# Exercise 1 - Prepare the lab environment:

# Before you start the assignment, complete the following steps to set up the lab:

# Step 1: Download Kafka.

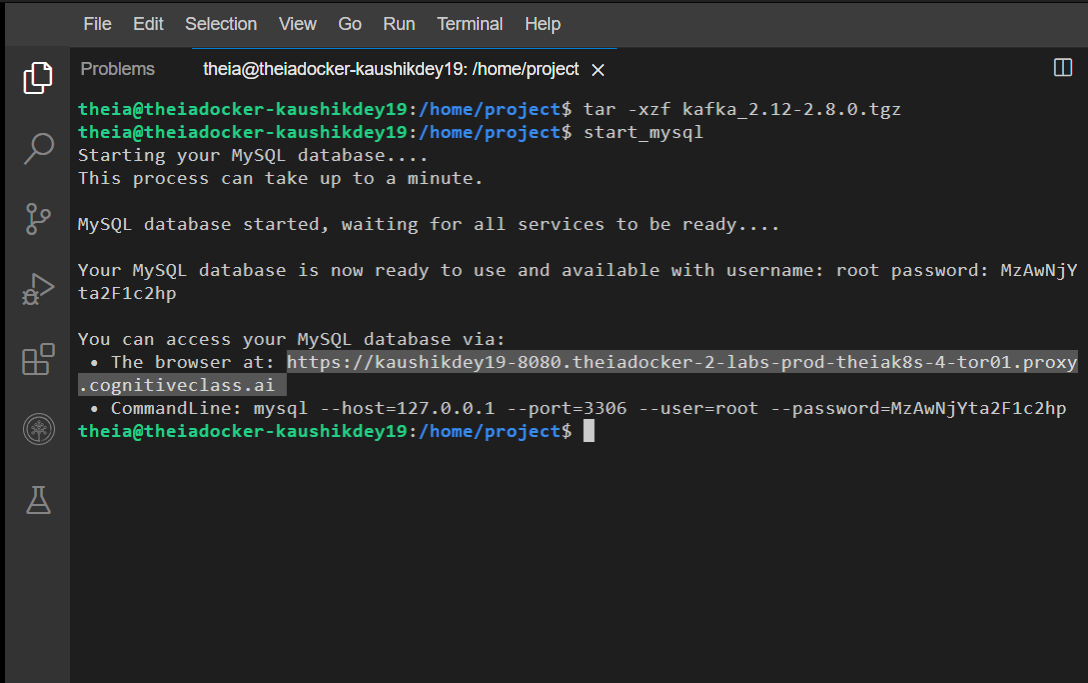
# wget <https://archive.apache.org/dist/kafka/2.8.0/kafka_2.12-2.8.0.tgz>

# 

1. **Step 2: Extract Kafka.**

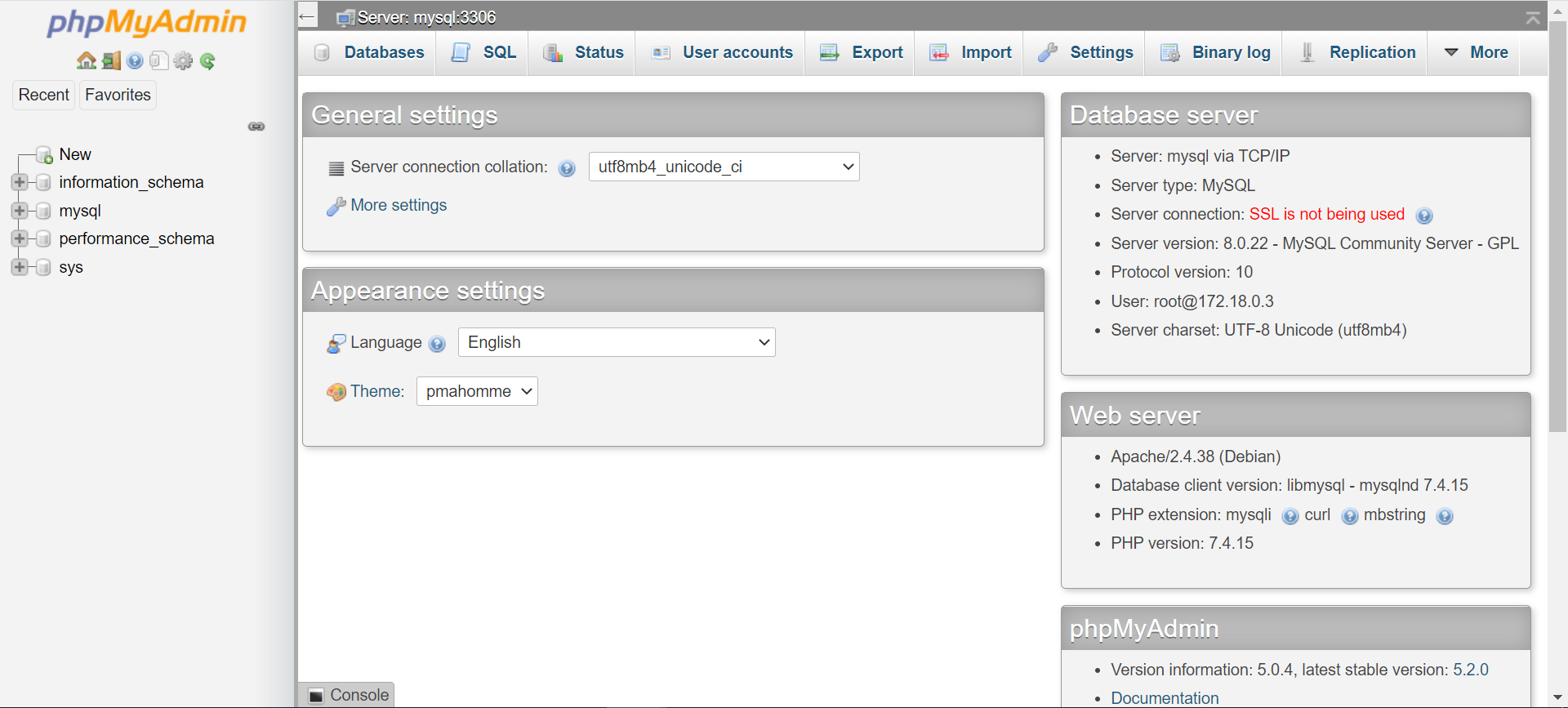
Graphical user interface, text, application

Description automatically generatedtar -xzf kafka\_2.12-2.8.0.tgz.

1. **Step 3: Start MySQL server**

**The Browser View of this MYSQL Instance :**

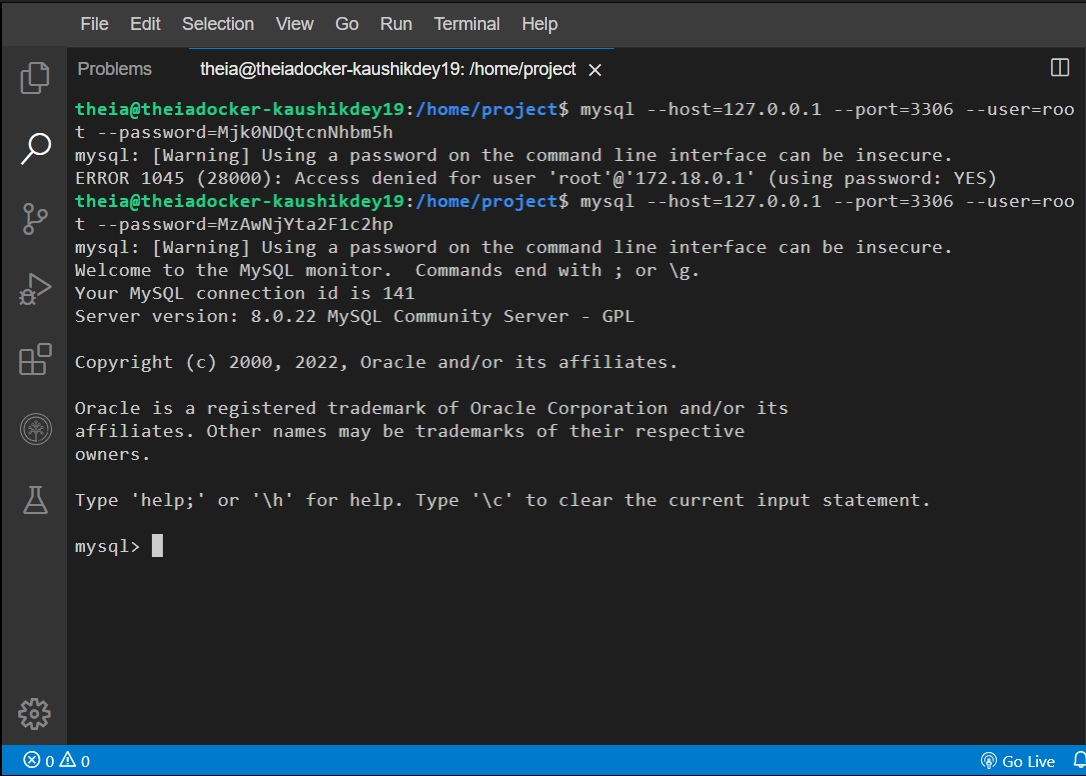
[**https://kaushikdey19-8080.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai/**](https://kaushikdey19-8080.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai/)



1. **Step 4: Connect to the mysql server, using the command below. Make sure you use the password given to you when the MySQL server starts. Please make a note or record of the password because you will need it later.**

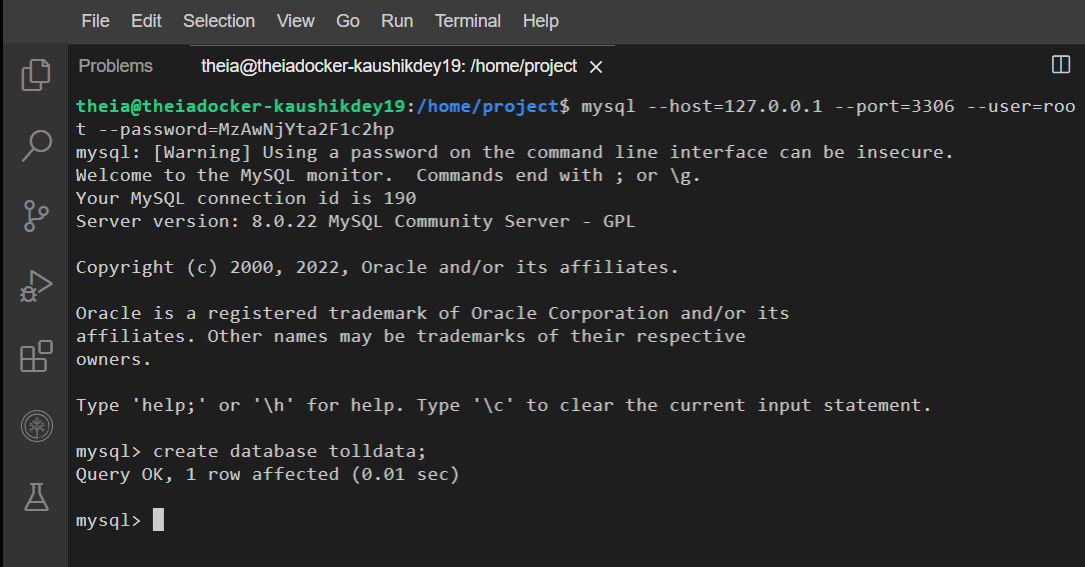
**mysql --host=127.0.0.1 --port=3306 --user=root --password=MzAwNjYta2F1c2hp**

**with this above mentioned command mySQL prompt are showing.**



1. **Step 5: Create a database named tolldata.**

**create database tolldata;**



**Also check from the Browser Url also that db is created or not.**

[**https://kaushikdey19-8080.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai/**](https://kaushikdey19-8080.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai/)

Graphical user interface, text, application, email

Description automatically generated

1. **Step 6: Create a table named livetolldata with the schema to store the data generated by the traffic simulator.**

**use tolldata;**

**create table livetolldata(timestamp datetime,vehicle\_id int,vehicle\_type char(15),toll\_plaza\_id smallint);**

Text

Description automatically generated

**Also Check from the Browser Side:**

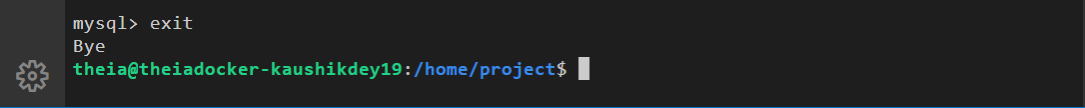
Graphical user interface, text, application

Description automatically generated

**This is the table where you would store all the streamed data that comes from kafka. Each row is a record of when a vehicle has passed through a certain toll plaza along with its type and anonymized id.**

1. **Step 7: Disconnect from MySQL server.**

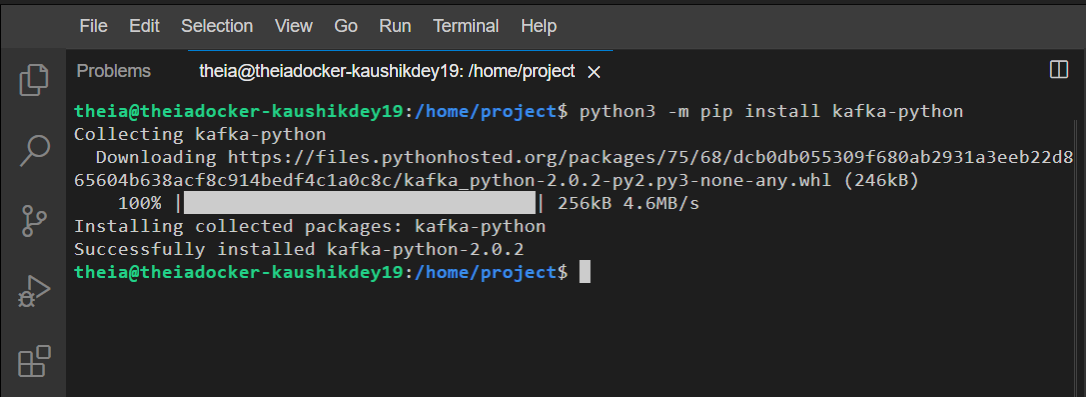
**Exit.**



**To clear the screen and take a fresh screenshot after that.**

1. **Step 8: Install the python module kafka-python using the pip command.**

**python3 -m pip install kafka-python.**



**This python module will help you to communicate with kafka server. It can used to send and receive messages from kafka.**

1. **Step 9: Install the python module mysql-connector-python using the pip command.**

**python3 -m pip install mysql-connector-python**

Graphical user interface, text, chat or text message

Description automatically generated

**This python module will help you to interact with mysql server.**

**===========================================================================**

# Exercise 2 – Start Kafka:

# Task 2.1 - Start Zookeeper

# Command for running the zookeeper => kafka\_2.12-2.8.0/bin/zookeeper-server-start.sh kafka\_2.12-2.8.0/config/zookeeper.properties ( client port is 2181) (start\_zookeeper.jpg)

# 

# Task 2.2 - Start Kafka Server

# Command for running kafka server

# kafka\_2.12-2.8.0/bin/kafka-server-start.sh kafka\_2.12-2.8.0/config/server.properties (start\_kafka.jpg)

Text

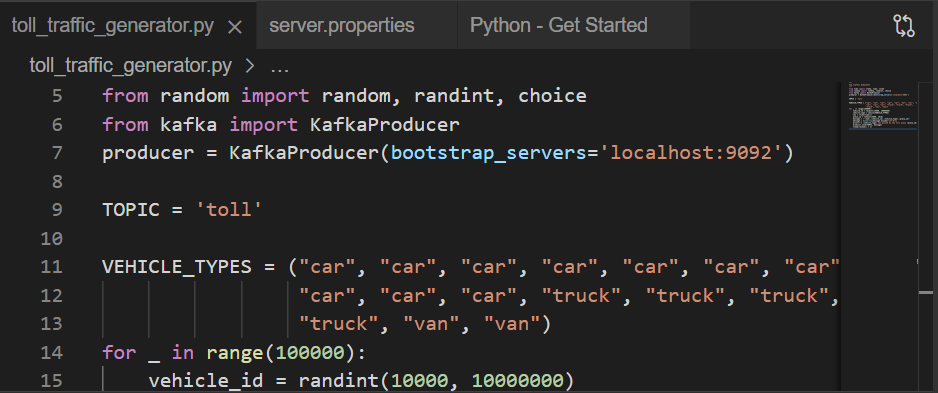
Description automatically generated

# Task 2.3 - Create a topic named toll

# Open the code using the their editor using the "Menu --> File -->Open" option.

# 

# topic named toll is created (create\_toll\_topic.jpg)



# Task 2.4 - Download the Toll Traffic Simulator

# First Download the toll traffic simulator.

# Download the toll traffic generator.py from the url below using 'wget'.

# Wget <https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0250EN-SkillsNetwork/labs/Final%20Assignment/toll_traffic_generator.py>

# Text Description automatically generated

# toll\_traffic\_generator.py (download\_simulator.jpg)

"""

Top Traffic Simulator

"""

from time import sleep, time, ctime

from random import random, randint, choice

from kafka import KafkaProducer

producer = KafkaProducer(bootstrap\_servers='localhost:9092')

TOPIC = 'set your topic here'

VEHICLE\_TYPES = ("car", "car", "car", "car", "car", "car", "car", "car",

                 "car", "car", "car", "truck", "truck", "truck",

                 "truck", "van", "van")

for \_ in range(100000):

    vehicle\_id = randint(10000, 10000000)

    vehicle\_type = choice(VEHICLE\_TYPES)

    now = ctime(time())

    plaza\_id = randint(4000, 4010)

    message = f"{now},{vehicle\_id},{vehicle\_type},{plaza\_id}"

    message = bytearray(message.encode("utf-8"))

    print(f"A {vehicle\_type} has passed by the toll plaza {plaza\_id} at {now}.")

    producer.send(TOPIC, message)

    sleep(random() \* 2)

**Task 2.5 : Configure the toll traffic simulator**

# Open the toll\_traffic\_generator.py and set the topic to toll. (configure\_simulator.jpg)

# Text Description automatically generated

# Task 2.6 - Run the Toll Traffic Simulator.

# Command=> python3 toll\_traffic\_generator.py. (simulator\_output.jpg)

# 

# Task 2.7 – Configure.

# Download the streaming\_data\_reader.py from the url below using 'wget'.

# Command = wget <https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0250EN-SkillsNetwork/labs/Final%20Assignment/streaming_data_reader.py>

# 

# Open the streaming\_data\_reader.py and modify the following details so that the program can connect to your mysql server. ((streaming\_reader\_code.jpg)

# TOPIC:toll

# DATABASE:tolldata

# USERNAME:root

# PASSWORD:MzAwNjYta2F1c2hp

# Text Description automatically generated

# Task 2.8 – Run

# Command : python3 streaming\_data\_reader.py. ( data\_reader\_output.jpg)

# Text Description automatically generated

# Task 2.9 - Health check of the streaming data pipeline.

# Now we are going to check our database where actually the data is inserted or not.

# [SELECT](https://kaushikdey19-8080.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) \* FROM `livetolldata`

# 