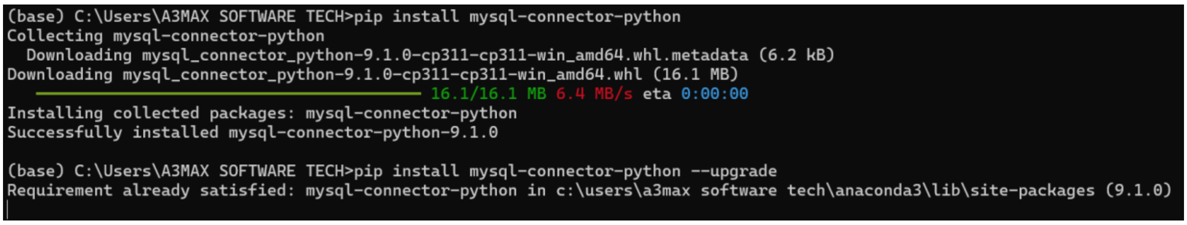
Python to MySQL Connection (Student Registration Project) - By Sunitha Mekala

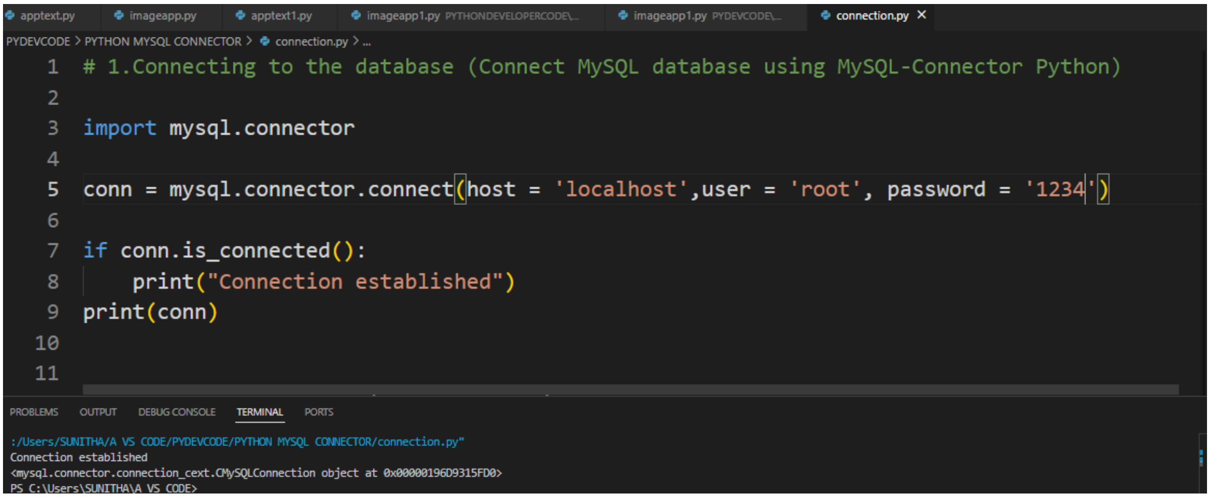
1) **How to install MySQL connector package in Python:**
   mysql-connector method can be installed on Windows with the use of following command:

   pip install mysql-connector-python



```
(base) C:\Users\A3MAX SOFTWARE TECH>pip install mysql-connector-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-9.1.0-cp311-cp311-win_amd64.whl.metadata (6.2 kB)
Downloading mysql_connector_python-9.1.0-cp311-cp311-win_amd64.whl (16.1 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 16.1/16.1 MB 6.4 MB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.1.0

(base) C:\Users\A3MAX SOFTWARE TECH>pip install mysql-connector-python --upgrade
Requirement already satisfied: mysql-connector-python in c:\users\a3max software tech\anaconda3\lib\site-packages (9.1.0)
```

2) **Connect MySQL database using MySQL-Connector Python:**
   Create a new file in VSCode **– connection.py**

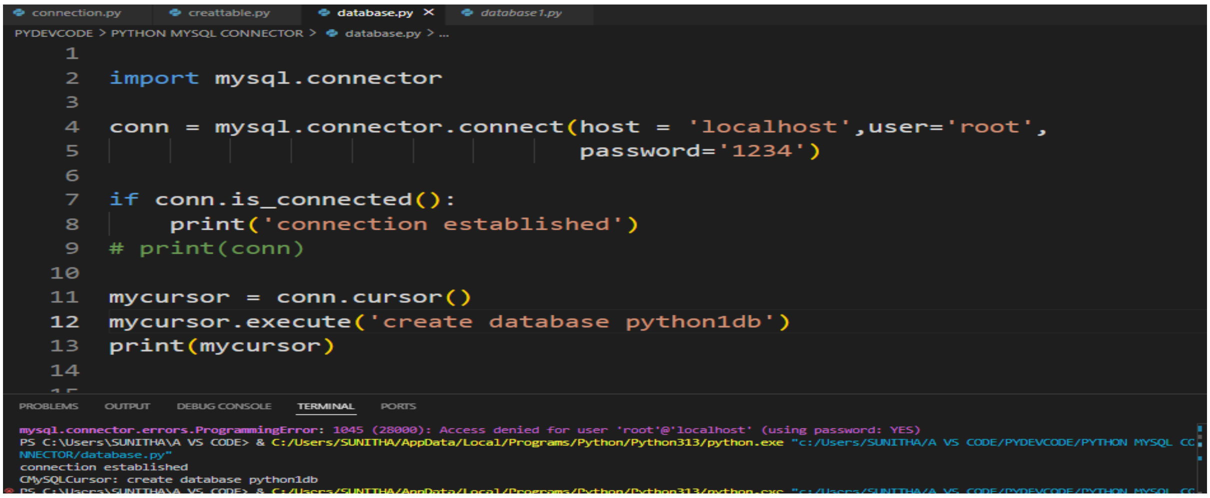


apptext.py
imageapp.py
apptext1.py
imageapp1.py PYTHONDEVELOPERCODE\...
imageapp1.py PYDEVCODE\...
connection.py
PYDEVCODE > PYTHON MYSQL CONNECTOR > connection.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS

```python
# 1.Connecting to the database (Connect MySQL database using MySQL-Connector Python)

import mysql.connector

conn = mysql.connector.connect(host = 'localhost',user = 'root', password = '1234')

if conn.is_connected():
    print("Connection established")
print(conn)
```

```
:/Users/SUNITHA/A VS CODE/PYDEVCODE/PYTHON MYSQL CONNECTOR/connection.py"
Connection established
<mysql.connector.connection_cext.CMySQLConnection object at 0x00000196D9315FD0>
PS C:\Users\SUNITHA\A VS CODE>
```

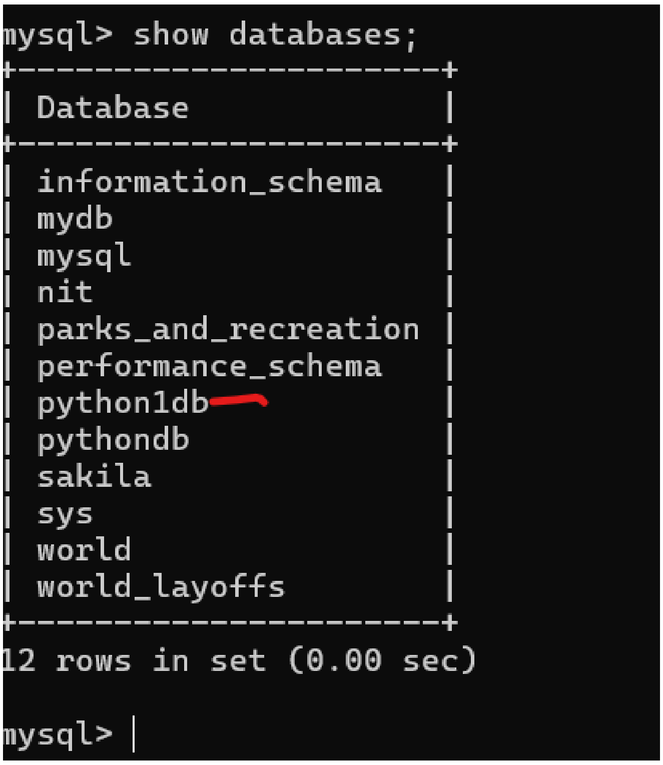
3) Save the below code in a new file – **database.py** for Creating a database from python code – **python1db**



```python

import mysql.connector

conn = mysql.connector.connect(host = 'localhost',user='root',
                               password='1234')

if conn.is_connected():
    print('connection established')
# print(conn)

mycursor = conn.cursor()
mycursor.execute('create database python1db')
print(mycursor)

```

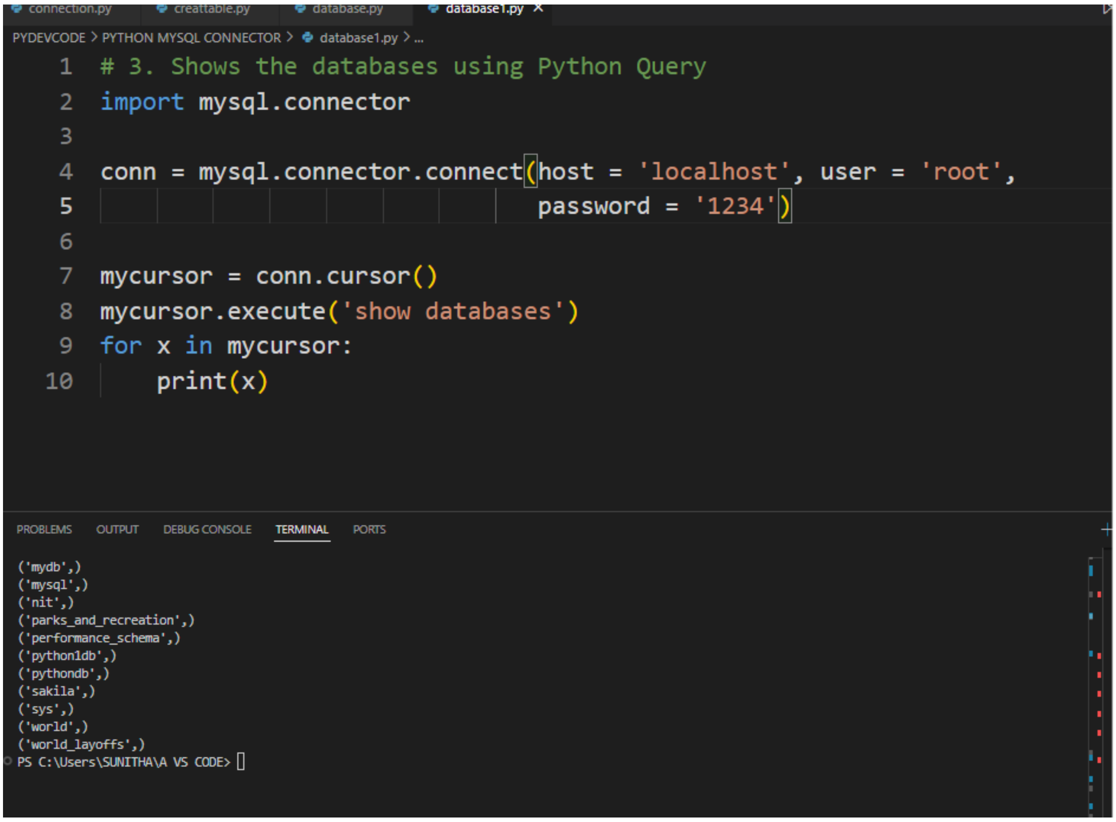
```
mysql.connector.errors.ProgrammingError: 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
PS C:\Users\SUNITHA\A VS CODE> & C:/Users/SUNITHA/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/SUNITHA/A VS CODE/PYDEVCODE/PYTHON MYSQL CO
NNECTOR/database.py"
connection established
CMySQLCursor: create database python1db
PS C:\Users\SUNITHA\A VS CODE> & C:/Users/SUNITHA/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/SUNITHA/A VS CODE/PYDEVCODE/PYTHON MYSQL CO
```

Manually check if our new database - **python1db** is created or not (Here its created)



```
mysql> show databases;
+-----------------------+
| Database              |
+-----------------------+
| information_schema    |
| mydb                  |
| mysql                 |
| nit                   |
| parks_and_recreation  |
| performance_schema    |
| python1db             |
| pythondb              |
| sakila                |
| sys                   |
| world                 |
| world_layoffs         |
+-----------------------+
12 rows in set (0.00 sec)

mysql> 
```

4) Shows the list of databases present – creating a new file name – **database1.py**

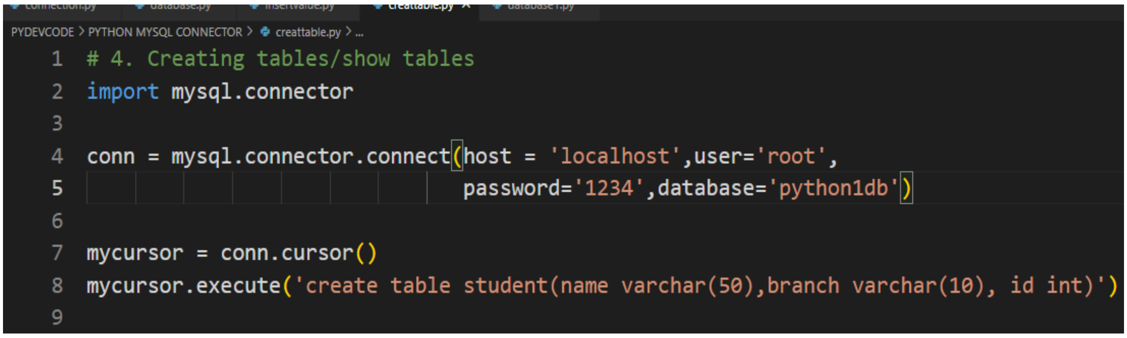


```python
# 3. Shows the databases using Python Query
import mysql.connector

conn = mysql.connector.connect(host = 'localhost', user = 'root',
                               password = '1234')

mycursor = conn.cursor()
mycursor.execute('show databases')
for x in mycursor:
    print(x)
```

```
('mydb',)
('mysql',)
('nit',)
('parks_and_recreation',)
('performance_schema',)
('python1db',)
('pythondb',)
('sakila',)
('sys',)
('world',)
('world_layoffs',)
PS C:\Users\SUNITHA\A VS CODE> 
```

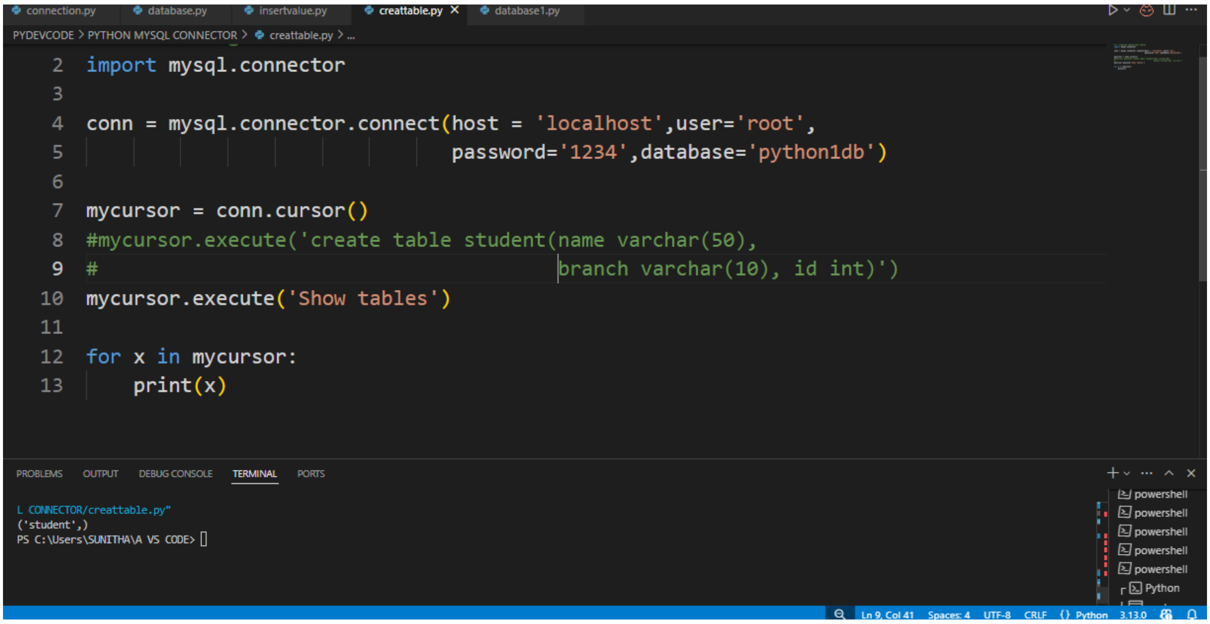
5) Creating tables and showing tables – for this create a file named - **creattable.py**

a) Creating the table student



```python
# 4. Creating tables/show tables
import mysql.connector

conn = mysql.connector.connect(host = 'localhost',user='root',
                               password='1234',database='python1db')

mycursor = conn.cursor()
mycursor.execute('create table student(name varchar(50),branch varchar(10), id int)')

```

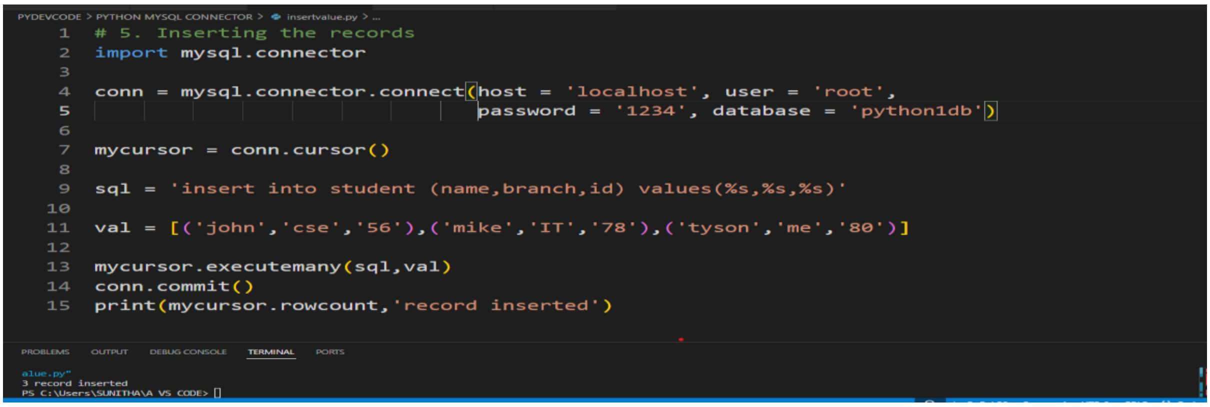
b) Once the table is created we can see the tables in the sql



```python
import mysql.connector

conn = mysql.connector.connect(host = 'localhost',user='root',
                               password='1234',database='python1db')

mycursor = conn.cursor()
#mycursor.execute('create table student(name varchar(50),
#                                       branch varchar(10), id int)')
mycursor.execute('Show tables')

for x in mycursor:
    print(x)
```

```
L CONNECTOR/creattable.py"
('student',)
PS C:\Users\SUNITHA\A VS CODE> 
```

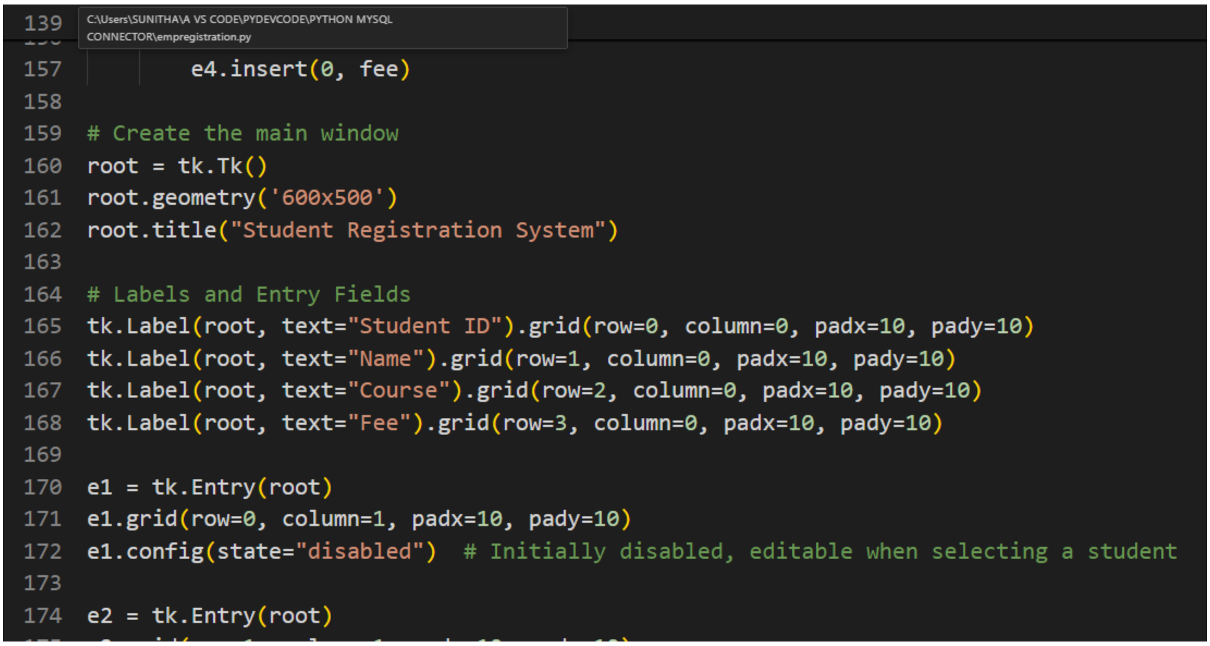
6) Inserting the records into student table – file name – **insertvalue.py**



```python
# 5. Inserting the records
import mysql.connector

conn = mysql.connector.connect(host = 'localhost', user = 'root',
                               password = '1234', database = 'python1db')

mycursor = conn.cursor()

sql = 'insert into student (name,branch,id) values(%s,%s,%s)'

val = [('john','cse','56'),('mike','IT','78'),('tyson','me','80')]

mycursor.executemany(sql,val)
conn.commit()
print(mycursor.rowcount,'record inserted')
```

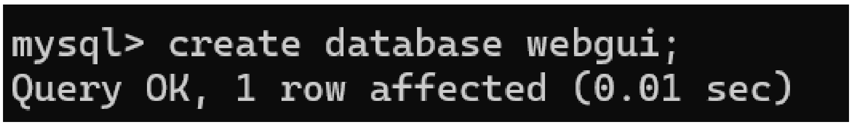
```
alue.py"
3 record inserted
PS C:\Users\SUNITHA\A VS CODE> 
```

7.) Now we are working on creating a student registration form
Save with **studregistration.py file**

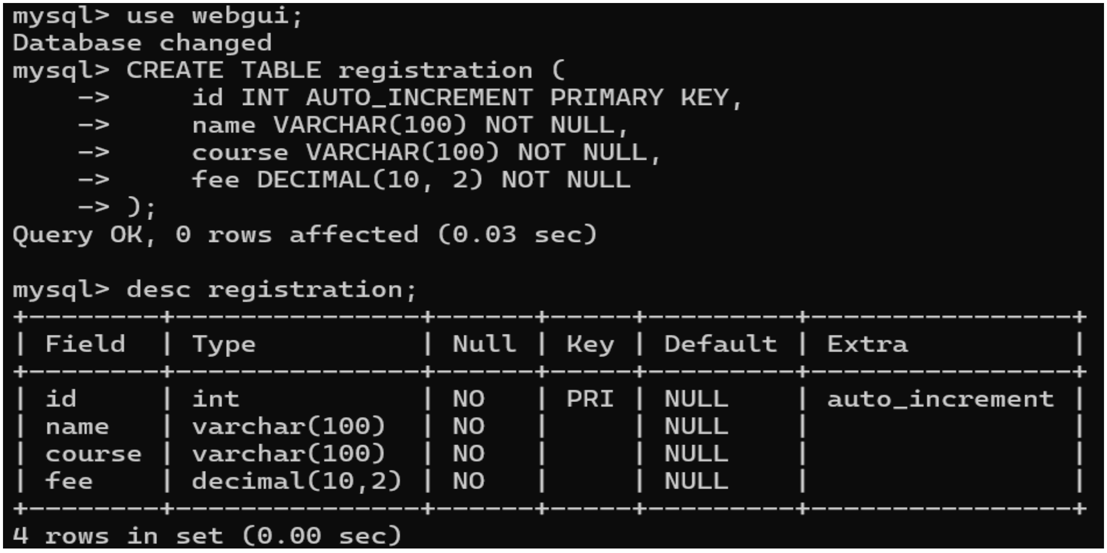

C:\Users\SUNITHA\A VS CODE\PYDEVCODE\PYTHON MYSQL CONNECTOR\empregistration.py

```
        e4.insert(0, fee)

# Create the main window
root = tk.Tk()
root.geometry('600x500')
root.title("Student Registration System")

# Labels and Entry Fields
tk.Label(root, text="Student ID").grid(row=0, column=0, padx=10, pady=10)
tk.Label(root, text="Name").grid(row=1, column=0, padx=10, pady=10)
tk.Label(root, text="Course").grid(row=2, column=0, padx=10, pady=10)
tk.Label(root, text="Fee").grid(row=3, column=0, padx=10, pady=10)

e1 = tk.Entry(root)
e1.grid(row=0, column=1, padx=10, pady=10)
e1.config(state="disabled")  # Initially disabled, editable when selecting a student

e2 = tk.Entry(root)
```

After that create the table and insert values,

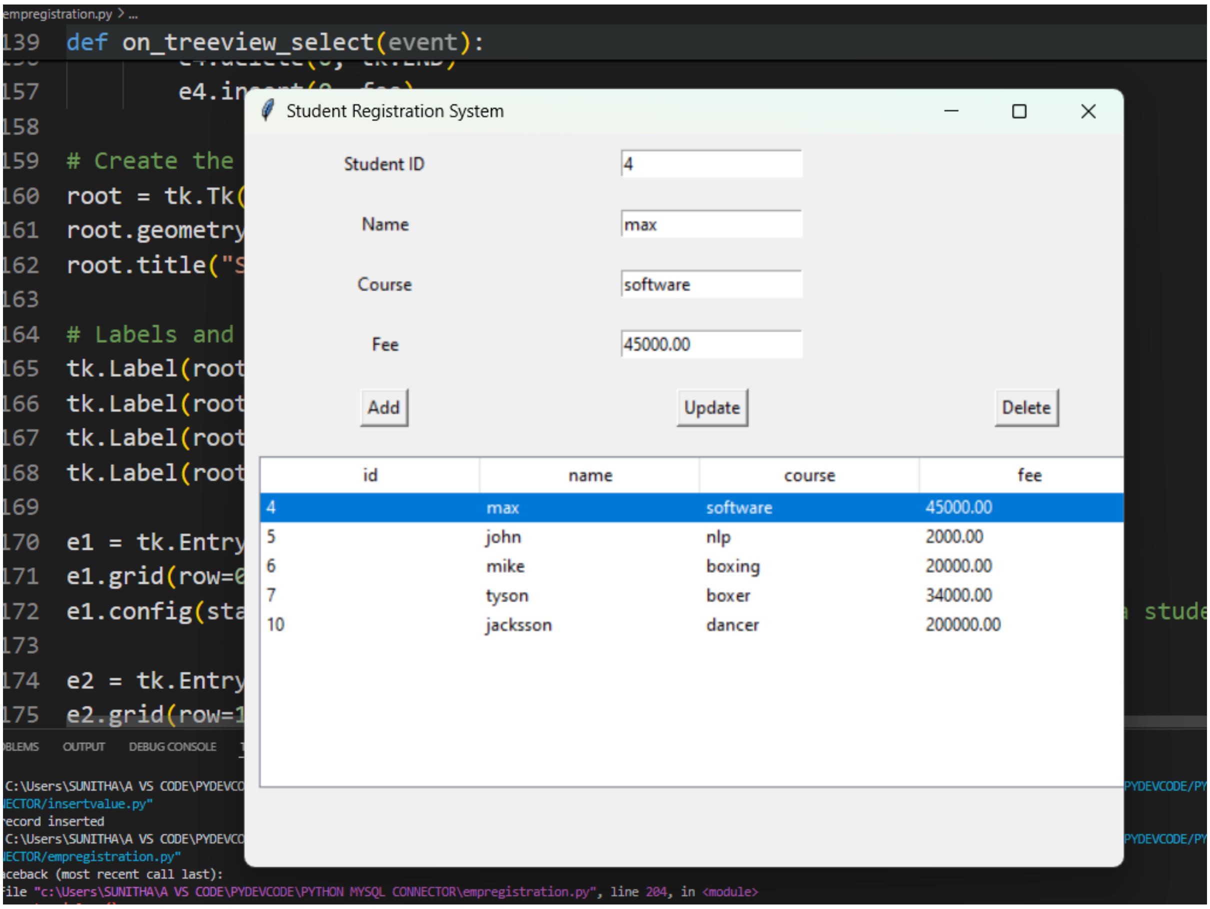


```
mysql> create database webgui;
Query OK, 1 row affected (0.01 sec)
```



```
mysql> use webgui;
Database changed
mysql> CREATE TABLE registration (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(100) NOT NULL,
    ->     course VARCHAR(100) NOT NULL,
    ->     fee DECIMAL(10, 2) NOT NULL
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> desc registration;
+--------+---------------+------+-----+---------+----------------+
| Field  | Type          | Null | Key | Default | Extra          |
+--------+---------------+------+-----+---------+----------------+
| id     | int           | NO   | PRI | NULL    | auto_increment |
| name   | varchar(100)  | NO   |     | NULL    |                |
| course | varchar(100)  | NO   |     | NULL    |                |
| fee    | decimal(10,2) | NO   |     | NULL    |                |
+--------+---------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
```

Once you write the backend code then below query will execute using **tkinter frontend**



empregistration.py > ...

```
def on_treeview_select(event):
        e4.in
# Create the
root = tk.Tk(
root.geometry
root.title("S

# Labels and
tk.Label(root
tk.Label(root
tk.Label(root
tk.Label(root

e1 = tk.Entry
e1.grid(row=0
e1.config(sta                                                          stude

e2 = tk.Entry
e2.grid(row=1
```

```
C:\Users\SUNITHA\A VS CODE\PYDEVCO                                      PYDEVCODE/PY
IECTOR/insertvalue.py"
ecord inserted
C:\Users\SUNITHA\A VS CODE\PYDEVCO                                      PYDEVCODE/PY
IECTOR/empregistration.py"
ceback (most recent call last):
ile "c:\Users\SUNITHA\A VS CODE\PYDEVCODE\PYTHON MYSQL CONNECTOR\empregistration.py", line 204, in <module>
```

Student Registration System

Student ID	4
Name	max
Course	software
Fee	45000.00

id	name	course	fee
4	max	software	45000.00
5	john	nlp	2000.00
6	mike	boxing	20000.00
7	tyson	boxer	34000.00
10	jacksson	dancer	200000.00

Then, we can check in the webgui database to see if the records are inserted into the registration table.