

1. Set the Password of postgresql.

```
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:

psql (18.0)
WARNING: Console code page (850) 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=#
```

2. Available already databases \l.

```
postgres=# \l
      Name | Owner | Encoding | Locale Provider | List of databases Collate | Ctype | Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 postgres | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
 template0 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | postgres=CTc/postgres +
 template1 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
           |         |     |     |                     |                     | | | | postgres=CTc/postgres
(3 rows)

postgres=# |
```

3. Creating the demo database create database demodb;

```
postgres=# create database demodb;
CREATE DATABASE
postgres=# \l
      Name | Owner | Encoding | Locale Provider | List of databases Collate | Ctype | Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 demodb    | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
 postgres | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | postgres=CTc/postgres +
 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
(4 rows)

postgres=# |
```

4. Switch database \c demodb

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

5. Creation db test and deleted create database test; For deletion drop database test;

```
demodb=# create database test;
CREATE DATABASE
demodb=# \l
      Name | Owner | Encoding | Locale Provider | List of databases Collate | Ctype | Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 demodb    | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
 postgres | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | postgres=CTc/postgres +
 template0 | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
 test      | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | postgres=CTc/postgres
(5 rows)

demodb=# drop database test;
DROP DATABASE
demodb=# \l
      Name | Owner | Encoding | Locale Provider | List of databases Collate | Ctype | Locale | ICU Rules | Access privileges
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 demodb    | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | =c/postgres +
 postgres | postgres | UTF8 | libc | English_India.1252 | English_India.1252 | | | | postgres=CTc/postgres +
 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
(4 rows)

demodb=# |
```

6. Student db created, switch into student CREATE TABLE students(name text, number int, age int)

```
demodb=# create database student;
CREATE DATABASE
demodb=# \c student
You are now connected to database "student" as user "postgres".
student=# CREATE TABLE students(name text,number int,age int);
CREATE TABLE
student=# \d
          List of relations
Schema |   Name   | Type  | Owner
-----+-----+-----+-----
public | students | table | postgres
(1 row)

student=# |
```

7. Insert data into table INSERT INTO students(name,number,age) VALUES ('Sam',12,20);

```
student=# INSERT INTO students(name,number,age) VALUES('Sam',12,20);
INSERT 0 1
student=# INSERT INTO students(name,number,age) VALUES('Shino',13,21);
INSERT 0 1
student=# |
```

8. Retriving data from database.

```
student=# SELECT * from students;
 name | number | age
-----+-----+-----
 Sam  |      12 |   20
 Shino |      13 |   21
(2 rows)

student=# SELECT name from students;
 name
-----
 Sam
 Shino
(2 rows)

student=# SELECT number from students;
 number
-----
      12
      13
(2 rows)

student=# SELECT age from students;
 age
-----
   20
   21
(2 rows)

student=# |
```

9. Filter the data using WHERE in table.

```
student=# SELECT * from students WHERE age=20;
 name | number | age
-----+-----+-----
 Sam  |     12 |    20
(1 row)
```

```
student=# SELECT * from students WHERE number=13;
 name | number | age
-----+-----+-----
 Shino |     13 |    21
(1 row)
```

```
student=# SELECT number from students WHERE name='Shino';
 number
-----
     13
(1 row)
```

```
student=# SELECT number from students WHERE age=20;
 number
-----
     12
(1 row)
```

10. Removing all data from table.

```
student=# TRUNCATE TABLE students;
TRUNCATE TABLE
student=# \d
          List of relations
Schema |   Name   | Type  | Owner
-----+-----+-----+-----
 public | students | table | postgres
(1 row)
```

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

11. Installing virtual environment.

```
PS D:\Coding\Python\Assignment-07> pip install virtualenv
Collecting virtualenv
  Downloading virtualenv-20.34.0-py3-none-any.whl.metadata (4.6 kB)
Collecting distlib<=0.3.7 (from virtualenv)
  Downloading distlib-0.4.0-py2.py3-none-any.whl.metadata (5.2 kB)
Collecting filelock<=3.12.2 (from virtualenv)
  Downloading filelock-3.19.1-py3-none-any.whl.metadata (2.1 kB)
Collecting platformdirs<=3.9.1 (from virtualenv)
  Downloading platformdirs-4.4.0-py3-none-any.whl.metadata (12 kB)
Downloaded virtualenv-20.34.0-py3-none-any.whl (6.0 MB)
-----
 0.0 MB/s | 0.0 MB/s | 0.0 MB/s | eta 0:00:00
Downloaded distlib-0.4.0-py2.py3-none-any.whl (469 kB)
-----
 0.0 MB/s | 0.0 MB/s | 0.0 MB/s | eta 0:00:00
Downloaded filelock-3.19.1-py3-none-any.whl (15 kB)
Downloaded platformdirs-4.4.0-py3-none-any.whl (18 kB)
Installing collected packages: distlib, platformdirs, filelock, virtualenv
WARNING: The script virtualenv.exe is installed in 'C:\Users\Siddharth Kumar\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\Scripts' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed distlib-0.4.0 filelock-3.19.1 platformdirs-4.4.0 virtualenv-20.34.0

[notice] A new release of pip is available: 24.0 -> 26.2
[notice] To update, run: C:\Users\Siddharth Kumar\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\python.exe -m pip install --upgrade pip
PS D:\Coding\Python\Assignment-07>
```

12. Run python virtual environment with `.\env\scripts\activate` or `cd env\script\activate`.

```
PS D:\Coding\Python\Assignment-07> python --version
Python 3.11.9
PS D:\Coding\Python\Assignment-07> python -m venv env
PS D:\Coding\Python\Assignment-07> .\env\Scripts\activate
(env) PS D:\Coding\Python\Assignment-07> |
```

13. Goto the file location and run the python file.

```
(env) PS D:\Coding\Python\Assignment-07> python .\test.py
hello world
(env) PS D:\Coding\Python\Assignment-07>
```

14. For deactivate the environment.

```
(env) PS D:\Coding\Python\Assignment-07> deactivate
PS D:\Coding\Python\Assignment-07> |
```

15. Psycopg2 package connects Python and postgresql db.

```
PS D:\Coding\Python\Assignment-07> pip install psycopg2
Collecting psycopg2
  Downloading psycopg2-2.9.10-cp311-cp311-win_amd64.whl.metadata (5.0 kB)
  Downloading psycopg2-2.9.10-cp311-cp311-win_amd64.whl (1.2 MB)
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.2/1.2 MB 1.3 MB/s 0:00:00
Installing collected packages: psycopg2
Successfully installed psycopg2-2.9.10
PS D:\Coding\Python\Assignment-07> |
```

16. Making connection with python and postgresql.

```
Assignment-07 > test.py > ...
1 import psycopg2
2 conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
3 |
4 print('connected successfully with postgres')

(env) PS D:\Coding\Python\Assignment-07> python .\test.py
connected successfully with postgres
(env) PS D:\Coding\Python\Assignment-07> |
```

17. Table creation

```
Assignment-07 > test.py > ...
1 import psycopg2
2 conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
3 cursor = conn.cursor()
4 cursor.execute('create table employees(Name Text,ID int,Age int);')
5
6 print('print table successfully')
7 conn.commit()
8 conn.close()
9

(env) PS D:\Coding\Python\Assignment-07> python .\test.py
print table successfully
(env) PS D:\Coding\Python\Assignment-07> |
```

```
postgres=# \d
               List of relations
 Schema |   Name   | Type | Owner
-----+-----+-----+-----
 public | employees | table | postgres
(1 row)

postgres=# select * from employees;
 name | id | age
-----+---+----
(0 rows)

postgres=# |
```

18. Insert data in database using function.

```
1 import psycopg2
2
3 def table():
4     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
5     cursor = conn.cursor()
6     cursor.execute('''create table employees(Name Text,ID int,Age int);''')
7     print('print table successfully')
8     conn.commit()
9     conn.close()
10
11 def data():
12     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
13     cursor = conn.cursor()
14     cursor.execute('''insert into employees(Name,ID,Age) values('sam',01,30);''')
15     print('print table successfully')
16     conn.commit()
17     conn.close()
18
19 data()
20
```

(env) PS D:\Coding\Python\Assignment-07> python .\test.py
Data added successfully
(env) PS D:\Coding\Python\Assignment-07>

```
postgres=# select * from employees;
 name | id | age
-----+---+----
 sam  |  1 |  30
 sam  |  1 |  30
(2 rows)
```

19. Data extract from database. We can also use `show = cursor.fetchone()` and `print(show[])` with index number of the data.

```

1 import psycopg2
2
3 def table():
4     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
5     cursor = conn.cursor()
6     cursor.execute('create table employees(Name Text,ID int,Age int);')
7     print('print table successfully')
8     conn.commit()
9     conn.close()
10
11 def data():
12     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
13     cursor = conn.cursor()
14     cursor.execute('insert into employees(Name,ID,Age) values('sam',01,30);')
15     print('Data added successfully')
16     conn.commit()
17     conn.close()
18
19 def extract():
20     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
21     cursor = conn.cursor()
22     cursor.execute('select * from employees;')
23     print(cursor.fetchone())
24     print("Data extract successfully")
25     conn.commit()
26     conn.close()
27 extract()
28
29 |

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter

[Running] python -u "d:\Coding\Python\Assignment-07\test.py"

('sam', 1, 30)

Data extract successfully

[Done] exited with code=0 in 0.513 seconds

20. Adding input from the user.

```

Assignment-07 > test.py > ...
1 import psycopg2
2
3 def table():
4     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
5     cursor = conn.cursor()
6     cursor.execute('create table employees(Name Text,ID int,Age int);')
7     print('print table successfully')
8     conn.commit()
9     conn.close()
10
11 def data():
12     conn = psycopg2.connect(dbname="postgres",user="postgres",password="Admin@1234",host="localhost",port="5432")
13     cursor = conn.cursor()
14     name = input("Enter your name: ")
15     id = input("Enter your id: ")
16     age = input("Enter your age: ")
17     query = 'insert into employees(Name,ID,Age) values(%s,%s,%s);'
18     cursor.execute(query,(name,id,age))
19     print('Data added successfully')
20     conn.commit()
21     conn.close()
22 data()

```

(env) PS D:\Coding\Python\Assignment-07> python .\test.py

Enter your name: Superman

Enter your id: 01

Enter your age: 30

Data added successfully

(env) PS D:\Coding\Python\Assignment-07> |

```

postgres=# select * from employees;
 name | id | age
-----+---+----
 sam   |  1 |  30
 sam   |  1 |  30
 Superman |  1 |  30
(3 rows)

```

***** THE END *****