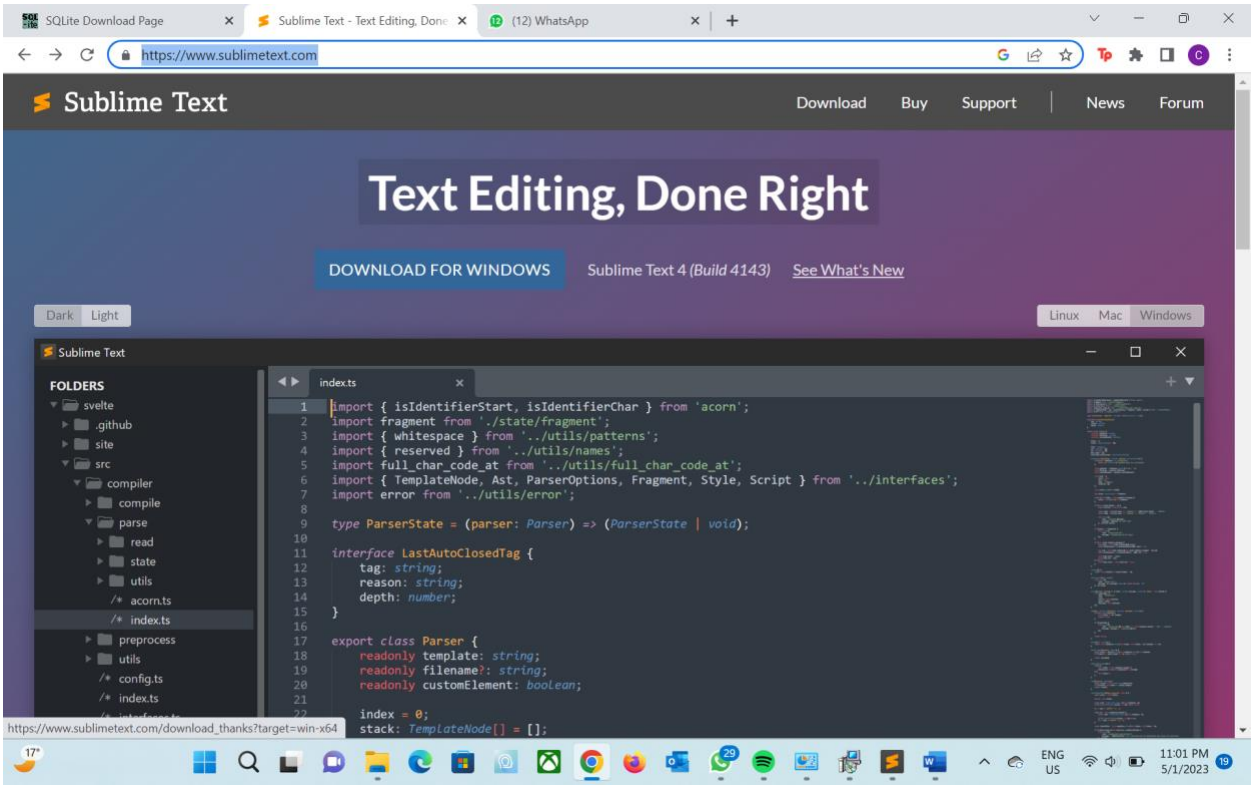
Python GUI Developing tool:

Sublime Text is a popular text editor used for programming and code editing. It is known for its minimalist interface, powerful customization options, and extensive package ecosystem. Sublime Text supports many programming languages and features syntax highlighting, code autocompletion, and a range of productivity features to improve coding speed and accuracy. It is available for Windows, macOS, and Linux operating systems.

Download link : https://www.sublimetext.com/

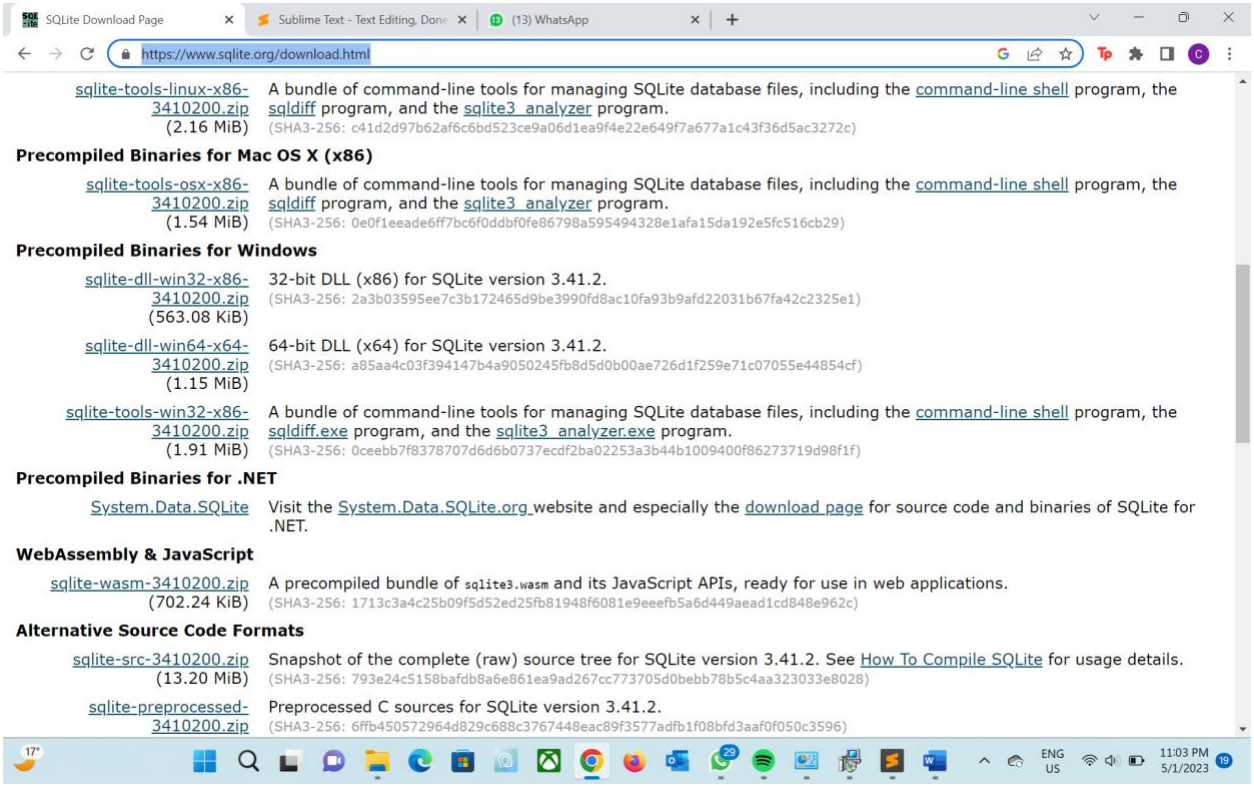
Here, click on ***Download for Windows***



SQLITE3:

SQLite3 is a software library that provides a relational database management system. It is embedded into various operating systems and applications, and it allows developers to implement a compact, self-contained, and serverless database engine. SQLite3 is commonly used in mobile apps, web browsers, and other software that requires a small, fast, and reliable database solution. It uses a SQL-based syntax for data manipulation and retrieval and is considered one of the most widely deployed database engines in the world.

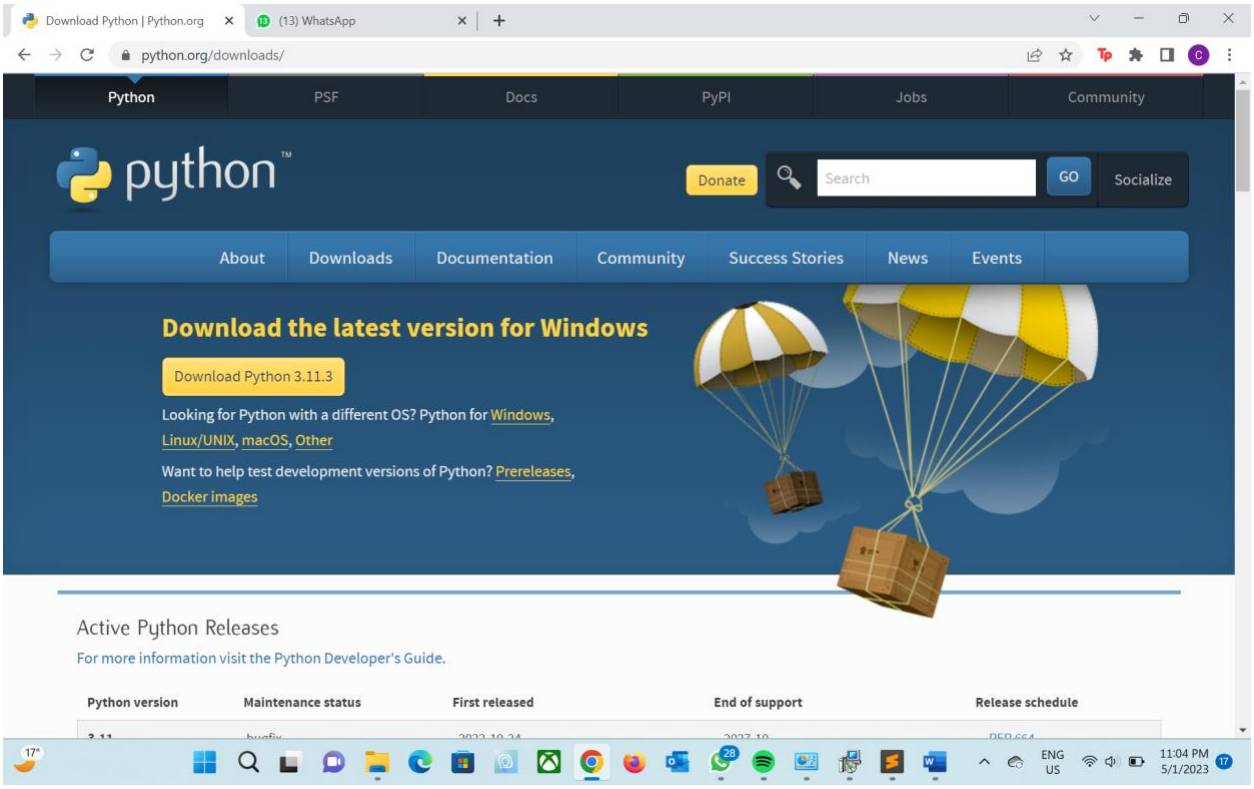
Download link: https://www.sqlite.org/download.html



Python Installation:

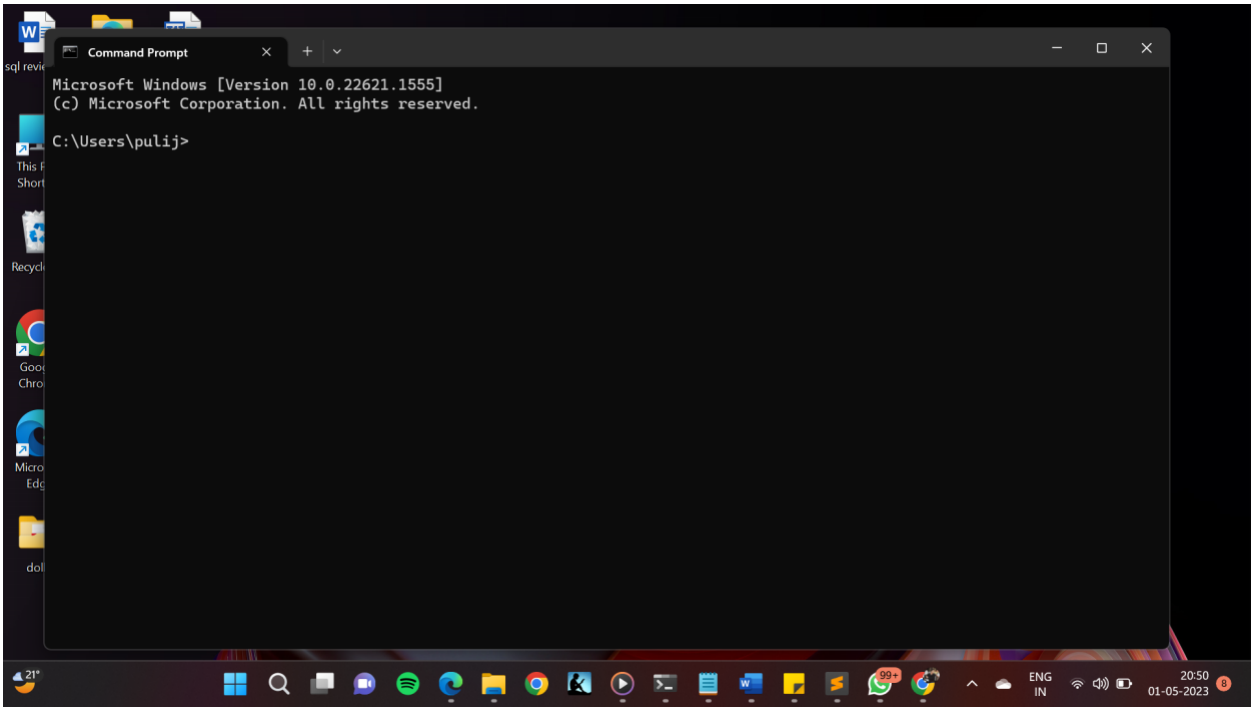
Python must be installed on to your Windows as we are using Python GUI for implementation.

Download link: https://www.python.org/downloads/



Command prompt:

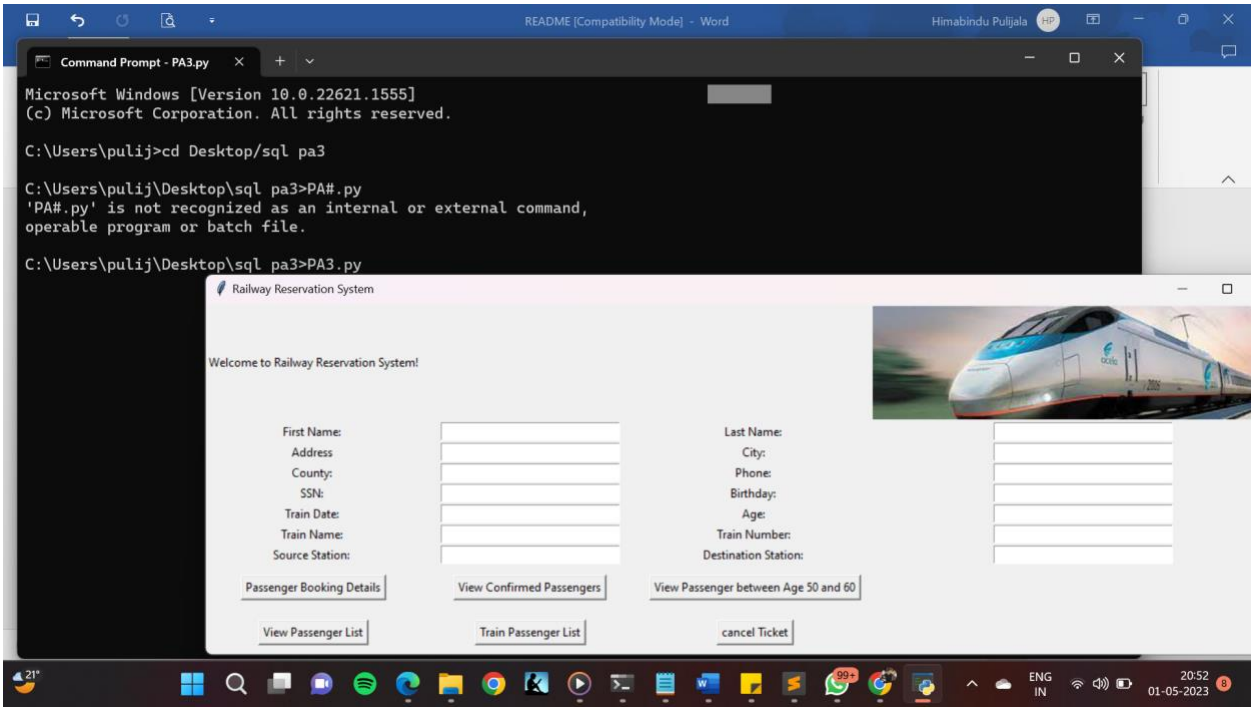
Through command prompt we will be directed to the folder where our project is placed. Our project is in the placed in the path ‘C:\Users\pulij\Desktop\sql pa3’.



After going into the folder, we need to give command as > python3 PA3.py

`python3` is the command used to run Python version 3 and later on the terminal or command prompt. It is used to execute Python scripts or open a Python interpreter shell to interactively write and run Python code.

PA3.py is the name our programming file. Once successfully execute our code without any errors, the GUO window will be opened as shown.



Queries:

SQL Query 1:

SELECT Train.'Train Name', Train.'Train Number', booked.Ticket_Type, Booked.Status

from train

join booked on Train.'Train Number' = Booked.Train_Number

join passenger on passenger.SSN = Booked.Passanger_ssn

where Passenger.last_name = ? and passenger.first_name = ? ;

SQL QUERY 2:

```
SELECT distinct passenger.first_name, passenger.last_name, Booked.Ticket_Type,
Booked.Status
FROM Passenger, Booked,Train,Train_status
Where passenger.SSN = Booked.Passanger_ssn
and Booked.Train_Number = Train.'Train Number'
And Booked.Status = 'Booked'
AND Train_status.TrainDate = ? ;
```

SQL QUERY 3:

```
SELECT Train.'Train Number', Train.'Train Name', Train.'Source Station', Train.'Destination
Station', Passenger.first_name, Passenger.last_name, Passenger.address, Booked.Ticket_Type,
Booked.Status FROM Booked INNER JOIN Passenger ON Booked.Passanger_ssn =
Passenger.SSN INNER JOIN Train ON Booked.Train_Number = Train.'Train Number' WHERE
strftime('%Y', "now") - CAST(substr(Passenger.bdate,7,4) as INTEGER) BETWEEN ? AND ?
ORDER BY Train.'Train Number
```

SQL QUERY 4:

```
SELECT Train.'Train Name', COUNT(Booked.Passanger_ssn) AS Passenger_Count

FROM Train JOIN Booked ON Train.'Train Number' = Booked.Train_Number

GROUP BY Train.'Train Name';
```

SQL Query 5:

```
SELECT Passenger.first_name, Passenger.last_name, Booked.Ticket_Type

FROM Booked, passenger, train

where Booked.Train_Number = Train.'Train Number' and Booked.Passanger_ssn =

Passenger.SSN and Train.'Train Name' = ? AND Booked.Status = 'Booked';
```

SQL QUERY 6:

```
SELECT * FROM Booked,passenger WHERE Passanger_ssn = ? AND Train_Number = ? AND

Status = 'Booked'",(ssn.get(), TrainNumber.get(), ))

    cancelTicket_cur.execute("DELETE FROM Booked,passenger WHERE Passanger_ssn = ?

AND Train_Number = ? ",(ssn.get(),TrainNumber.get(),))

    if ticket_type == 'Premium':

        cancelTicket_cur.execute("UPDATE Booked SET Status = 'Booked' WHERE

Train_Number = ? AND Ticket_Type = 'Premium' AND Status = 'WaitL' AND Passanger_ssn IN

(SELECT Passanger_ssn FROM Booked WHERE Train_Number = ? AND Ticket_Type =

'Premium' AND Status = 'WaitL' ORDER BY Train_Number LIMIT

1)",(TrainNumber.get(),TrainNumber.get(),))
```

else:

cancelTicket_cur.execute("UPDATE Booked SET Status = 'Booked' WHERE Train_Number = ? AND Ticket_Type = 'General' AND Status = 'WaitL' AND Passanger_ssn IN (SELECT Passanger_ssn FROM Booked WHERE Train_Number = ? AND Ticket_Type = 'General' AND Status = 'WaitL' ORDER BY Train_Number LIMIT 1