

Module 12. Building Database Apps with PostgreSQL & Python

1. Creating DataBase

```
SQL Shell (psql)
postgres=# create database demodb;
ERROR: database "demodb" already exists
postgres=# create database hellodb;
CREATE DATABASE
postgres=# /l
postgres=# \l
```

Name	Owner	Encoding	Locale Provider	Collate	Ctype	Locale	ICU Rules	Access privileges
demodb	postgres	UTF8	libc	English_India.1252	English_India.1252			
hellodb	postgres	UTF8	libc	English_India.1252	English_India.1252			
postgres	postgres	UTF8	libc	English_India.1252	English_India.1252			
students	postgres	UTF8	libc	English_India.1252	English_India.1252			
template0	postgres	UTF8	libc	English_India.1252	English_India.1252			=c/postgres +
template1	postgres	UTF8	libc	English_India.1252	English_India.1252			postgres=CTc/postgres +

```
(6 rows)

postgres=# \c demodb
You are now connected to database "demodb" as user "postgres".
demodb=#
```

2. Deleting DataBase

```
SQL Shell (psql)
demodb=# create database test;
CREATE DATABASE
demodb=# \l

      Name | Owner | Encoding | Locale Provider | Collate | Ctype | Locale | ICU Rules | Access privileges
-----|-----|-----|-----|-----|-----|-----|-----|-----
demodb    | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
hellodb   | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
postgres  | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
students  | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
template0 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | =c/postgres +
template1 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres +
test      | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres
(7 rows)

demodb=# drop dataase test;
ERROR:  syntax error at or near "dataase"
LINE 1: drop dataase test;
        ^

demodb=# drop database test;
DROP DATABASE
demodb=# \l

      Name | Owner | Encoding | Locale Provider | Collate | Ctype | Locale | ICU Rules | Access privileges
-----|-----|-----|-----|-----|-----|-----|-----|-----
demodb    | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
hellodb   | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
postgres  | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
students  | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
template0 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | =c/postgres +
template1 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres +
test      | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres
(6 rows)

demodb=# |
```

3. Creatlng table and adding data

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5433]:
Username [postgres]:
Password for user postgres:

psql (18.0)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \l

      Name | Owner | Encoding | Locale Provider | Collate | Ctype | Locale | ICU Rules | Access privileges
-----|-----|-----|-----|-----|-----|-----|-----|-----
postgres  | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | 
template0 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | =c/postgres +
template1 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres +
test      | postgres | UTF8 | libc | English_India.1252 | English_India.1252 |  |  | postgres=CTc/postgres
(3 rows)

postgres=# create database students;
CREATE DATABASE
postgres=# \c students
You are now connected to database "students" as user "postgres".
students=# create table students(name text, number int, age int);
CREATE TABLE
students=# \d

      List of relations
Schema | Name | Type | Owner
-----|-----|-----|-----
public | students | table | postgres
(1 row)

students=# insert into students(name,number,age) values('Sam',12,20);
INSERT 0 1
students=# insert into students(name,number,age) values('Jake',22,20);
INSERT 0 1
students=# |
```

4. Retrieving data from database and deleting contents in the table

```
SQL Shell (psql)
students=# select * from students;
 name | number | age
-----+-----+----
 Sam   |      12 |   20
 Jake  |      22 |   20
(2 rows)

students=# select name from students;
 name
-----
 Sam
 Jake
(2 rows)

students=# select * from students where number=12;
 name | number | age
-----+-----+----
 Sam   |      12 |   20
(1 row)

students=# select * from students where age=12;
 name | number | age
-----+-----+----
(0 rows)

students=# select * from students where age=20;
 name | number | age
-----+-----+----
 Sam   |      12 |   20
 Jake  |      22 |   20
(2 rows)

students=# select number from students where age=20;
 number
-----
      12
      22
(2 rows)

students=# select number from students where name='Sam';
 number
-----
      12
(1 row)

students=# truncate table
students=# truncate table students;
ERROR:  syntax error at or near "table"
LINE 2: truncate table students;
                    ^
students=# truncate table students;
TRUNCATE TABLE
students=# \d
          list of relations
Schema | Name      | Type  | Owner
-----+-----+-----+-----
public | students | table | postgres
(1 row)

students=# select * from students;
 name | number | age
-----+-----+----
(0 rows)

students=#
```

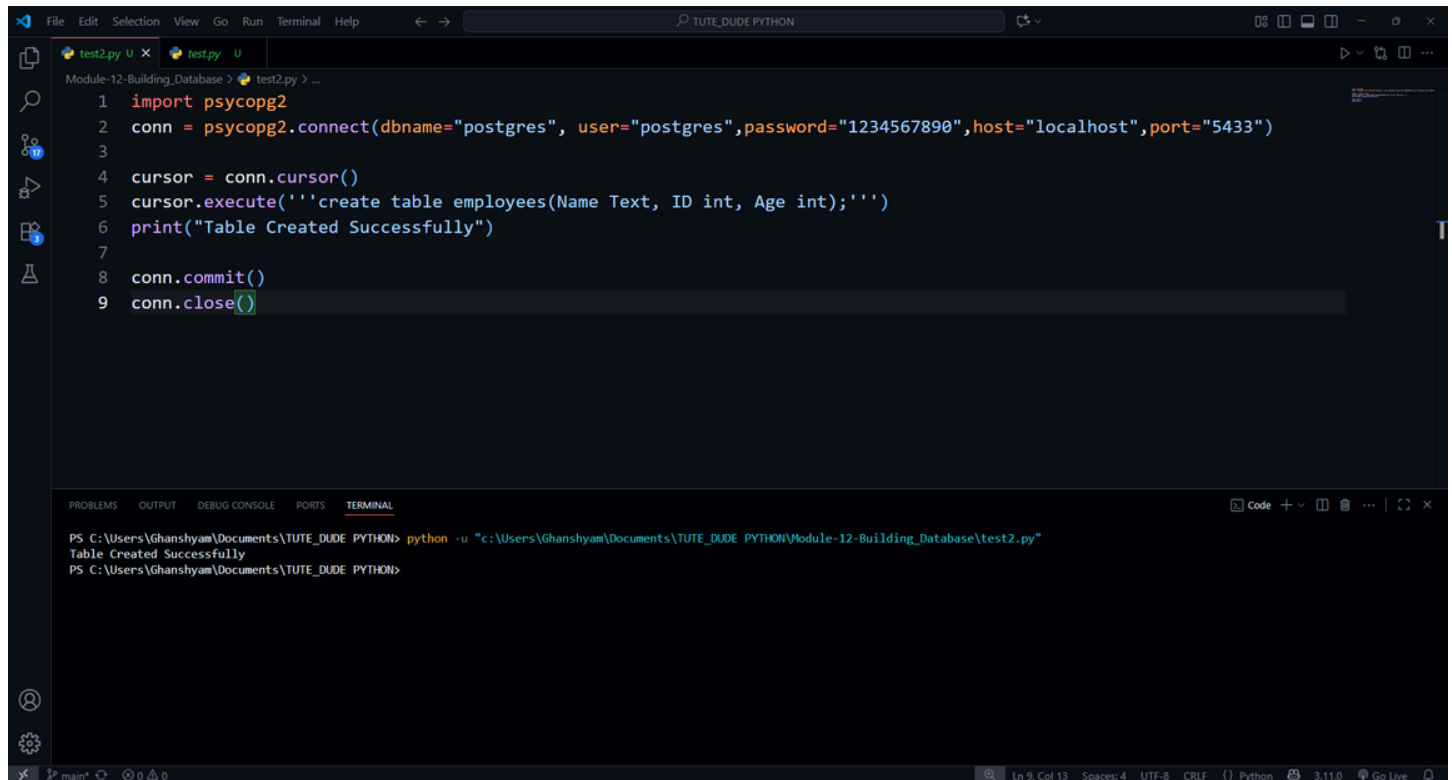
4. Connecting to Database

```
File Edit Selection View Go Run Terminal Help
TUTE_DUDE PYTHON

test2.py U
Module-12-Building_Database > test2.py > ...
1 import psycopg2
2 connect = psycopg2.connect(dbname="postgres", user="postgres",password="1234567890",host="localhost",port="5433")
3
4 print("Connected Successfully")

PROBLEMS OUTPUT DEBUG CONSOLE PORTS TERMINAL
PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON> python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"
Connected Successfully
PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON>
```

5. Creating table using python

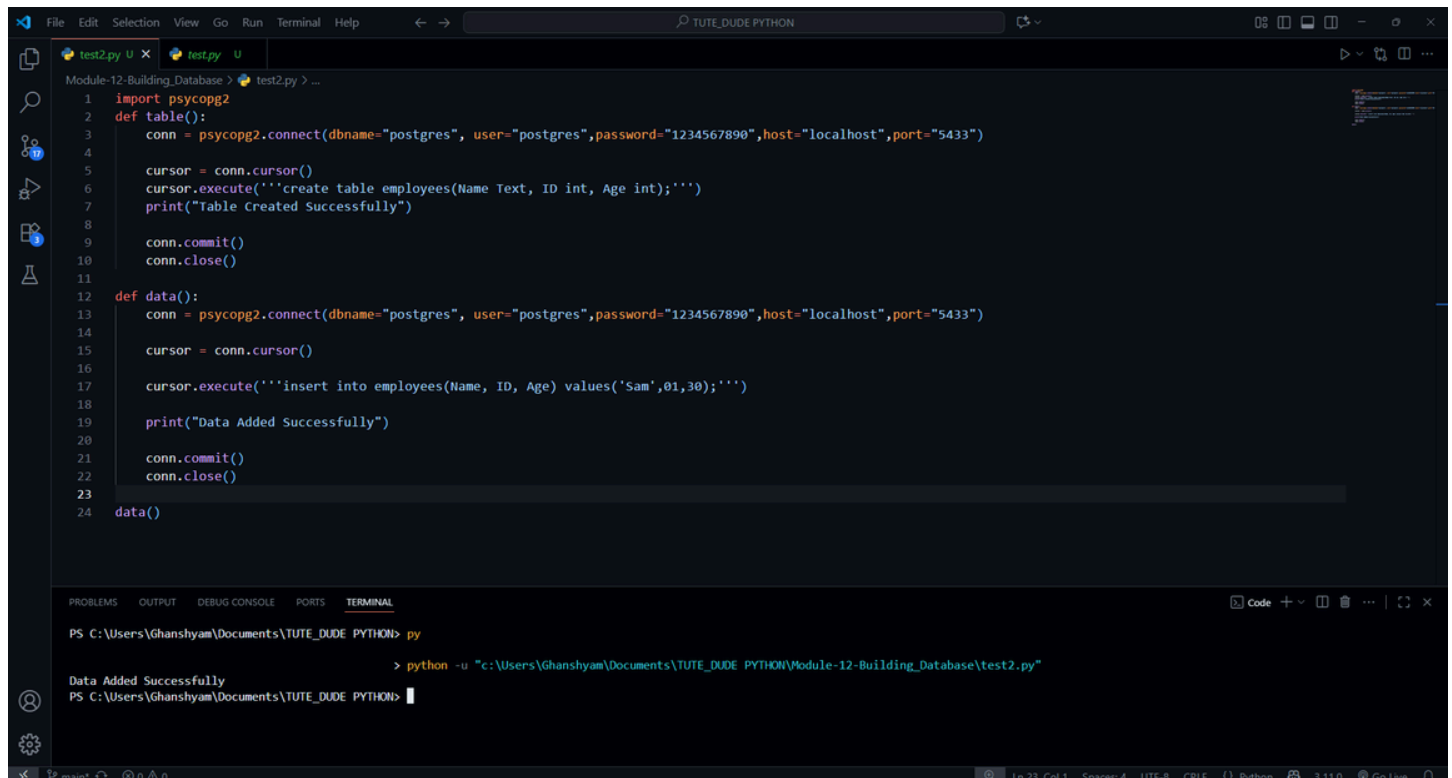


The screenshot shows a VS Code editor with a file named `test2.py` open. The code in the editor is as follows:

```
1 import psycopg2
2 conn = psycopg2.connect(dbname="postgres", user="postgres",password="1234567890",host="localhost",port="5433")
3
4 cursor = conn.cursor()
5 cursor.execute('''create table employees(Name Text, ID int, Age int);''')
6 print("Table Created Successfully")
7
8 conn.commit()
9 conn.close()
```

Below the editor, the terminal window shows the command `python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"` being executed, resulting in the output `Table Created Successfully`.

5. Inserting data using python

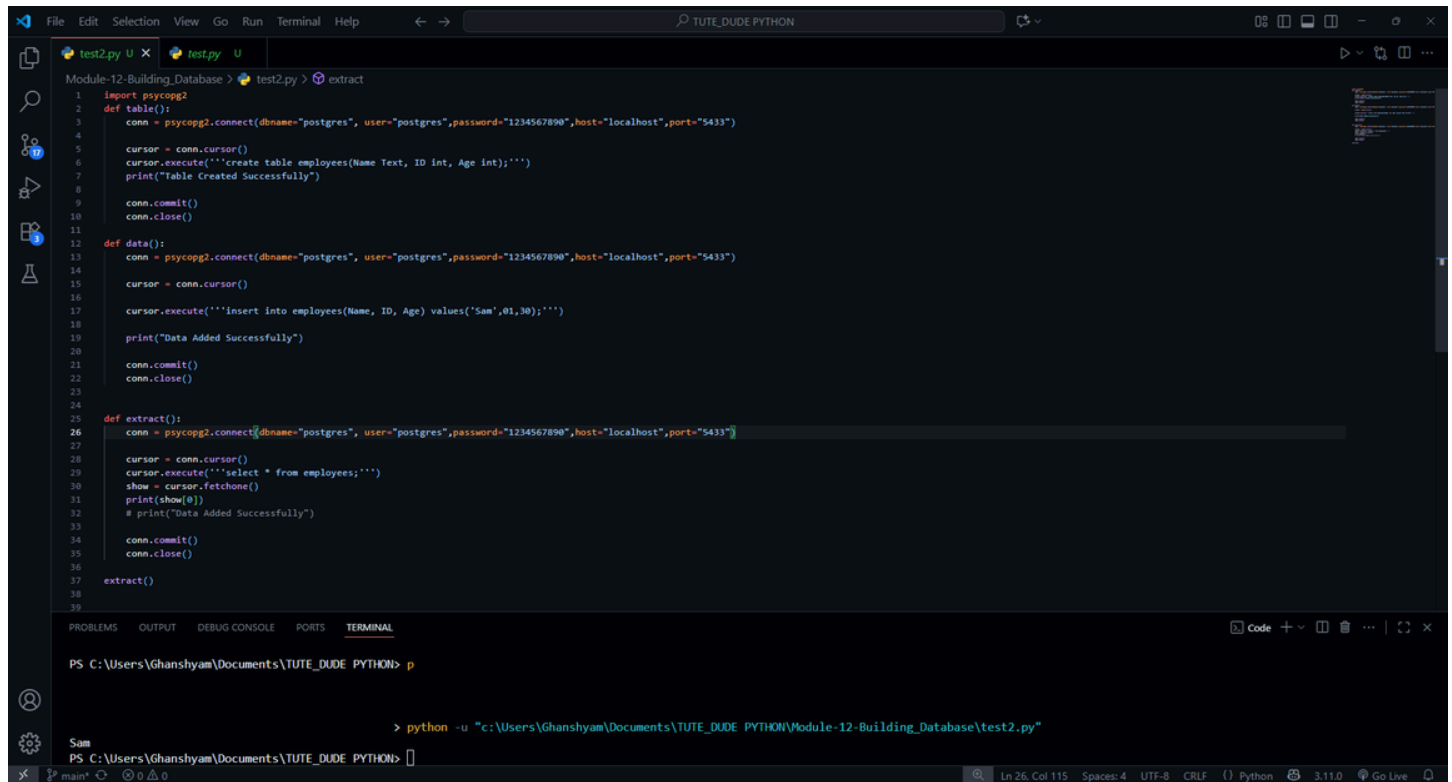


The screenshot shows a VS Code editor with a file named `test2.py` open. The code in the editor is as follows:

```
1 import psycopg2
2 def table():
3     conn = psycopg2.connect(dbname="postgres", user="postgres",password="1234567890",host="localhost",port="5433")
4
5     cursor = conn.cursor()
6     cursor.execute('''create table employees(Name Text, ID int, Age int);''')
7     print("Table Created Successfully")
8
9     conn.commit()
10    conn.close()
11
12 def data():
13     conn = psycopg2.connect(dbname="postgres", user="postgres",password="1234567890",host="localhost",port="5433")
14
15     cursor = conn.cursor()
16     cursor.execute('''insert into employees(Name, ID, Age) values('Sam',01,30);''')
17
18     print("Data Added Successfully")
19
20     conn.commit()
21     conn.close()
22
23 data()
24
```

Below the editor, the terminal window shows the command `python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"` being executed, resulting in the output `Data Added Successfully`.

6. Extracting data from the database



The screenshot shows a VS Code editor with a Python script named `test2.py` in the `Module-12-Building_Database` project. The script defines three functions: `table()`, `data()`, and `extract()`. The `extract()` function is highlighted in blue. The terminal at the bottom shows the command `python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"` being executed, resulting in the output `Sam`.

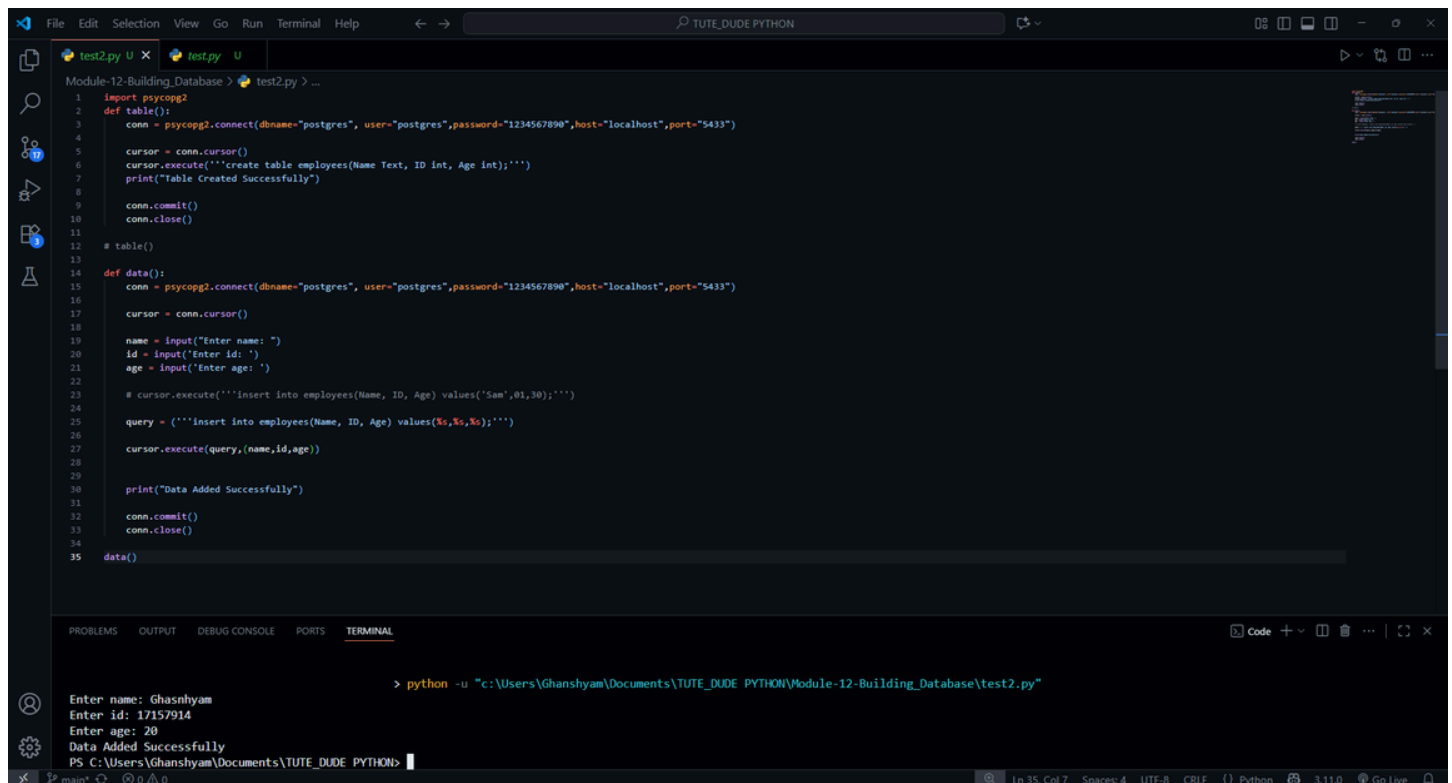
```
1 import psycopg2
2 def table():
3     conn = psycopg2.connect(dbname="postgres", user="postgres", password="1234567890", host="localhost", port="5433")
4
5     cursor = conn.cursor()
6     cursor.execute('create table employees(Name Text, ID int, Age int);')
7     print("Table Created Successfully")
8
9     conn.commit()
10    conn.close()
11
12 def data():
13     conn = psycopg2.connect(dbname="postgres", user="postgres", password="1234567890", host="localhost", port="5433")
14
15     cursor = conn.cursor()
16     cursor.execute('insert into employees(Name, ID, Age) values("Sam",01,30);')
17     print("Data Added Successfully")
18
19     conn.commit()
20     conn.close()
21
22 def extract():
23     conn = psycopg2.connect(dbname="postgres", user="postgres", password="1234567890", host="localhost", port="5433")
24
25     cursor = conn.cursor()
26     cursor.execute('select * from employees;')
27     show = cursor.fetchone()
28     print(show[0])
29     # print("Data Added Successfully")
30
31     conn.commit()
32     conn.close()
33
34 extract()
35
```

PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON> python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"

Sam

PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON>

6. Adding the input from the user



The screenshot shows a VS Code editor with a Python script named `test2.py` in the `Module-12-Building_Database` project. The script defines three functions: `table()`, `data()`, and `extract()`. The `data()` function is highlighted in blue. The terminal at the bottom shows the command `python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"` being executed, resulting in the output `Enter name: Ghasnhyam`, `Enter id: 17157914`, `Enter age: 20`, and `Data Added Successfully`.

```
1 import psycopg2
2 def table():
3     conn = psycopg2.connect(dbname="postgres", user="postgres", password="1234567890", host="localhost", port="5433")
4
5     cursor = conn.cursor()
6     cursor.execute('create table employees(Name Text, ID int, Age int);')
7     print("Table Created Successfully")
8
9     conn.commit()
10    conn.close()
11
12 # table()
13
14 def data():
15     conn = psycopg2.connect(dbname="postgres", user="postgres", password="1234567890", host="localhost", port="5433")
16
17     cursor = conn.cursor()
18
19     name = input("Enter name: ")
20     id = input("Enter id: ")
21     age = input("Enter age: ")
22
23     # cursor.execute('insert into employees(Name, ID, Age) values("Sam",01,30);')
24
25     query = ('insert into employees(Name, ID, Age) values(%s,%s,%s);')
26     cursor.execute(query,(name,id,age))
27
28     print("Data Added Successfully")
29
30     conn.commit()
31     conn.close()
32
33 data()
34
```

PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON> python -u "c:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON\Module-12-Building_Database\test2.py"

Enter name: Ghasnhyam

Enter id: 17157914

Enter age: 20

Data Added Successfully

PS C:\Users\Ghanshyam\Documents\TUTE_DUDE PYTHON>