

1. 在命令行中建立数据库 mawenzhuo, 并且连接到 postgres

(1) 使用 `createdb -U postgres -E UTF8 -e mawenzhuo` 命令创建用户为 postgres、名称为 mawenzhuo 的数据库 (这里 windows 系统下数据库名要放在最后)

(2) 使用 `psql -U postgres -d mawenzhuo` 连接到数据库, 进入数据库操作状态

(3) 使用 `CREATE EXTENSION postgis;` 以及 `CREATE EXTENSION postgis_raster;` 加入扩展 (这里的分号不能掉, 第一次出现错误就是在这里)

```
D:\Users\mwz\PostgreSQL\bin>createdb -U postgres -E UTF8 -e mawenzhuo
命令:
SELECT pg_catalog.set_config('search_path', '', false);
CREATE DATABASE mawenzhuo ENCODING 'UTF8';

D:\Users\mwz\PostgreSQL\bin>psql -U postgres -d mawenzhuo
用户 postgres 的口令:
psql (14.2)
输入 "help" 来获取帮助信息.

mawenzhuo=# CREATE EXTENSION postgis;
CREATE EXTENSION
mawenzhuo=# CREATE EXTENSION postgis_raster
mawenzhuo=# \q

D:\Users\mwz\PostgreSQL\bin>psql -U postgres -d mawenzhuo
用户 postgres 的口令:
psql (14.2)
输入 "help" 来获取帮助信息.

mawenzhuo=# CREATE EXTENSION postgis;
错误: 扩展 "postgis" 已经存在
mawenzhuo=# CREATE EXTENSION postgis_raster;
CREATE EXTENSION
mawenzhuo=#
mawenzhuo=# \l
```

名称	拥有者	字元编码	校对规则	Ctype	存取权限
mawenzhuo	postgres	UTF8	Chinese (Simplified)_China.936	Chinese (Simplified)_China.936	
postgis_32_sample	postgres	UTF8	Chinese (Simplified)_China.936	Chinese (Simplified)_China.936	
postgis	postgres	UTF8	Chinese (Simplified)_China.936	Chinese (Simplified)_China.936	
template0	postgres	UTF8	Chinese (Simplified)_China.936	Chinese (Simplified)_China.936	=c/postgres + postgres=CtC/postgres
template1	postgres	UTF8	Chinese (Simplified)_China.936	Chinese (Simplified)_China.936	=c/postgres + postgres=CtC/postgres

(5 行记录)

使用 `ogr2ogr` 命令行模式将 china。Shp 数据导入数据库 mawenzhuo

通过 `ogr2ogr` 命令和 `shp2pgsql` 进行矢量导入, 其中要注意在使用 `shp2pgsql` 命令的时候文件名后面一定要加入 `|` 再写 `psql -d mawenzhuo -U postgres`, 否则会报错

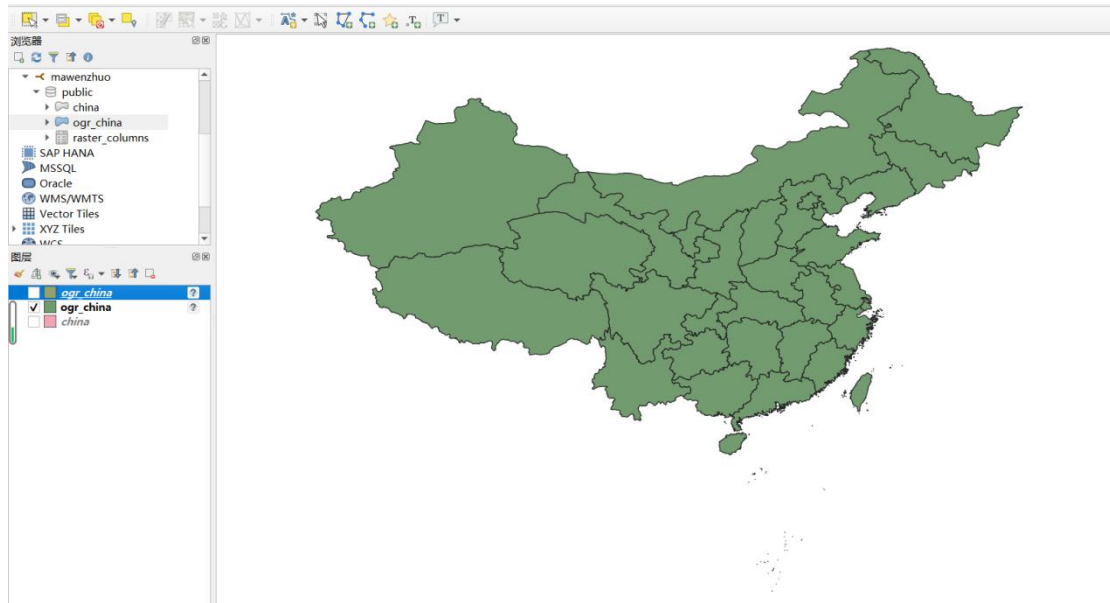
```
D:\Users\mwz\GIS\bin>ogr2ogr -f "PostgreSQL" "PG:dbname=mawenzhuo user=postgres password=090012" -lco PG_USE_COPY=YES -lco SHAPE_ENCODING=GBK -progress -update -append -gt -l -nln ogr_china D:/本科/时空数据处理与组织/data/china.shp
Warning 6: dataset PG:dbname=mawenzhuo user=postgres password=XXXXXX does not support layer creation option PG_USE_COPY
Warning 6: dataset PG:dbname=mawenzhuo user=postgres password=XXXXXX does not support layer creation option SHAPE_ENCODING
0...10...20...30...40...50...60...70...80...90...100 - done.

D:\Users\mwz\GIS\bin>shp2pgsql -s 4326 -W GBK D:/本科/时空数据处理与组织/data/china.shp public.china | psql -d mawenzhuo -U postgres
'shp2pgsql' 不是内部或外部命令, 也不是可运行的程序
或批处理文件。

D:\Users\mwz\GIS\bin>cd D:\Users\mwz\PostgreSQL\bin

D:\Users\mwz\PostgreSQL\bin>shp2pgsql -s 4326 -W GBK D:/本科/时空数据处理与组织/data/china.shp public.china | psql -d mawenzhuo -U postgres
Field area is an FTDouble with width 12 and precision 3
Field perimeter is an FTDouble with width 12 and precision 3
Field bou2_4m_ is an FTDouble with width 11 and precision 0
Field bou2_4m_id is an FTDouble with width 11 and precision 0
Shapefile type: Polygon
Postgis type: MULTIPLYGON[2]
用户 postgres 的口令:
SET
SET
BEGIN
CREATE TABLE
ALTER TABLE
addgeometrycolumn
```

用 QGIS 连接数据库 mawenzhuo 进行显示



2. 用 org2org 命令行模式将 org_china 导出到 geojson 数据文件 chinalayer.geojson 中

```
D:\Users\mwz\QGIS\bin>ogr2ogr -f GeoJson D:/本科/时空数据处理与组织/data/chinalayer.geojson "PG:dbname=mawenzhuo user=postgres password=090012" "ogr_china"
```

在 QGIS 中显示 chinalayer.geojson

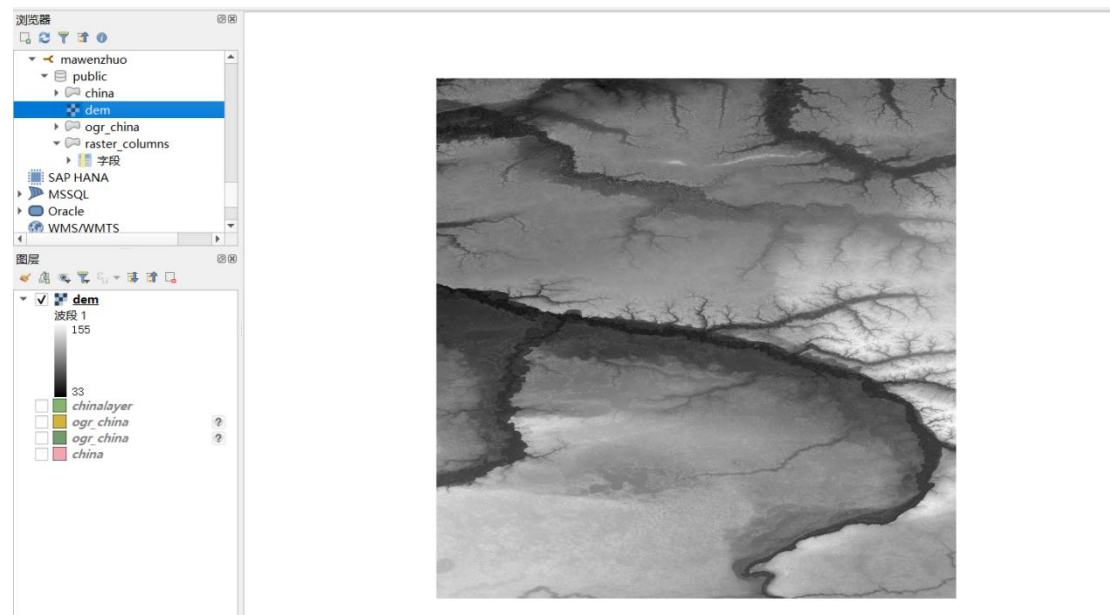


3. 使用命令行模式将 s51 文件数据导入到数据库 mawenzhuo 中

最开始我将 tif 文件放在中文目录下会报错

```
D:\Users\mwz\PostgreSQL\bin>raster2pgsql -s 4326 -l -C -M D:/Users/mwz/PostgreSQL/filedata/s51.tif -F -t 256x256 public.dem | psql -d mawenzhuo -U postgres
Processing 1/1: D:/Users/mwz/PostgreSQL/filedata/s51.tif
用户 postgres 的口令:
BEGIN
CREATE TABLE
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
```

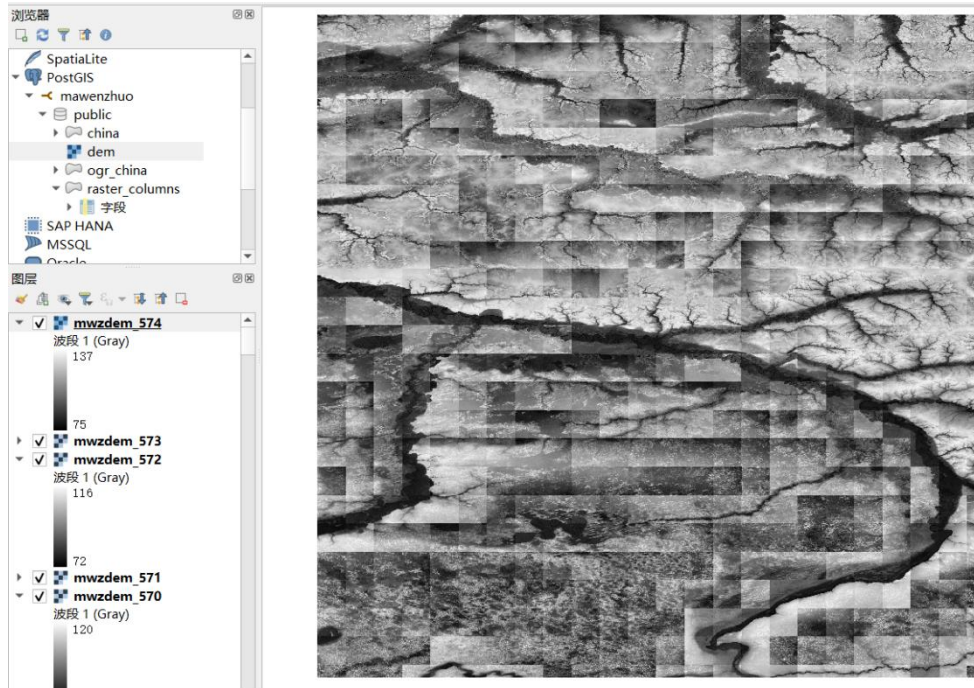
在 QGIS 连接数据库并显示



4. 将上面的 s51 数据使用 gdal_slate 导出到本地文件夹中

```
D:\Users\mwz\QGIS\bin>gdal_translate -of GTiff "PG:dbname=mawenzhuo schema=public table=dem user=postgres password=090012" -sds D:\本科\时空数据处理与组织\data\mwzdem.tif
Input file size is 256, 256
```

拖拽至 QGIS 中全部显示出来



*遇到的一些问题:

(1) 最开始我是使用的 linux 系统，但是在启动数据库连接时出现了无法启动服务的现象（在 log 日志里面查看之后发现是无法建立 TCP/IP 连接，在之后的尝试中还是无法创建数据库）。不过好在之后老师提供了一些解决此类问题的参考资料，对于 linux 系统下的操作我会在课后完成。

```
gorgeous@gorgeous-virtual-machine:~/桌面$ cd ..
gorgeous@gorgeous-virtual-machine:~$ pg_ctl -D pg -l pg/server.log start
waiting for server to start.... stopped waiting
pg_ctl: could not start server
Examine the log output.
gorgeous@gorgeous-virtual-machine:~$ pg_ctl -D pg -l pg/server.log start
waiting for server to start.... stopped waiting
pg_ctl: could not start server
显示应用程序 log output.
gorgeous@gorgeous-virtual-machine:~$
```

```
sockets
12 2022-03-07 13:13:47.183 CST [2918] LOG:  database system is shut down
13 2022-03-07 21:12:01.952 CST [2981] LOG:  starting PostgreSQL 12.9 (Ubuntu
12.9-0ubuntu0.20.04.1) on x86_64-pc-linux-gnu, compiled by gcc (Ubuntu
9.3.0-17ubuntu1~20.04) 9.3.0, 64-bit
14 2022-03-07 21:12:01.955 CST [2981] LOG:  could not bind IPv4 address
"127.0.0.1": 地址已在使用
15 2022-03-07 21:12:01.955 CST [2981] HINT:  Is another postmaster already
running on port 5432? If not, wait a few seconds and retry.
16 2022-03-07 21:12:01.955 CST [2981] WARNING:  could not create listen socket
for "localhost"
17 2022-03-07 21:12:01.956 CST [2981] FATAL:  could not create any TCP/IP
sockets
18 2022-03-07 21:12:01.956 CST [2981] LOG:  database system is shut down
19 2022-03-07 21:13:40.249 CST [2994] LOG:  starting PostgreSQL 12.9 (Ubuntu
12.9-0ubuntu0.20.04.1) on x86_64-pc-linux-gnu, compiled by gcc (Ubuntu
9.3.0-17ubuntu1~20.04) 9.3.0, 64-bit
20 2022-03-07 21:13:40.250 CST [2994] LOG:  could not bind IPv4 address
"127.0.0.1": 地址已在使用
21 2022-03-07 21:13:40.250 CST [2994] HINT:  Is another postmaster already
running on port 5432? If not, wait a few seconds and retry.
22 2022-03-07 21:13:40.250 CST [2994] WARNING:  could not create listen socket
for "localhost"
23 2022-03-07 21:13:40.250 CST [2994] FATAL:  could not create any TCP/IP
sockets
显示应用程序 21:13:40.250 CST [2994] LOG:  database system is shut down
```

纯文本 制表符宽度: 8 第 1 行, 第 1 列 插入

(2) 在使用使用 `createdb -U postgres -E UTF8 -e mawenzhuo` 创建数据库时，开始由于我把数据库名称放在前面，会出现报错，不过后来在老师的提醒下，改正过来