lab

In this lab, you will use [MySQL](#). MySQL is a Relational Database Management System (RDBMS) designed to efficiently store, manipulate, and retrieve data.



To complete this lab you will utilize MySQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

Prework - Create and populate database

TASK A: Create a Database


1. Start the MySQL service session using the `Open MySQL Page in IDE` button.

[Open MySQL Page in IDE](#)

To start the MySQL, click `Start`.

MySQL

INACTIVE

 8.0.22 |  5.0.4 |  2.0.2

Connect to MySQL and phpMyAdmin directly in your Skills Network L

Start

Stop

Summary

Connection Information



Details

Get started with MySQL in a faster, easier way. To launch your databa



2. Once MySQL has started, click on phpMyAdmin button to open phpMyAdmin in the same window.

File Edit Selection View Go Run Terminal Help


SKIL...  


> DATABASES

> BIG DATA

> CLOUD




> OTHER

 Launch A...

MySQL 

MySQL

ACTIVE

 v8.0.22 |  v5.0.4 |  v14.14

Connect to MySQL and phpMyAdmin directly in your Skills Network Labs environment.

Stop

Summary


Connection Information

Details

Your database and phpMyAdmin server are now ready to use and available with the following login credentials. For more details on how to navigate MySQL, please check out the Details section.


Username:

lavanyas




Password:

NzgyNS1sYXZhbnlh



You can manage MySQL via:

phpMyAdmin



3. You will see the phpMyAdmin GUI tool.



Recent

Favorites

- + New
- + information_schema
- + mysql
- + performance_schema
- + sakila
- + sys

← Server: mysql:3306



Databases



SQL



Status



User accounts



Export

General settings



Server connection collation: ⓘ

utf8mb4_unicode_ci



More settings

Appearance settings



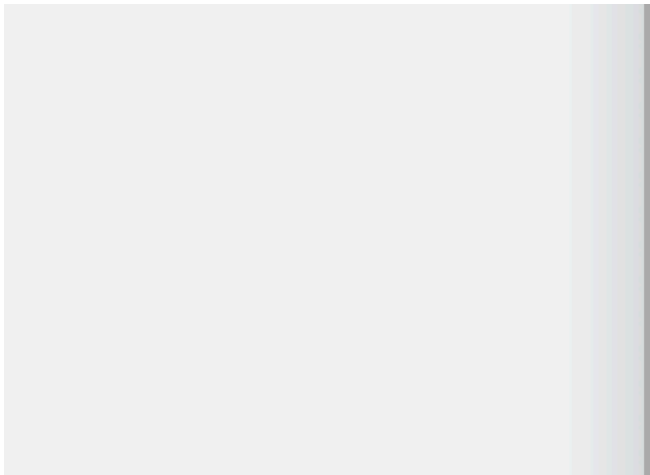
Language ⓘ

English



Theme:

pmahomme ▼



4. In the tree-view, click **New** to create a new empty database. Then enter **Mysql_Learners** as the name of the database and select utf8_general_ci and click **Create**.

UTF-8 is the most commonly used character encoding for content or data.

 **Databases**

 **SQL**


 **Status**


 **User accounts**

 **Export**










 **Import**

Databases

 **Create database** 



Create

	Database 	Collation	Master replication	Action
<input type="checkbox"/>	information_schema	utf8_general_ci	 Replicated	 Check privileges
<input type="checkbox"/>	mysql	utf8mb4_0900_ai_ci	 Replicated	 Check privileges
<input type="checkbox"/>	performance_schema	utf8mb4_0900_ai_ci	 Replicated	 Check privileges
<input type="checkbox"/>	sys	utf8mb4_0900_ai_ci	 Replicated	 Check privileges
Total: 4				

TASK B: Create and load tables using csv files.

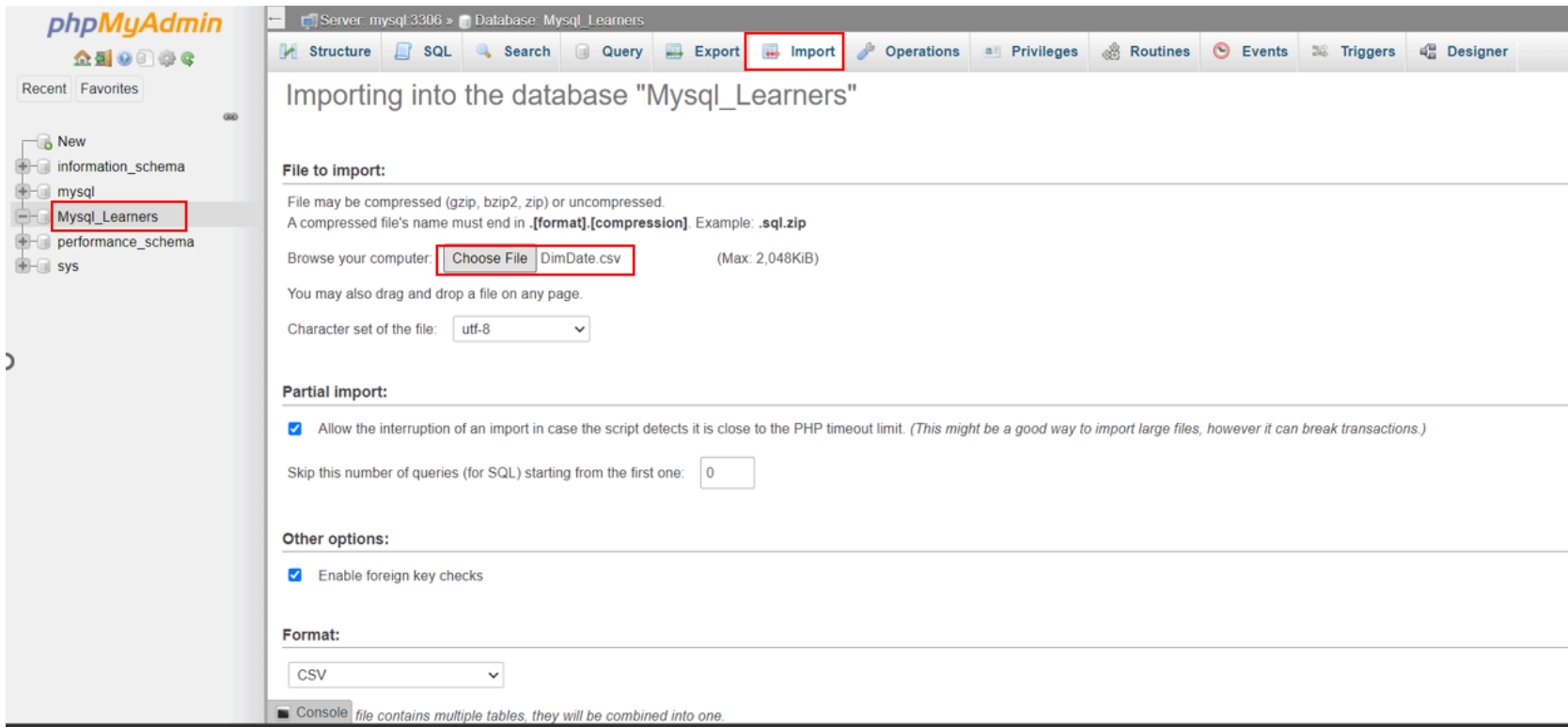
1. Download the 4 csv files below to your local computer:

- [dimdate.csv](#)
- [dimtruck.csv](#)
- [dimstation.csv](#)

- [facttrips.csv](#)

2. To load each **csv** file do the following steps.

- Select your database **Mysql_Learners** and click on **Import** tab and select the **csv** file.



The screenshot shows the phpMyAdmin web interface. On the left sidebar, the database 'Mysql_Learners' is selected and highlighted with a red box. The main panel shows the 'Import' tab, also highlighted with a red box. The title of the page is 'Importing into the database "Mysql_Learners"'. Under the 'File to import:' section, there is a text box for the file name, currently containing 'DimDate.csv', and a 'Choose File' button next to it, both highlighted with red boxes. Below this, there is a dropdown menu for 'Character set of the file' set to 'utf-8'. The 'Partial import:' section has a checked checkbox for 'Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to import large files, however it can break transactions.)'. The 'Other options:' section has a checked checkbox for 'Enable foreign key checks'. The 'Format:' section has a dropdown menu set to 'CSV'. At the bottom, there is a 'Go' button and a console area with the message 'file contains multiple tables, they will be combined into one.'

- Then scroll down and check the box as shown below and click on **Go** to load the csv file.

phpMyAdmin

Server: mysql:3306 » Database: Mysql_Learners

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Designer

Recent Favorites

New
information_schema
mysql
Mysql_Learners
performance_schema
sys

Note: If the file contains multiple tables, they will be combined into one.

Format-specific options:

☐ Update data when duplicate keys found on import (add ON DUPLICATE KEY UPDATE)

Columns separated with:

Columns enclosed with:

Columns escaped with:

Lines terminated with:

Name of the new table (optional):

Import these many number of rows (optional):

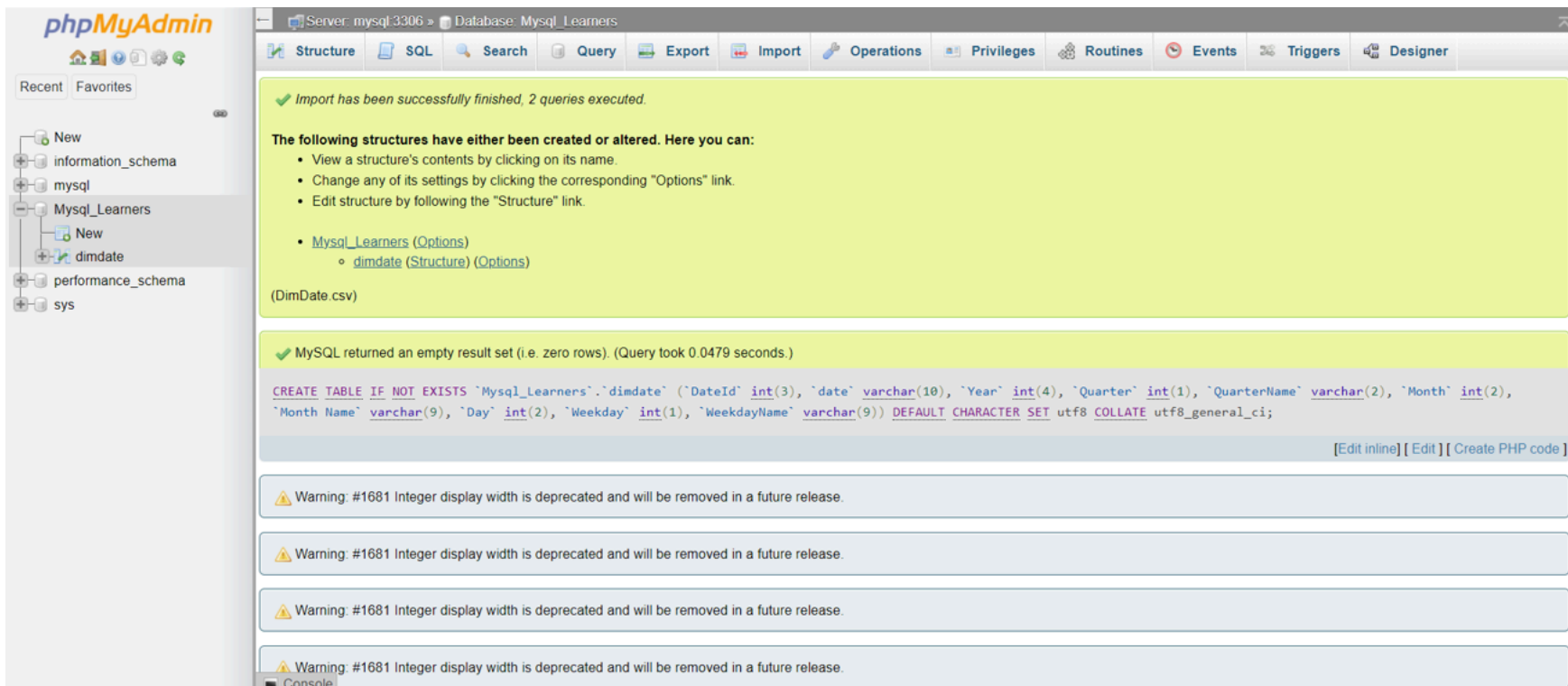
☒ The first line of the file contains the table column names (if this is unchecked, the first line will become part of the data)

☐ Do not abort on INSERT error

Console

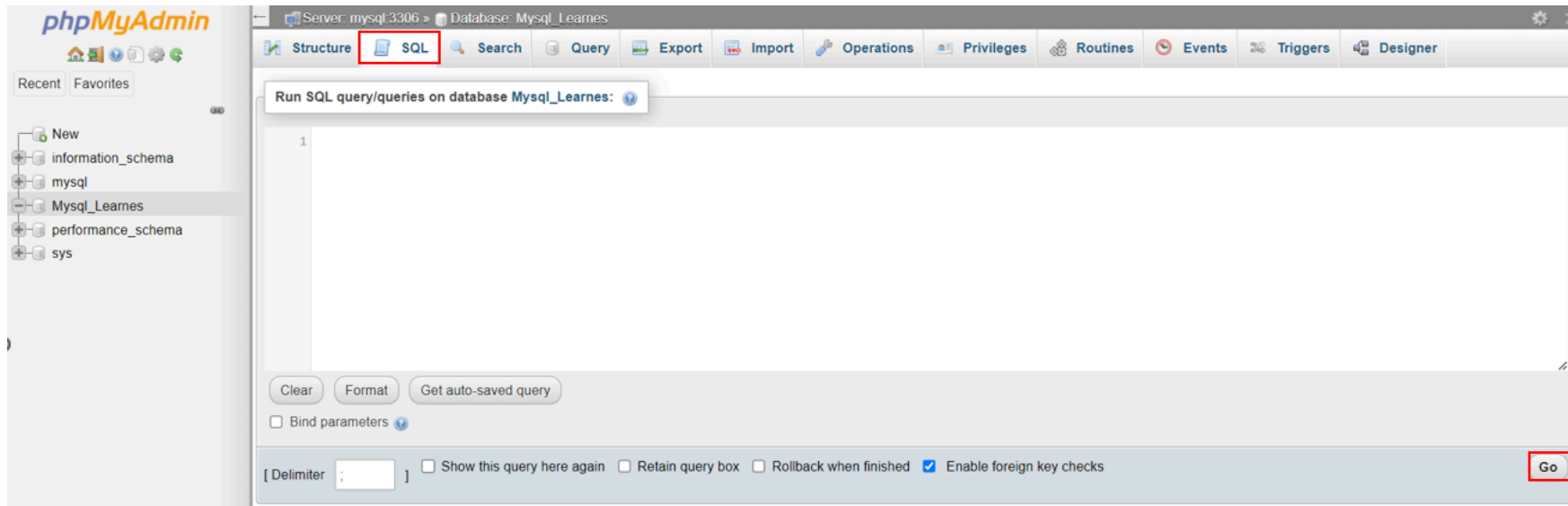
Go

- Once the tables are loaded , you will get a message that the records are inserted successfully.



Further You can import all the other **csv** files in the same way.

3. To run the SQL queries you need to copy the given codes and paste it to the text area of the SQL page and click on **Go**.



Execute SQL Queries

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

▼ Solution Syntax

```
Select StationId, StationName
From dimstation
Order by StationName asc
```

▼ Output

phpMyAdmin

Recent Favorites

New

information_schema

mysql

MySQL_Learners

- New
- dimdate
- dimstation
- dimtruck
- facttrips

performance_schema

sys

Server: mysql:3306 » Database: MySQL_Learners » Table: dimstation

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Triggers

Showing rows 0 - 18 (19 total, Query took 0.0006 seconds.) [StationName: BRASILIA... - SAO PAULO...]

Select StationId, StationName From dimstation Order by StationName asc

Profiling

Edit inline

Edit

Explain SQL

Create PHP code

Show all

Number of rows: 25

Filter rows: Search this table

+ Options

StationId	StationName
97	Brasilia
40	Brasilia
43	Brasilia
46	Brasilia
71	Brasilia
77	Brasilia
31	Rio de Janeiro
32	Rio de Janeiro
44	Rio de Janeiro
86	Rio de Janeiro
84	Rio de Janeiro
57	Rio de Janeiro
48	Salvador
82	Salvador
81	Salvador
47	Salvador
83	Sao Paulo
19	Sao Paulo
24	Sao Paulo

Console

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

▼ Solution Syntax

```
Select TripId, Waste
From facttrips
Where Waste > 40
```

▼ Output

phpMyAdmin

Recent Favorites

Server: mysql 3306 » Database: Mysql_Learners » Table: facttrips

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Showing rows 0 - 24 (17181 total, Query took 0.0004 seconds.)

Select TripId, Waste From facttrips Where Waste > 40

Profiling [Edit inline] [Edit] [Explain SQL]

1 > >> | Number of rows: 25 Filter rows: Search this table

+ Options

TripId	Waste
23480	40.33
23481	41.92
23483	43.56
23486	41.89
23488	43.25
23498	43.11
23499	43.00
23504	41.67
23508	44.95
23509	44.53
23510	43.42
23512	43.20
23514	43.57
23516	43.23
23528	41.02
23529	43.95
23532	44.60
23542	44.83
Console	59

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

▼ Solution Syntax

```
Select DateId, AVG(Waste) as avg_Waste
From facttrips
Group by DateId
```

▼ Output

phpMyAdmin

Server: mysql 3306 » Database: Mysql_Learners » Table: facttrips

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Showing rows 0 - 24 (172 total, Query took 0.0501 seconds.)

Select DateId, AVG(Waste) as avg_Waste From facttrips Group by DateId

☐ Profiling [Edit inline] [Edit] [Explain SQL]

1 > >> | ☐ Show all | Number of rows: 25 Filter rows: Search this table

+ Options

DateId	avg_Waste
1	37.449803
2	37.135263
3	37.715757
4	36.995099
5	37.344046
6	37.442007
7	37.615921
8	37.293586
9	37.620954
10	37.640329
11	37.572171
12	37.510954
13	37.116382
14	37.134408
15	37.588191
16	37.211382
17	37.052434
18	37.640395
Console	7.517138

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

▼ Solution Syntax

```
Select TruckName, count(TruckId) as count_Trucks
From dimtruck
Group by TruckName
```

▼ Output

Server: mysql:3306 » Database: Mysql_Learners » Table: dimtruck

Recent Favorites

New
information_schema
mysql
Mysql_Learners
New
dimdate
dimstation
dimtruck
facttrips
performance_schema
sys

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available. ⓘ

✓ Showing rows 0 - 1 (2 total, Query took 0.0006 seconds.)

Select TruckName, count(TruckId) as count_Trucks From dimtruck Group by TruckName

☐ Profiling [Edit inline] [Edit] [Explain SQL] []

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

+ Options

TruckName	count_Trucks
Volvo	32
Scania	39

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

▼ Solution Syntax

```
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
```

► Output

Exercise 6: List minimum waste collected per quarter in 2019. Output should contain QuarterName, minimum waste.

▼ Solution Syntax

```
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
```



skills Network

► Output

Exercise 7: List maximum waste collected in Q1 in Sao Paulo. Output should contain QuarterName, City, maximum Waste.

▼ Solution Syntax

```
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
```

▼ Output

phpMyAdmin

Server: mysql3306 > Database: Mysql_Learners > Table: dt

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Recent Favorites

New
information_schema
mysql
Mysql_Learners
New
dimdate
dimstation
dimtruck
facttrips
performance_schema
sys

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✔ Showing rows 0 - 0 (1 total, Query took 0.0314 seconds.)

```
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
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

+ Options

QuarterName	StationName	max_Waste
Q1	Sao Paulo	45.00

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.

▼ Solution Syntax

```
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
```

► Output

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

▼ Solution Syntax

```
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
```

▼ Output

phpMyAdmin

Recent

Favorites

New

information_schema

mysql

Mysql_Learners

New

dimdate

dimstation

dimtruck

facttrips

performance_schema

sys

Server: mysql:3306 » Database: Mysql_Learners » Table: dt

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Triggers

Show query box

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available. ⓘ

✔ Showing rows 0 - 0 (1 total, Query took 0.0662 seconds.)

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 1

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

+ Options

WeekDayName	truckName	avg_Waste
Saturday	Volvo	37.563715

Query results operations

Congratulations! You have completed this lab successfully.

Authors

Sudhir Buddhavarapu

Other Contributors

[Pratiksha Verma](#)

© Copyright IBM Corporation. All rights reserved.