Safety Escort Service Database Management System

Milestone – Application in Python

Group 22

Veda Upasan Pedagadi
Satyajit Lanka

857-241-7760
857-395-1955
pedagadi.v@northeastern.edu
lanka.sat@northeastern.edu

Percentage of Effort Contributed by Veda: 50
Percentage of Effort Contributed by Satyajit: 50

Signature of Veda: VEDA
Signature of Satyajit: SATYAJIT

Submission Date: 04/01/2023

In [1]:
```
!pip install mysql-connector-python
```

```
Requirement already satisfied: mysql-connector-python in c:\users\veda1\anaconda3\lib\site-packages (8.0.32)
Requirement already satisfied: protobuf<=3.20.3,>=3.11.0 in c:\users\veda1\anaconda3\lib\site-packages (from mysql-connector-python) (3.20.3)
```

In [2]:
```
%pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\veda1\anaconda3\lib\site-packages (1.5.3)
Requirement already satisfied: numpy>=1.21.0 in c:\users\veda1\anaconda3\lib\site-packages (from pandas) (1.24.2)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\veda1\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\veda1\anaconda3\lib\site-packages (from pandas) (2022.7)
Requirement already satisfied: six>=1.5 in c:\users\veda1\anaconda3\lib\site-packages (from python-dateutil>=2.8.1->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [3]:
```
# Importing libraries
import mysql.connector
from mysql.connector import Error
import matplotlib.pyplot as plt
import pandas as pd
```

In [4]:
```
# Establish connection
try:
    connection = mysql.connector.connect(host='localhost',
                                         database='sesdb',
                                         user='root',
                                         password='root')
    if connection.is_connected():
        db_Info = connection.get_server_info()
        print("Connected to MySQL Server version ", db_Info)
        cursor = connection.cursor()
        cursor.execute("select database();")
        record = cursor.fetchone()
        print("You're connected to database: ", record)

except Error as e:
    print("Error while connecting to MySQL", e)
```

```
Connected to MySQL Server version  8.0.32
You're connected to database:  ('sesdb',)
```

In [5]:
```
    # Fetch Students details
    Query= "select * from student"
    cursor=connection.cursor()
    cursor.execute(Query)
    records=cursor.fetchall()
    for row in records:
        print(row)
```

```
(1, 'Farrand Pell', 'fpell0@sogou.com', 'H00DG8Sv', 481239621, '9 Iowa Parkway', 905, 'Green Ridge', 24741)
(2, 'Jeri Penchen', 'jpenchen1@boston.com', 'ruSOOcR6j2', 747577510, '407 Express Place', 899, 'Sloan', 20729)
(3, 'Elna Balmadier', 'ebalmadier2@homestead.com', 'cviAd3sX', 799184305, '77 Straubel Point', 3136, 'Buell', 26661)
(4, 'Estell Barlee', 'ebarlee3@google.nl', 'ZkatYfcCEQU', 557408756, '1087 Carey Place', 5210, 'Macpherson', 13950)
(5, 'Druci Davidofski', 'ddavidofski4@amazon.co.uk', 'ZzTpTyn', 759309149, '4 Messerschmidt Alley', 67545, 'John Wall', 29610)
(6, 'Trula Antcliff', 'tantcliff5@zcombinator.com', 'FVihBvSKSM', 646381130, '515 Northport Alley', 609, 'Pawling', 12102)
(7, 'Angeline Gooday', 'agooday6@businesswire.com', '8SY6p5', 386266901, '21972 Shelley Drive', 69, 'Leroy', 16361)
(8, 'Charyl Brunotti', 'cbrunotti7@vk.com', 'nOYywAk9P', 208399648, '52988 Stoughton Road', 70168, 'Crowley', 27037)
(9, 'Tawsha Cuss', 'tcuss8@mlb.com', 'eajMJe', 706453634, '0 Ilene Drive', 186, 'Kenwood', 14237)
(10, 'Carmita Backshaw', 'cbackshaw9@photobucket.com', 'FAA8EAI', 224173129, '516 Macpherson Trail', 91, 'Paget', 27007)
```

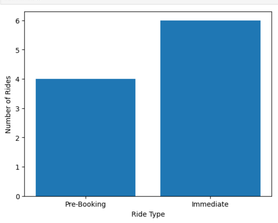
In [6]:
```
#Get the names and contact information of all drivers who are currently busy
sql_select_Query= "SELECT Dname, Dphone, Demail FROM Driver WHERE Dstatus = 'busy'"
cursor=connection.cursor()
cursor.execute(sql_select_Query)
records=cursor.fetchall()
for row in records:
    print(row)
```

```
('Hort Pollock', 786661168, 'hpollock1@nugedomains.com')
('Alexandros Makeswell', 644919098, 'amakeswell3@apple.com')
('Bard Drinkeld', 649084910, 'bdrinkeld4@tamu.edu')
('Omero Langhorne', 948827689, 'olanghorne5@go.com')
```

In [7]:
```
# fetch notifications table
sql_select_Query1= "select * from notifications;"
cursor=connection.cursor()
cursor.execute(sql_select_Query1)
records=cursor.fetchall()
for row in records:
    print(row)
```

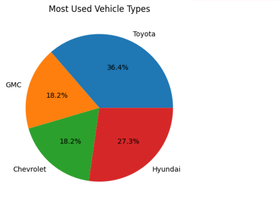
```
(1, 'Booked')
(2, 'Ride on the way')
(3, 'Arrived')
(4, 'Cancelled')
```

In [8]:
```
# Ride Type

sql_select_Query= "SELECT R_type, COUNT(*) as num FROM ride GROUP BY R_type"
cursor=connection.cursor()
cursor.execute(sql_select_Query)
data=cursor.fetchall()

# create lists to store the x and y values
x_values = [row[0] for row in data]
y_values = [row[1] for row in data]

# plot the bar graph
plt.bar(x_values, y_values)

# set the x and y labels
plt.xlabel('Ride Type')
plt.ylabel('Number of Rides')

# show the plot
plt.show()
```

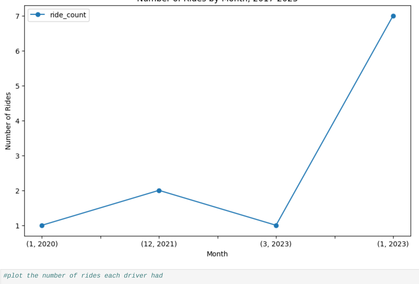


In [9]:
```
#count of rides for each vehicle, along with its type

query = """
    SELECT vh.V_model, COUNT(*) AS ride_count
    FROM ride_history rh
    JOIN vehicle vh ON rh.V_ID = vh.V_ID
    GROUP BY vh.V_model;
"""
df = pd.read_sql(query, connection)

# Create a pie chart
plt.pie(df['ride_count'], labels=df['V_model'], autopct='%1.1f%%')
plt.title('Most Used Vehicle Types')
plt.show()
```

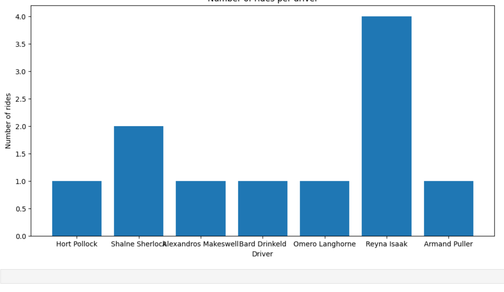
```
C:\Users\veda1\AppData\Local\Temp\ipykernel_6940\3566762683.py:9: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  df = pd.read_sql(query, connection)
```



In [10]:
```
## plot the number of rides in each month between 2017 and 2023

# construct SQL query to select ride count by month
query = """
    SELECT
        YEAR(RH_stime) AS year,
        MONTH(RH_stime) AS month,
        COUNT(*) AS ride_count
    FROM ride_history
    WHERE RH_stime >= '2017-01-01' AND RH_etime <= '2023-12-31'
    GROUP BY YEAR(RH_stime), MONTH(RH_stime)
"""

# execute query and fetch results
cursor=connection.cursor()
cursor.execute(query)
results=cursor.fetchall()

# convert results to Pandas DataFrame
df = pd.DataFrame(results, columns=['year', 'month', 'ride_count'])
df = df.sort_values(by='year', ascending=True)

# create a new datetime column from year and month columns
df['date'] = pd.to_datetime(df['month','year']).assign(day=1)

# set the date column as the DataFrame index
df.set_index(['month','year'], inplace=True)

# create a bar plot of ride count by month
ax = df.plot(kind='line', y='ride_count', figsize=(10, 6),marker='o')

# set plot title and axis labels
ax.set_title('Number of Rides by Month, 2017-2023')
ax.set_xlabel('Month')
ax.set_ylabel('Number of Rides')

# display the plot
plt.show()
```



In [11]:
```
#plot the number of rides each driver had

# execute the SQL query
query = """
SELECT d.Dname, COUNT(*) AS num_rides
FROM ride_history r
JOIN driver d ON r.D_ID = d.D_ID
GROUP BY d.Dname
"""

cursor = connection.cursor()
cursor.execute(query)

# extract the results
results = cursor.fetchall()

# extract the driver names and number of rides into separate lists
driver_names = [result[0] for result in results]
num_rides = [result[1] for result in results]

# create the bar plot
plt.figure(figsize=(12,6))
plt.bar(driver_names, num_rides)
plt.xlabel('Driver')
plt.ylabel('Number of rides')
plt.title('Number of rides per driver')
plt.show()
```



In [ ]:
```
```