

## SGD LAB EXP - 2

**Name :** Aditi Chhajer; **Reg. No. :** 221081009  
**Branch :** IT ; **Course Instructor :** Prof. Vedashree Awati

### Aim:

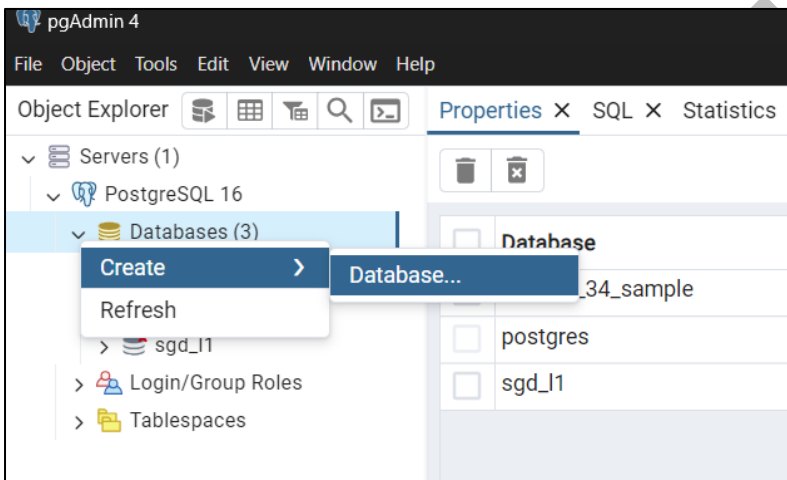
- A. Create a new Schema
- B. Load data from a shapefile
- C. Load data from CSV
- D. Load data using COPY command

### Implementation :

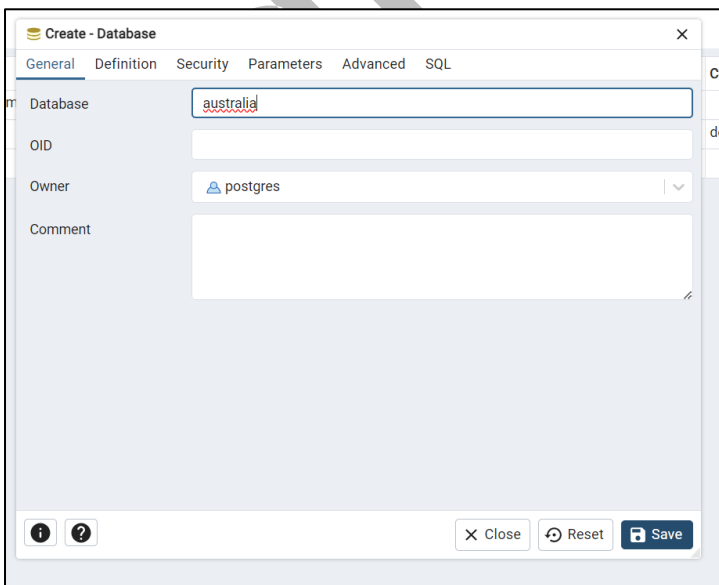
#### a. Create a new Schema

- Launch pgAdmin and create a new database

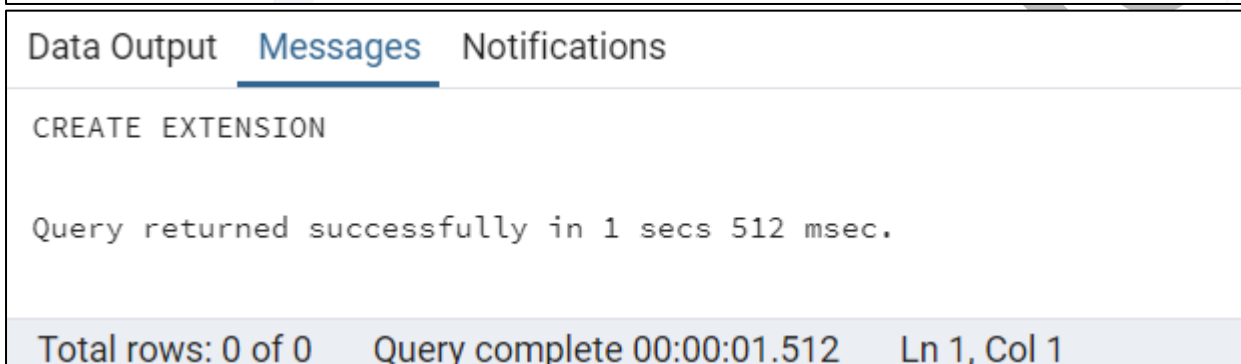
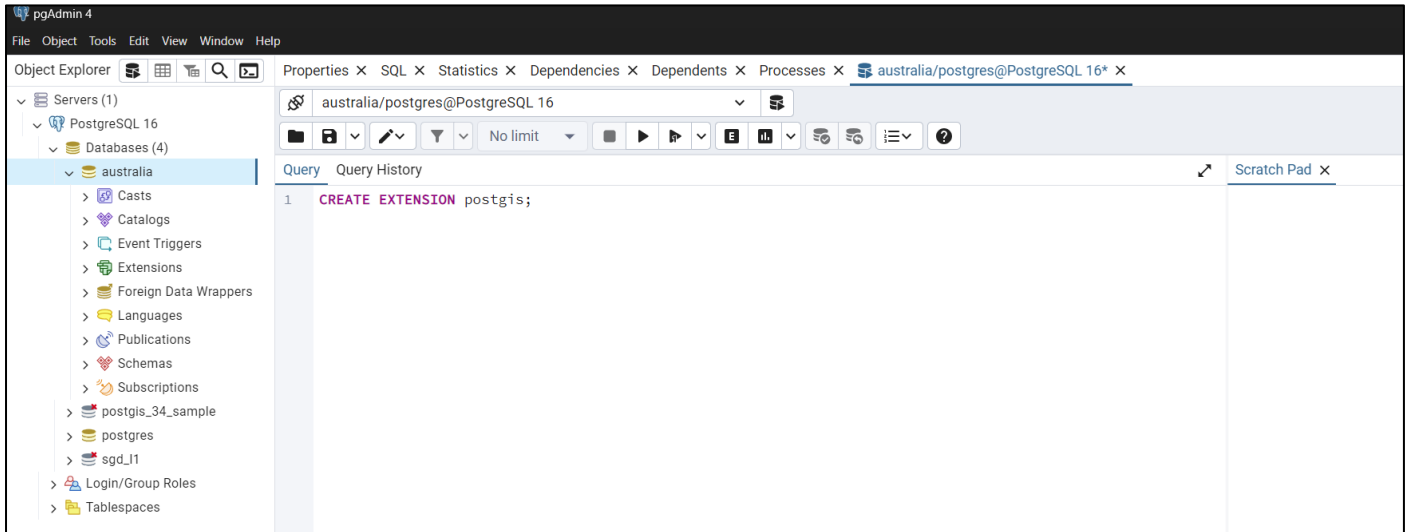
1. server->databases->create database



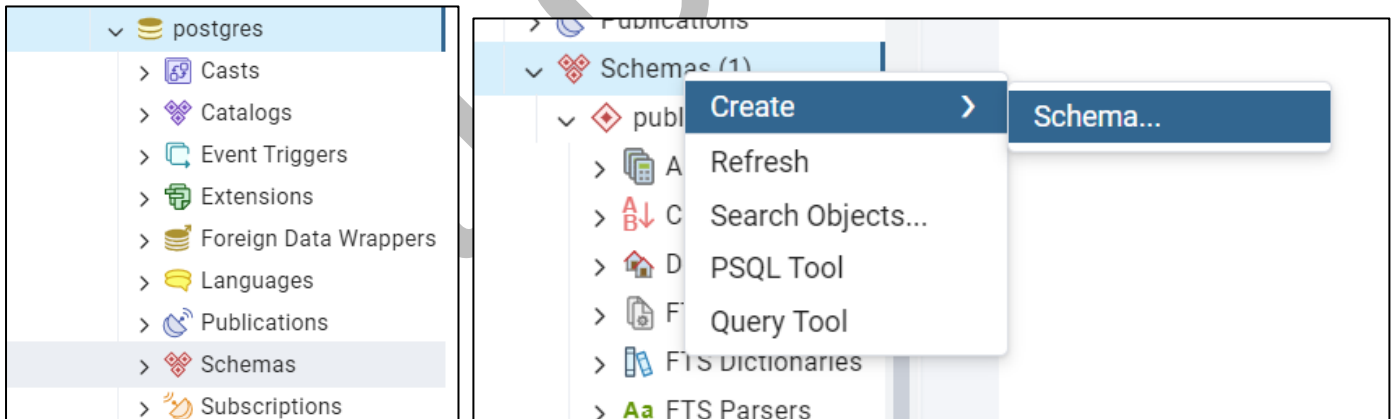
2. Enter database name.



- Under the new database create an extension using the following query  
**CREATE EXTENSION postgis;**



- Create a new schema under owner postgres.



**Create - Schema**

General Security Default privileges SQL

Name australia\_states

Owner postgres

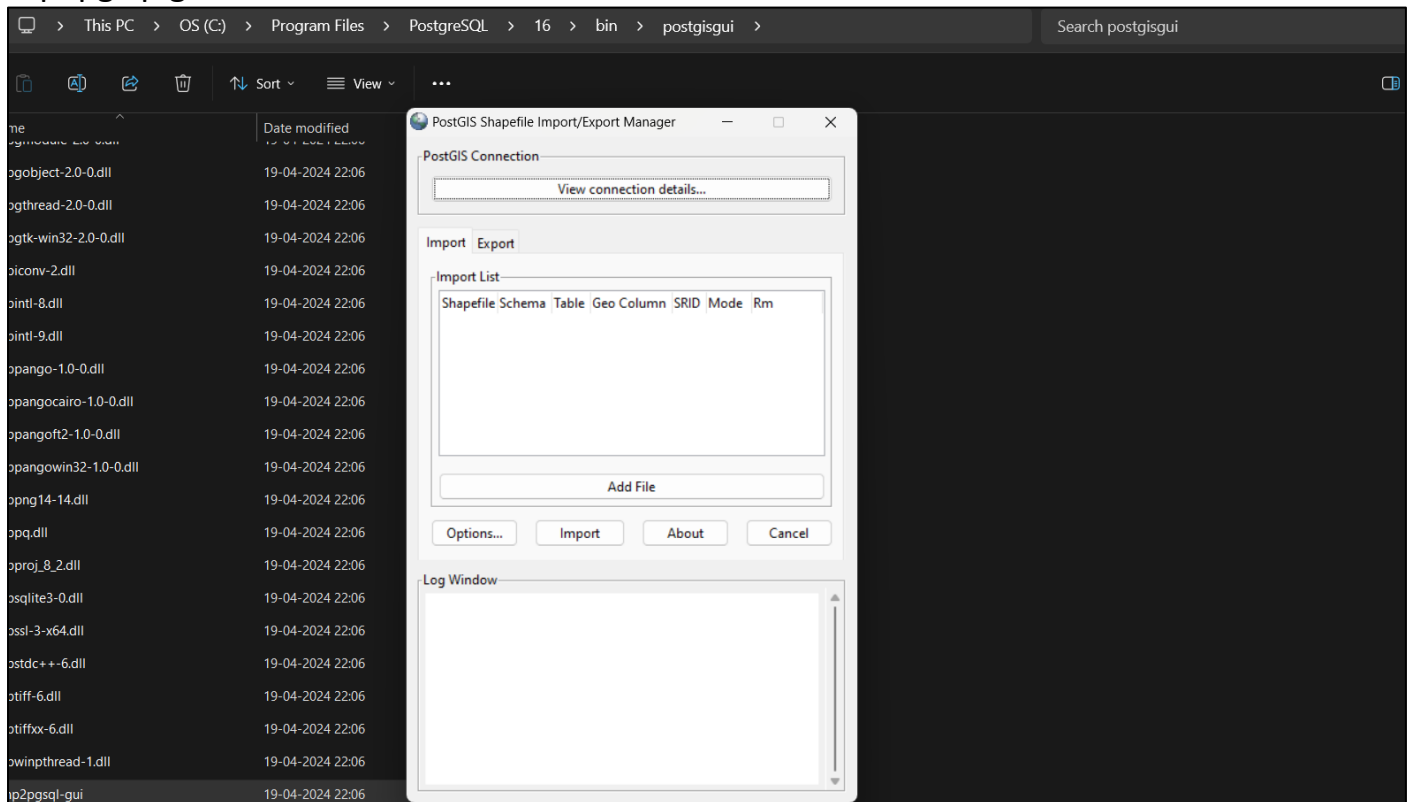
Comment

Close Reset Save

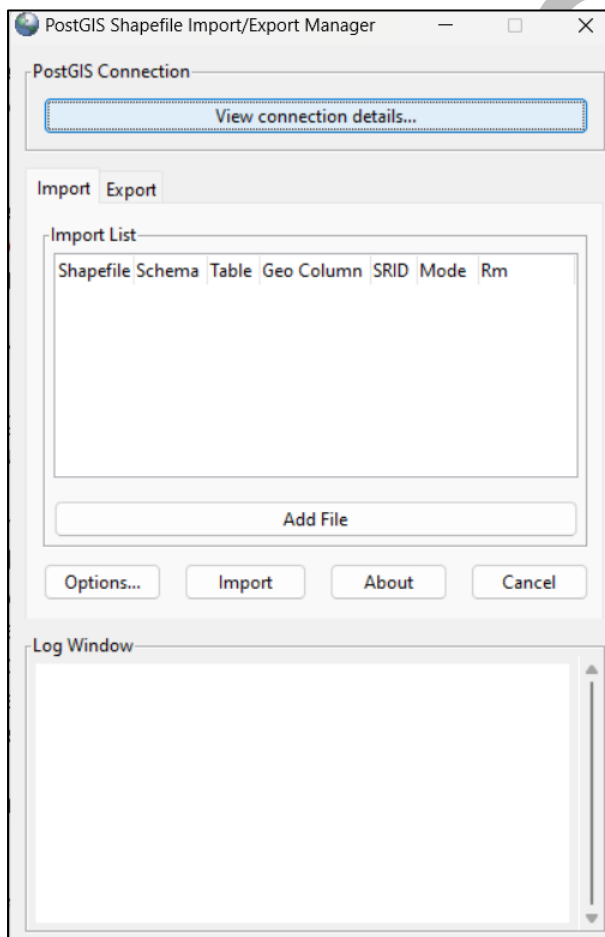
***we create "australia\_states" as our schema here***

## **B. Load data from a shapefile**

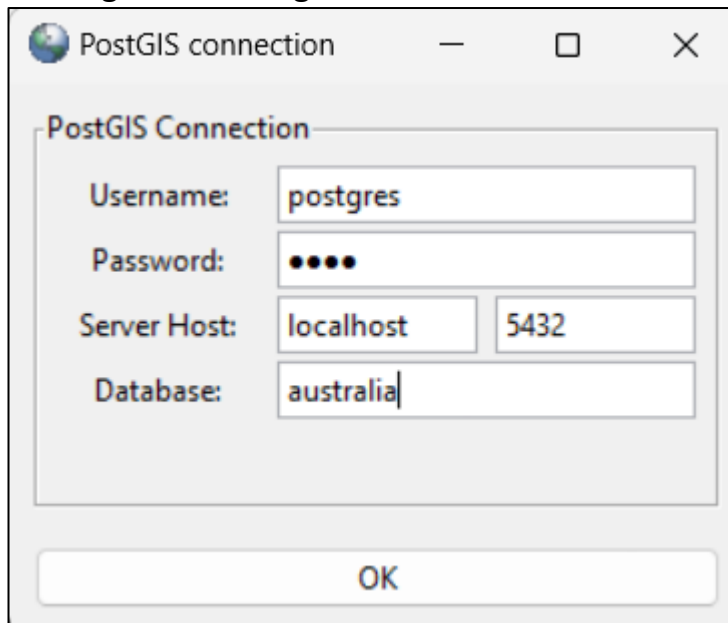
- Navigate to c:\Program Files\PostgreSQL\15\bin\postgis gui and run executable file shp2pgsql-gui.exe



- Click on view connection details



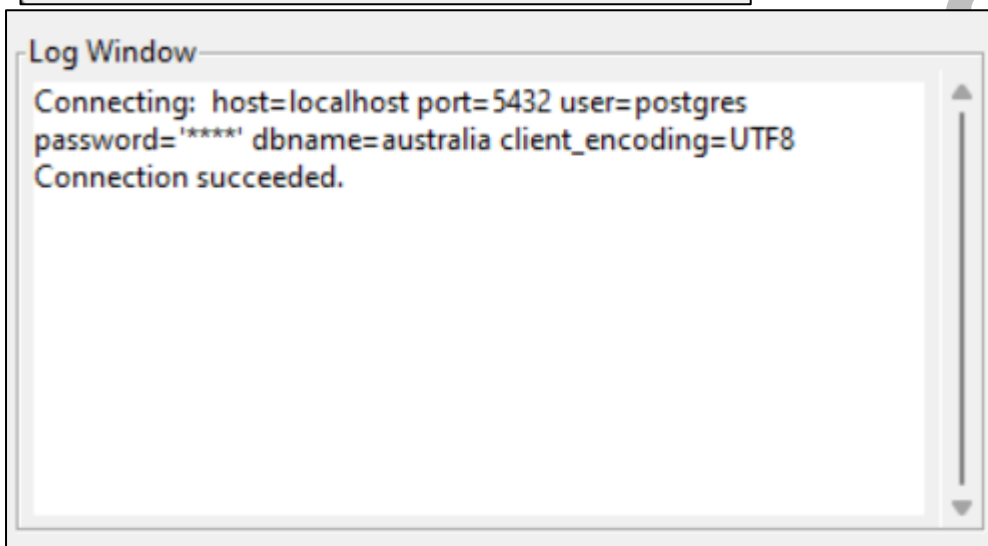
- Enter username as postgres, password, server host as localhost and port as 5432 and database name is the same as that was created above in step A.1. After clicking on OK, it should give a message connection succeeded.



A screenshot of a 'PostGIS connection' dialog box. The title bar shows a globe icon and the text 'PostGIS connection'. The dialog has a 'PostGIS Connection' section with four fields: 'Username' with 'postgres', 'Password' with four dots, 'Server Host' with 'localhost' and a port field with '5432', and 'Database' with 'australia'. An 'OK' button is at the bottom.

Username:	postgres	
Password:	••••	
Server Host:	localhost	5432
Database:	australia	

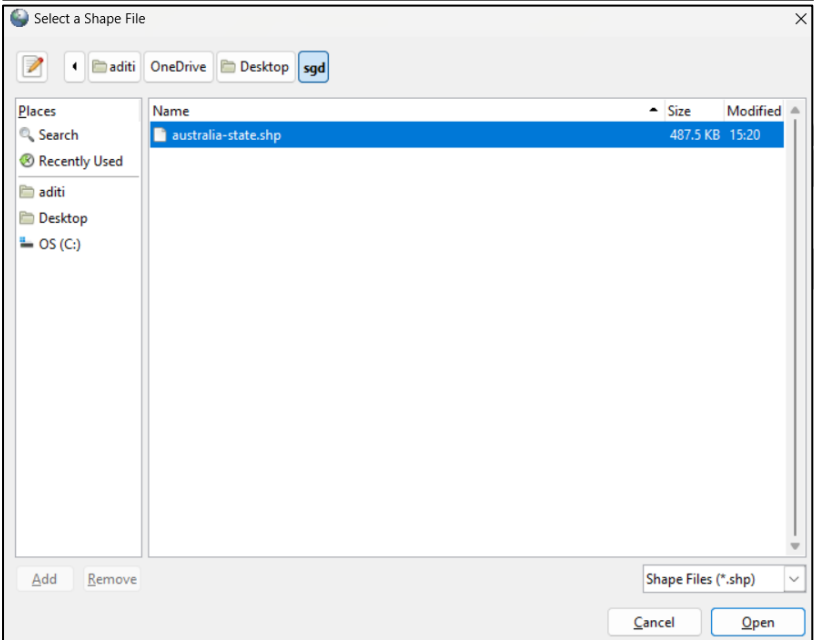
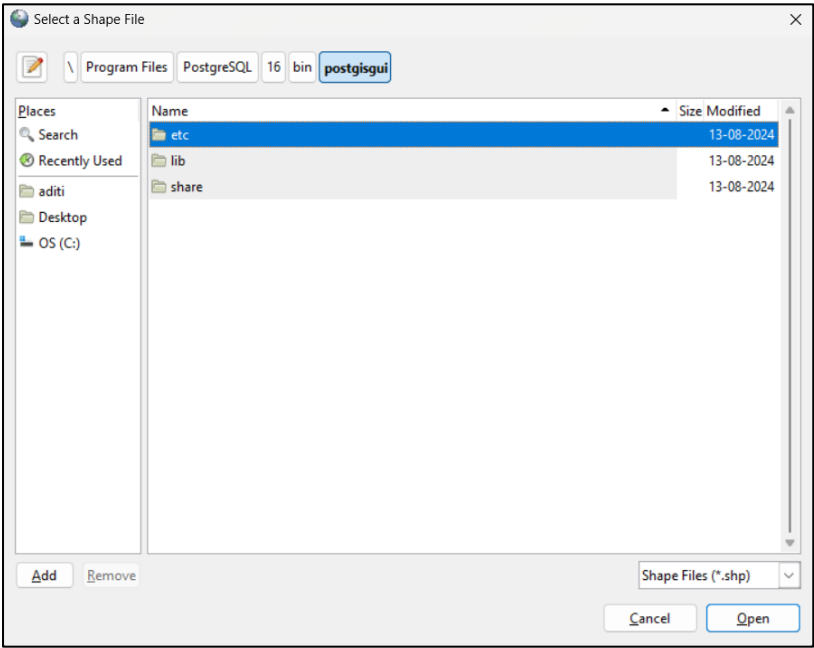
OK



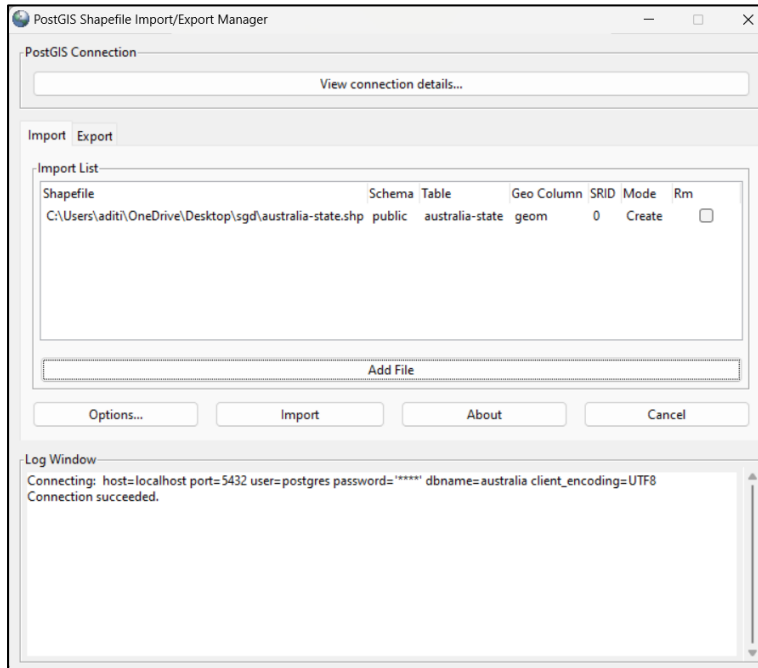
A screenshot of a 'Log Window' showing connection details. The text reads: 'Connecting: host=localhost port=5432 user=postgres password='\*\*\*\*\*' dbname=australia client\_encoding=UTF8 Connection succeeded.' The window has a scrollbar on the right.

```
Log Window
Connecting: host=localhost port=5432 user=postgres
password='*****' dbname=australia client_encoding=UTF8
Connection succeeded.
```

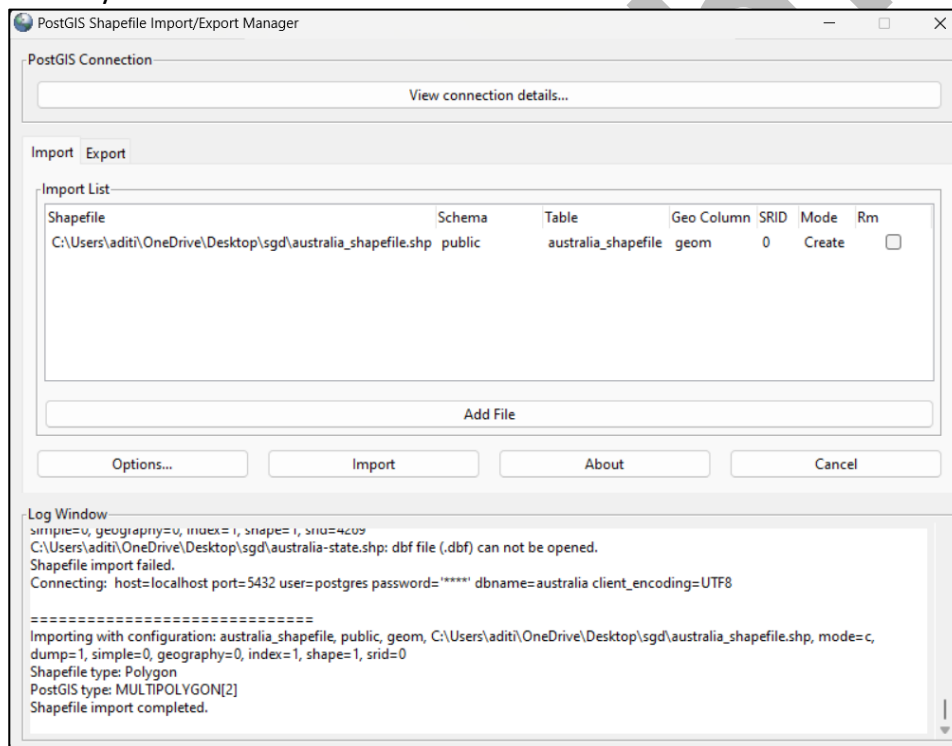
- Select the add files option and select the shapefile.

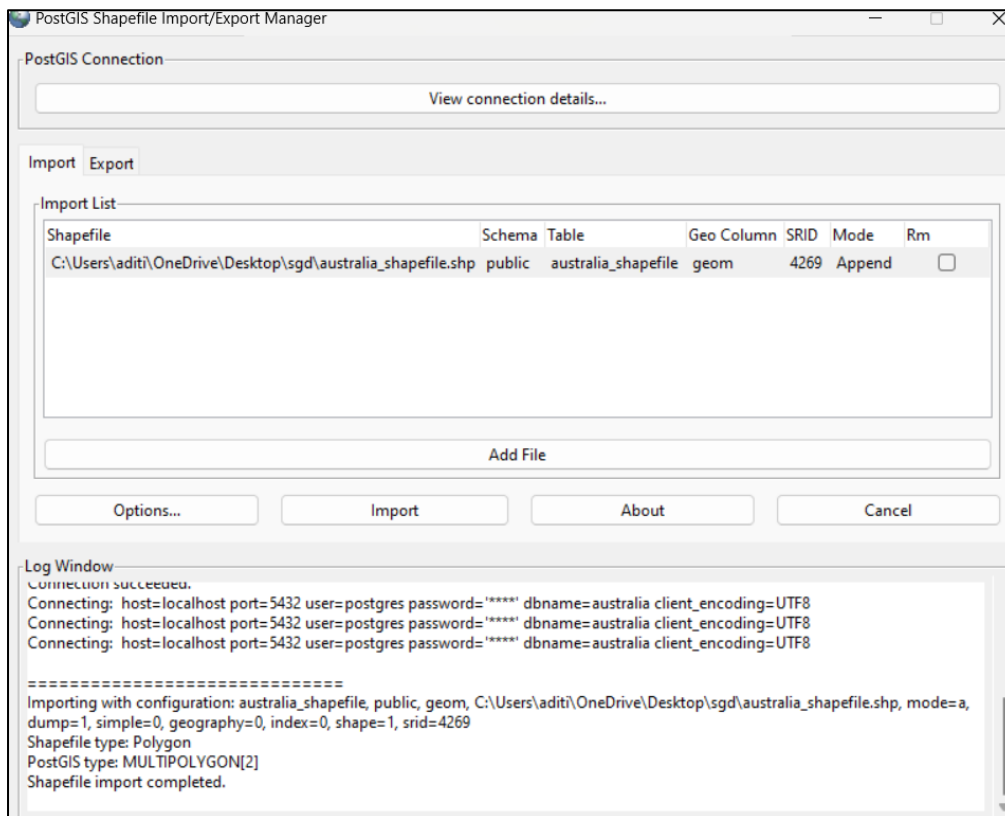


- In the import list set the schema parameter to public, table parameter to australia-state, geo column to geom and SRID to 4269. Now import the shapefile

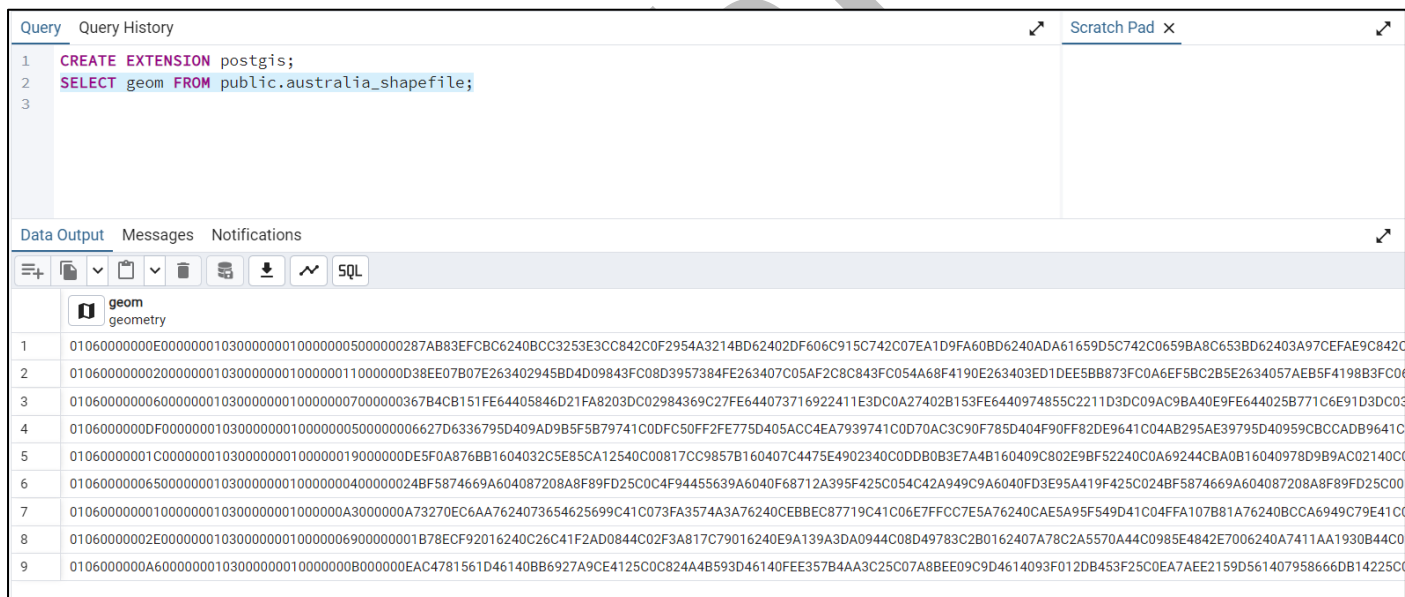


- Click on import and see the message in the log window. Query the data to ensure it has been imported correctly.





Shapefile is imported.







- Download the CSV file and view it in notepad.

- Write a query to create a table with attributes as the column headers of the CSV file.

- Create a table for the CSV file in pgAdmin

Execute query

Do you want to run this query -

```
1 CREATE TABLE states (  
2     gid SERIAL PRIMARY KEY,  
3     year INTEGER,  
4     ste_code TEXT,  
5     ste_name TEXT,  
6     ste_area_co TEXT,  
7     ste_type TEXT,  
8     ste_iso3166 TEXT,  
9     geom GEOMETRY  
10 );
```

☐ Don't ask again

Cancel Continue

Data Output Messages Geometry Viewer X Notifications

CREATE TABLE

Query returned successfully in 195 msec.

## D. Load data using COPY command

- Import the data from the CSV file to pgAdmin using the copy command.

```
13 COPY states(gid, year, ste_code, ste_name, ste_area_co, ste_type, ste_iso3166, geom)  
14 FROM 'C:\Program Files\PostgreSQL\16\aus_csvfile.csv'  
15 DELIMITER ','  
16 CSV HEADER;  
17  
18
```

Data Output Messages Geometry Viewer X Notifications

COPY 18

Query returned successfully in 132 msec.

- Run a select query to check successful loading of data.

18 `select * from states;`  
19

Data Output Messages Geometry Viewer X Notifications

	gid [PK] integer	year integer	ste_code text	ste_name text	ste_area_co text	ste_type text	ste_iso3166 text	geom geometry
1	1	2021	[2]	[Victoria]	AUS	state	VIC	0106000000E00000001030000000100000005000000287AB83EFCBC6240BCC3253E3A
2	2	2021	[1]	[New South Wales]	AUS	state	NSW	01060000000200000001030000000100000011000000D38EE07B07E263402945BD4D09
3	3	2021	[9]	[Other Territories]	AUS	territory	NULL	01060000000600000001030000000100000007000000367B4CB151FE64405846D21FA8
4	4	2021	[5]	[Western Australia]	AUS	state	WA	0106000000DF0000000103000000010000000500000006627D6336795D409AD9B5F5B7
5	5	2021	[4]	[South Australia]	AUS	state	SA	01060000001C00000001030000000100000019000000DE5F0A876BB1604032C5E85CA
6	6	2021	[7]	[Northern Territory]	AUS	territory	NT	0106000000650000000103000000010000000400000024BF5874669A604087208A8F89
7	7	2021	[8]	[Australian Capital Territory]	AUS	territory	ACT	010600000001000000010300000001000000A3000000A73270EC6AA762407365462569
8	8	2021	[6]	[Tasmania]	AUS	state	TAS	01060000002E0000000103000000010000000690000001B78ECF92016240C26C41F2AC
9	9	2021	[3]	[Queensland]	AUS	state	QLD	0106000000A60000000103000000010000000B000000EAC4781561D46140BB6927A9C1
10	10	2021	[2]	[Victoria]	AUS	state	VIC	0106000020AD1000000E000000010300000001000000050000000287AB83EFCBC6240B
11	11	2021	[1]	[New South Wales]	AUS	state	NSW	0106000020AD1000000200000001030000000100000011000000D38EE07B07E2634029
12	12	2021	[9]	[Other Territories]	AUS	territory	NULL	0106000020AD1000000600000001030000000100000007000000367B4CB151FE644058
13	13	2021	[5]	[Western Australia]	AUS	state	WA	0106000020AD100000DF0000000103000000010000000500000006627D6336795D409A
14	14	2021	[4]	[South Australia]	AUS	state	SA	0106000020AD1000001C00000001030000000100000019000000DE5F0A876BB160403

Total rows: 18 of 18 Query complete 00:00:00.225 Ln 18, Col 1

- Write queries in pgAdmin

19 `select gid from states;`  
20 `select gid,year from states where year = '2021';`

Data Output Messages Geometry Viewer X Notifications

	gid [PK] integer
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14

20 `select gid,year from states where year = '2021';`

Data Output Messages Geometry Viewer X Notifications

	gid [PK] integer	year integer
1	1	2021
2	2	2021
3	3	2021
4	4	2021
5	5	2021
6	6	2021
7	7	2021
8	8	2021
9	9	2021
10	10	2021
11	11	2021
12	12	2021
13	13	2021
14	14	2021

## Conclusion:

The process of importing a CSV file into PostgreSQL involves creating the appropriate table structure, ensuring the CSV file is properly formatted, and using the correct method to import the data. The `COPY` command is a powerful tool for server-side imports, but it requires that the file is accessible by the PostgreSQL server with appropriate permissions.

## Key takeaways:

1. Ensure the CSV file is placed in a directory accessible by the PostgreSQL server.
2. The file path and permissions play a crucial role in the success of the import operation.

By following these steps, we successfully imported CSV data into PostgreSQL, ensuring that all data is loaded into the database for further use or analysis.