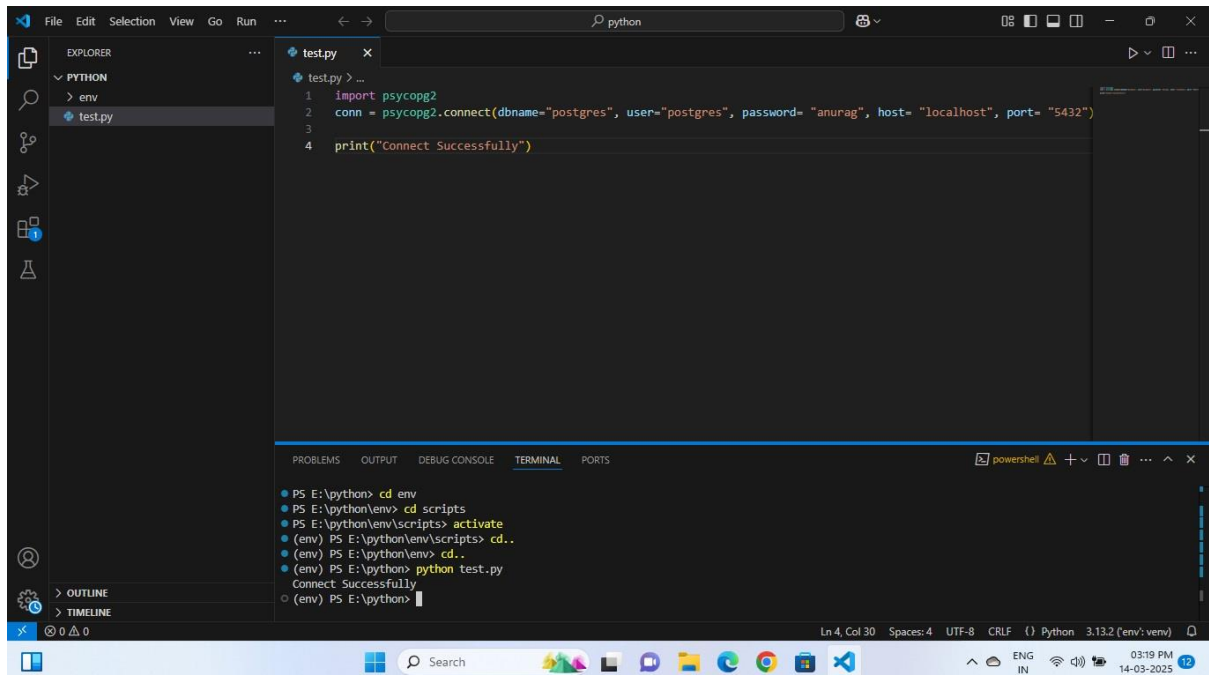


## 1. Creating Virtual environment



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project structure with a 'PYTHON' folder containing 'env' and 'test.py'. The main editor shows the content of 'test.py':

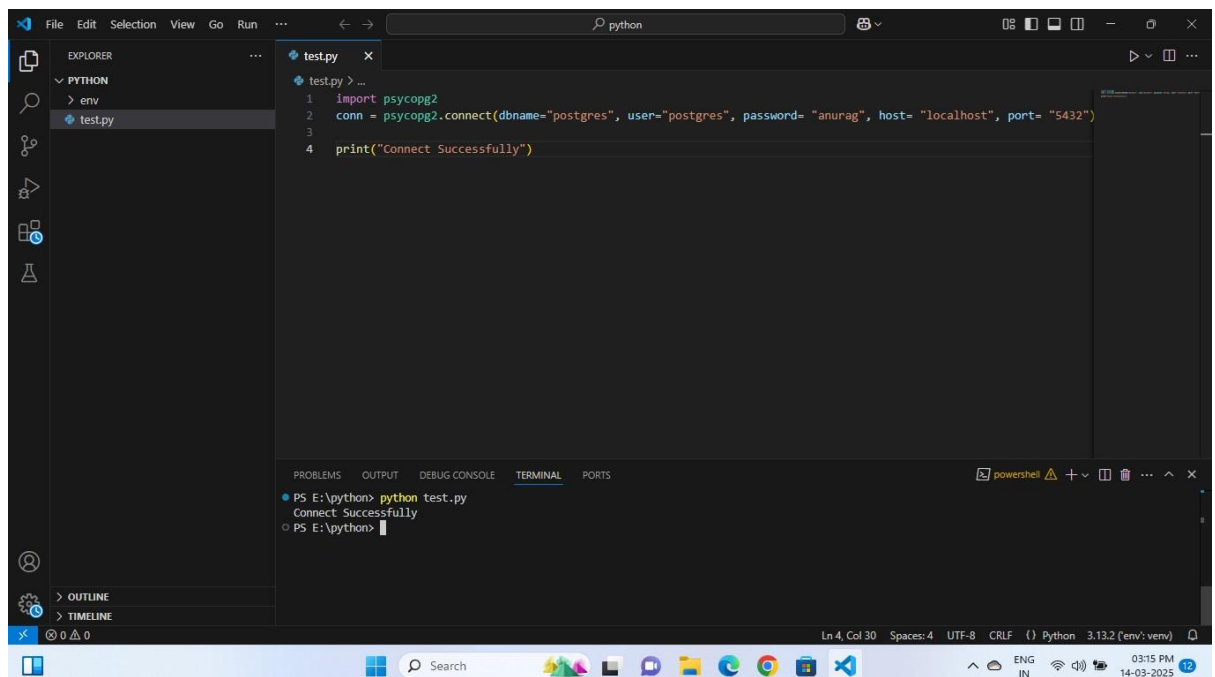
```
1 import psycopg2
2 conn = psycopg2.connect(dbname="postgres", user="postgres", password="anurag", host="localhost", port="5432")
3
4 print("Connect Successfully")
```

The integrated terminal at the bottom shows the following commands and output:

```
PS E:\python> cd env
PS E:\python\env> cd scripts
PS E:\python\env\scripts> activate
(env) PS E:\python\env\scripts> cd..
(env) PS E:\python\env> cd..
(env) PS E:\python> python test.py
Connect Successfully
(env) PS E:\python>
```

The status bar at the bottom indicates the file is at Line 4, Column 30, using UTF-8 encoding and CRLF line endings, with Python 3.13.2 (env\venv) selected.

## 2. Connect to database using psycopg2



This screenshot shows the same VS Code environment as the previous one, but the terminal now shows the execution of the test script:

```
PS E:\python> python test.py
Connect Successfully
PS E:\python>
```

The code in the editor remains the same. The status bar continues to show Python 3.13.2 (env\venv).

## 3. Create table using python

The screenshot shows the Visual Studio Code interface with a Python file named `test.py` open. The code in the file is as follows:

```
1 import psycopg2
2 conn = psycopg2.connect(dbname="postgres", user="postgres", password="anurag", host="localhost", port="5432")
3 cursor = conn.cursor()
4 cursor.execute('CREATE TABLE employees(ID int, name text, age int);')
5 print("Table Created")
6 conn.commit()
7 conn.close()
8
```

The terminal at the bottom shows the command `python test.py` being executed, resulting in the output `Table Created`.

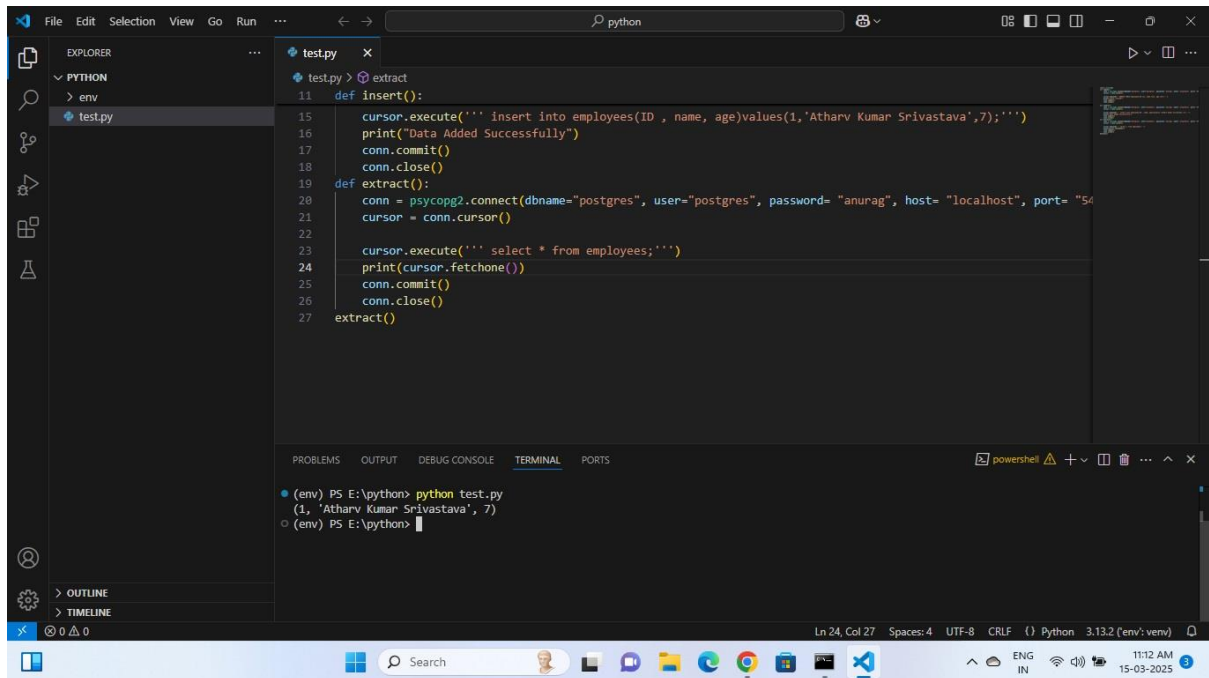
#### 4. Inserting data using python

The screenshot shows the Visual Studio Code interface with a Python file named `test.py` open. The code in the file is as follows:

```
2 def table():
3     conn = psycopg2.connect(dbname="postgres", user="postgres", password="anurag", host="localhost", port="5432")
4     cursor = conn.cursor()
5     cursor.execute('CREATE TABLE employees(ID int, name text, age int);')
6     print("Table Created")
7     conn.commit()
8     conn.close()
9
10 def insert():
11     conn = psycopg2.connect(dbname="postgres", user="postgres", password="anurag", host="localhost", port="5432")
12     cursor = conn.cursor()
13     cursor.execute('insert into employees(ID , name, age)values(1,\'Atharv Kumar Srivastava\',7);')
14     print("Data Added Successfully")
15     conn.commit()
16     conn.close()
17
18 insert()
19
```

The terminal at the bottom shows the command `python test.py` being executed, resulting in the output `Data Added Successfully`.

## 5. Extracting data from database using python



The screenshot shows a Visual Studio Code editor with a Python file named `test.py`. The file contains two functions: `insert()` and `extract()`. The `insert()` function connects to a PostgreSQL database, inserts a new employee record, and prints a success message. The `extract()` function connects to the same database, retrieves all records from the `employees` table, and prints them. The terminal at the bottom shows the command `python test.py` being executed, resulting in the output: `(1, 'Atharv Kumar Srivastava', 7)`.

```
11 def insert():
15     cursor.execute('insert into employees(ID , name, age)values(1,'Atharv Kumar Srivastava',7);')
16     print("Data Added Successfully")
17     conn.commit()
18     conn.close()
19 def extract():
20     conn = psycopg2.connect(dbname="postgres", user="postgres", password= "anurag", host= "localhost", port= "5432")
21     cursor = conn.cursor()
22
23     cursor.execute('select * from employees;')
24     print(cursor.fetchone())
25     conn.commit()
26     conn.close()
27 extract()
```

```
(env) PS E:\python> python test.py
(1, 'Atharv Kumar Srivastava', 7)
(env) PS E:\python>
```

## 6. Inserting data through user using python

