

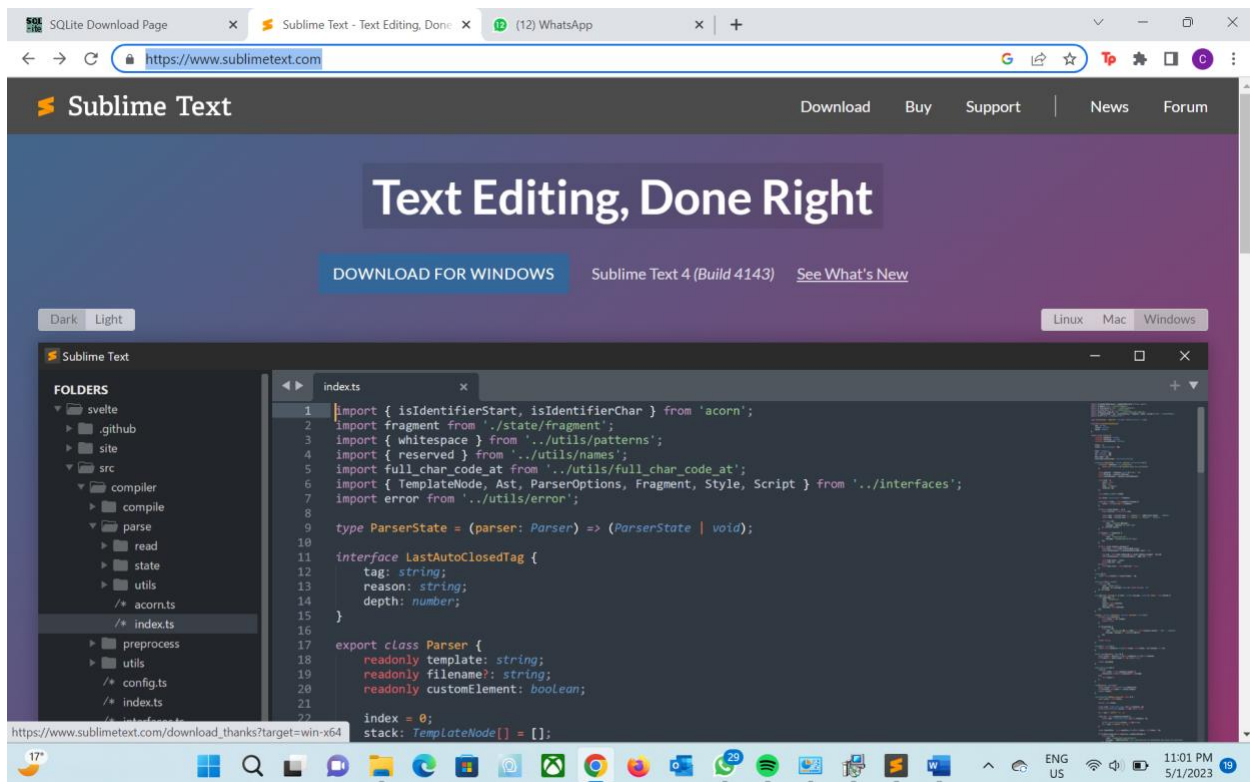
Python GUI Developing tool:

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/>

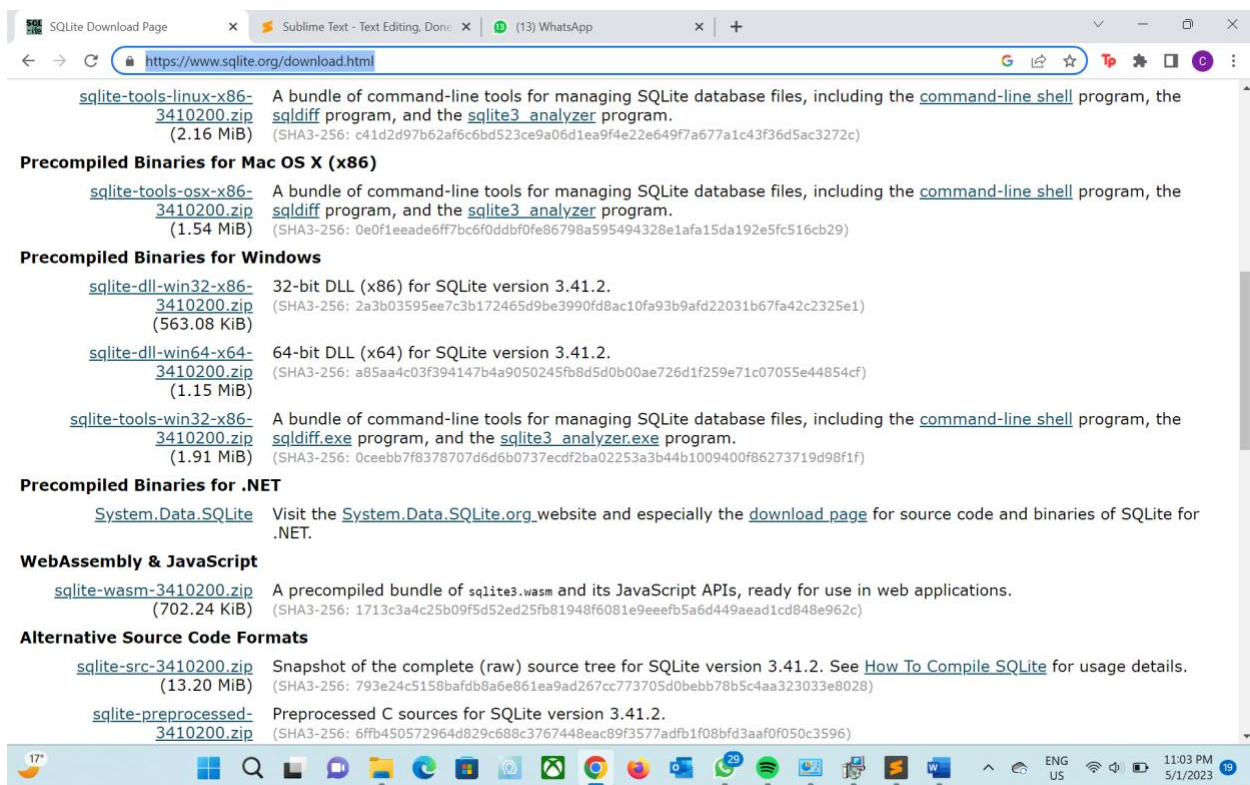
Here, click on **Download for Windows**



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>



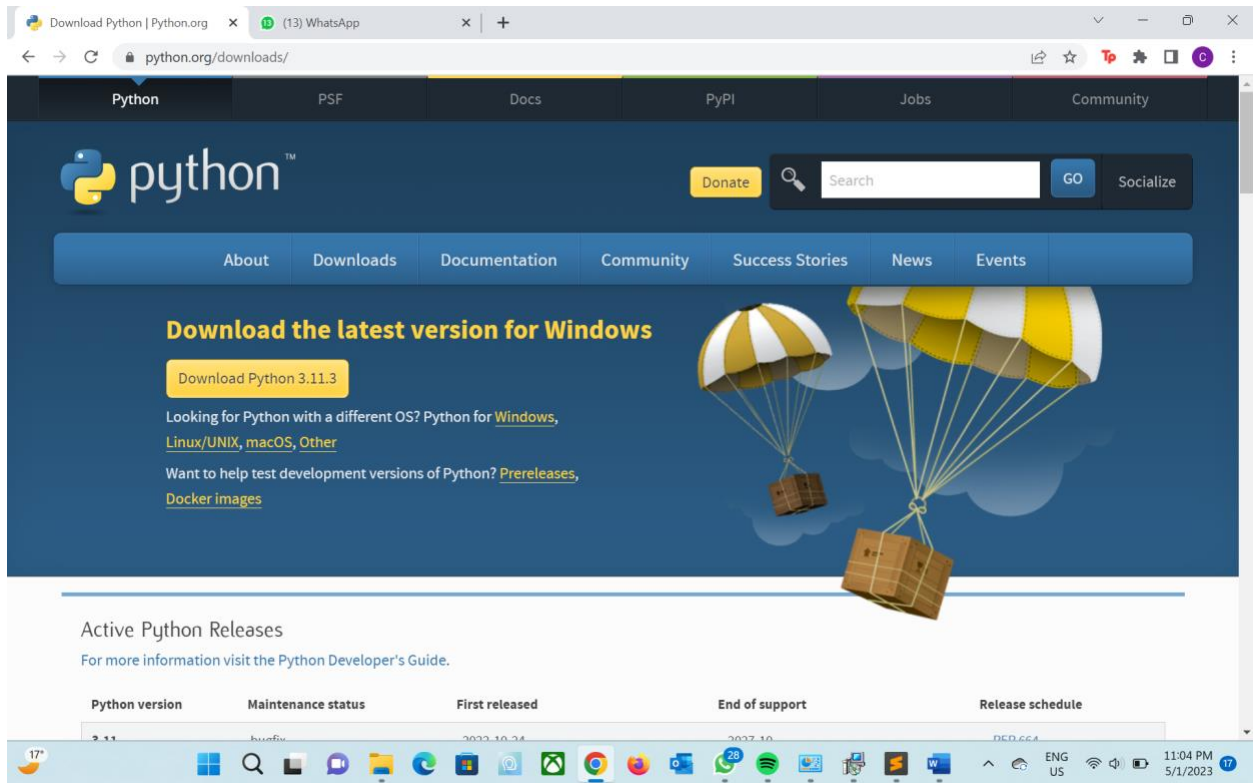
The screenshot shows a web browser window with the address bar displaying <https://www.sqlite.org/download.html>. The page content is as follows:

- [sqlite-tools-linux-x86-3410200.zip](#) (2.16 MiB)
A bundle of command-line tools for managing SQLite database files, including the [command-line shell](#) program, the [sqldiff](#) program, and the [sqlite3_analyzer](#) program.
(SHA3-256: c41d2d97b62af6c6bd523ce9a06d1ea9f4e22e649f7a677a1c43f36d5ac3272c)
- Precompiled Binaries for Mac OS X (x86)**
 - [sqlite-tools-osx-x86-3410200.zip](#) (1.54 MiB)
A bundle of command-line tools for managing SQLite database files, including the [command-line shell](#) program, the [sqldiff](#) program, and the [sqlite3_analyzer](#) program.
(SHA3-256: 0e0f1eeade6ff7bc6f0ddb0fe86798a595494328e1afa15da192e5fc516cb29)
- Precompiled Binaries for Windows**
 - [sqlite-dll-win32-x86-3410200.zip](#) (563.08 KiB)
32-bit DLL (x86) for SQLite version 3.41.2.
(SHA3-256: 2a3b03595ee7c3b172465d9be3990fd8ac10fa93b9afd22031b67fa42c2325e1)
 - [sqlite-dll-win64-x64-3410200.zip](#) (1.15 MiB)
64-bit DLL (x64) for SQLite version 3.41.2.
(SHA3-256: a85aa4c03f394147b4a9050245fb8d5d0b00ae726d1f259e71c07055e44854cf)
 - [sqlite-tools-win32-x86-3410200.zip](#) (1.91 MiB)
A bundle of command-line tools for managing SQLite database files, including the [command-line shell](#) program, the [sqldiff.exe](#) program, and the [sqlite3_analyzer.exe](#) program.
(SHA3-256: 0ceebb7f8378707d6d6b0737ecd2ba02253a3b44b1009400f86273719d98f1f)
- Precompiled Binaries for .NET**
 - [System.Data.SQLite](#)
Visit the [System.Data.SQLite.org](#) website and especially the [download page](#) for source code and binaries of SQLite for .NET.
- WebAssembly & JavaScript**
 - [sqlite-wasm-3410200.zip](#) (702.24 KiB)
A precompiled bundle of [sqlite3.wasm](#) and its JavaScript APIs, ready for use in web applications.
(SHA3-256: 1713c3a4c25b09f5d52ed25fb81948f6081e9eeefb5a6d449aead1cd848e962c)
- Alternative Source Code Formats**
 - [sqlite-src-3410200.zip](#) (13.20 MiB)
Snapshot of the complete (raw) source tree for SQLite version 3.41.2. See [How To Compile SQLite](#) for usage details.
(SHA3-256: 793e24c5158bafdb8a6e861ea9ad267cc773705d0bebb78b5c4aa323033e8028)
 - [sqlite-preprocessed-3410200.zip](#)
Preprocessed C sources for SQLite version 3.41.2.
(SHA3-256: 6ffb450572964d829c688c3767448eac89f3577adfb1f08bfd3aaf0f050c3596)

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/>



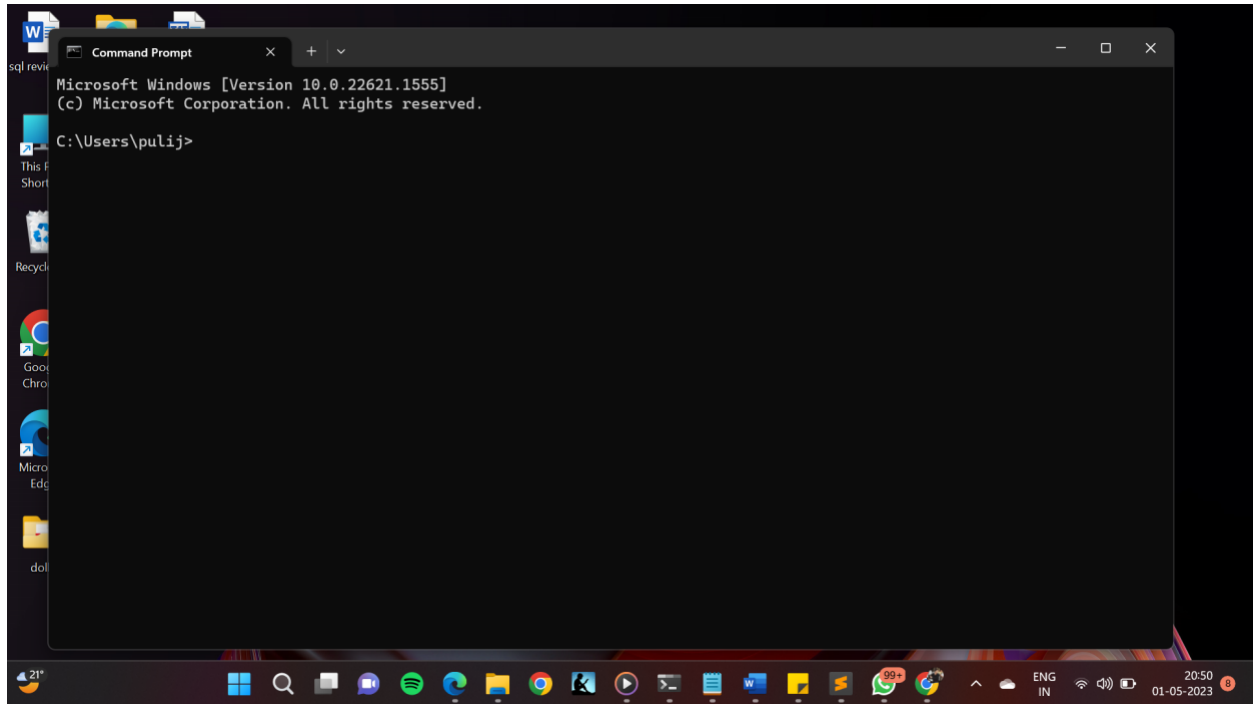
The screenshot shows the Python.org website's download page. The browser's address bar displays 'python.org/downloads/'. The website's navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below this, a secondary navigation bar contains links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features the Python logo, a 'Donate' button, a search bar, and a 'Socialize' button. A prominent yellow button labeled 'Download Python 3.11.3' is displayed, followed by links for other operating systems: 'Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [macOS](#), [Other](#)'. Below this, there are links for 'Want to help test development versions of Python? [Prereleases](#), [Docker images](#)'. The page also includes an illustration of two parachutes carrying boxes. At the bottom, there is a section titled 'Active Python Releases' with a link to the 'Python Developer's Guide'. Below this is a table with columns for 'Python version', 'Maintenance status', 'First released', 'End of support', and 'Release schedule'. The table shows the following data:

Python version	Maintenance status	First released	End of support	Release schedule
3.11	beta	2022-10-28	2027-10-28	DEC 664

The Windows taskbar at the bottom shows the system clock as 11:04 PM on 5/1/2023, with a temperature of 17°C and various application icons.

Command prompt:

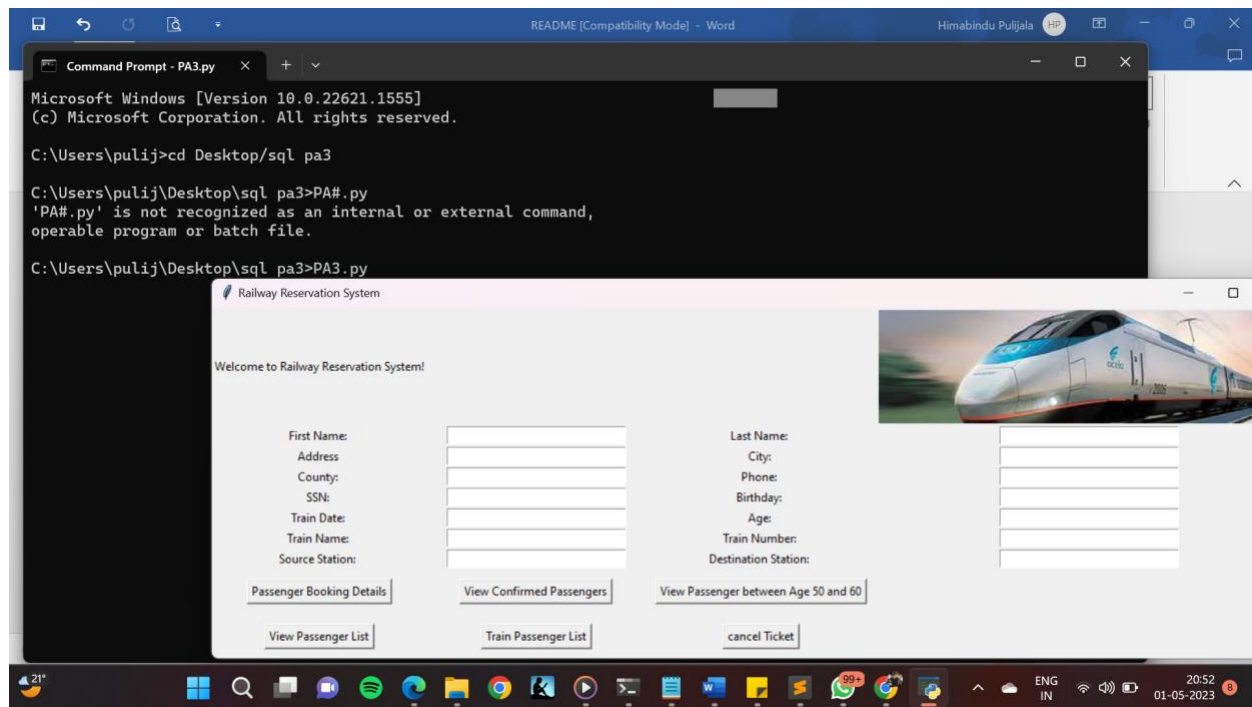
Through command prompt we will be directed to the folder where our project is placed. Our project is placed in the path 'C:\Users\pulij\Desktop\sql pa3'.



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 of our programming file. Once successfully executed, our code runs without any errors, and the GUI 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
```