



# **C**Ơ SỞ DỮ LIỆU KHÔNG GIAN

---

Giảng viên: Kiều Tuấn Dũng, Nguyễn Tu Trung  
BM HTTT, Khoa CNTT, Trường ĐH Thủy Lợi

Hà Nội, 2019

# Nội dung

---

- ❖ Tổng quan về CSDL không gian
- ❖ PostgreSQL và POSTGIS
- ❖ Chuẩn đối tượng không gian mở WKT
- ❖ Làm việc với CSDL không gian

# Tổng quan về CSDL không gian

---

- ❖ Khái niệm về CSDL không gian
- ❖ Đặc trưng về CSDL không gian
- ❖ Các đối tượng cơ bản trong mô hình không gian
- ❖ Mối quan hệ không gian
- ❖ Kết hợp hình học và mô hình dữ liệu DBMS

# Khái niệm về CSDL không gian

---

- ❖ Là một hệ thống csdl quan hệ cung cấp các kiểu dữ liệu không gian trong mô hình dữ liệu và các ngôn ngữ truy vấn
- ❖ Hỗ trợ các kiểu dữ liệu không gian như: **Point, Line, Polygon...**
- ❖ CSDL cung cấp mô hình trừu tượng cơ bản cho cấu trúc của thực thể hình học trong không gian cũng như mối quan hệ giữa chúng như: **giao nhau, thuộc nhau...**
- ❖ Cung cấp các kiểu đánh chỉ mục để tăng tốc độ thực thi các bảng dữ liệu lớn

# Đặc trưng về CSDL không gian

---

- ❖ Giống CSDL quan hệ thông thường:
  - ❖ Truy vấn SQL điển hình như câu lệnh SELECT
  - ❖ Cơ chế đánh chỉ mục không gian để tăng tốc hoạt động của cơ sở dữ liệu
- ❖ Bổ sung: thực thi đa dạng các thao tác không gian
  - ❖ Đo lường không gian: tìm khoảng cách giữa các điểm, các vùng...
  - ❖ Hàm không gian: sửa đổi các hàm hiện thời để tạo ra những hình mới: hàm tìm điểm hay vùng giao nhau...
  - ❖ Xác nhận không gian: cho phép thực hiện những truy vấn True/False
  - ❖ Hàm tạo: tạo ra các hình mới từ phân tích không gian
  - ❖ Hàm theo dõi: các câu truy vấn trả về thông tin cụ thể như : vị trí tâm của một đường tròn hay điểm đầu, điểm cuối của một đường

# Đối tượng cơ bản trong mô hình không gian

❖ Các đối tượng cơ bản:

❖ Point

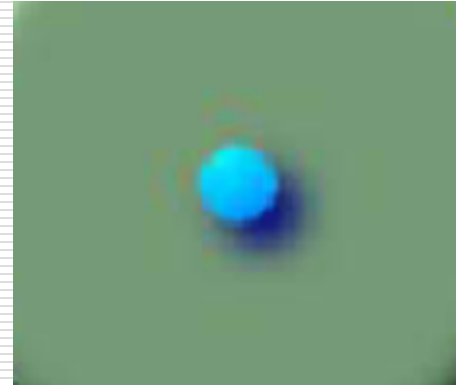
❖ Line

❖ Polygon

# Point

---

- ❖ Dùng để hiển thị một đối tượng mà chỉ có vị trí của nó trong không gian
- ❖ VD:
  - ❖ Một thành phố có thể được mô phỏng như 1 điểm trong mô hình mô tả 1 khu vực rộng lớn về địa lý
- ❖ Đặc điểm:
  - ❖ Là tọa độ đơn
  - ❖ Không cần thể hiện chiều dài và diện tích
  - ❖ Dùng để hiển thị cho các vùng khi chúng được hiển thị ở quy mô nhỏ
  - ❖ Không có phép đo nào được áp dụng cho điểm



# Line

---

- ❖ Là tập hợp đầy các điểm, mô tả đối tượng địa lý dạng tuyến tính
- ❖ Ví dụ:
  - ❖ Các con đường: quốc lộ, tỉnh lộ, đường sắt có thể biểu diễn bởi line trong mô hình mô tả 1 khu vực rộng lớn về địa lý
- ❖ Đặc điểm:
  - ❖ Là một dãy các cặp tọa độ
  - ❖ Bắt đầu và kết thúc là một điểm
  - ❖ Các đường nối với nhau hoặc cắt nhau tại một điểm
  - ❖ Có phép đo chiều dài đối với đường

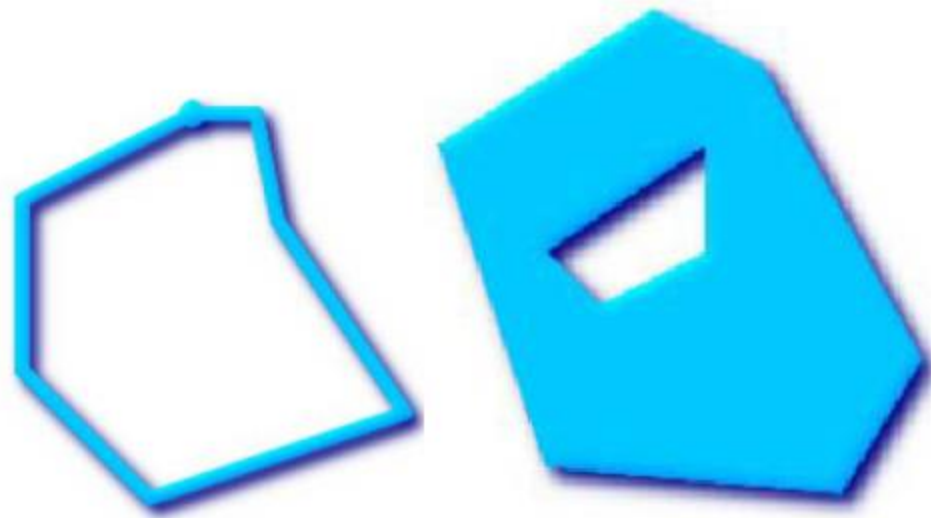




# Polygon

---

- ❖ Định nghĩa:
  - ❖ Được xác định bởi ranh giới các đường thẳng
  - ❖ Các đối tượng địa lý có diện tích và đóng kín bởi một đường được gọi là vùng
- ❖ VD: hồ, các toà nhà, công viên, thành phố... (trường hợp muốn mô tả chi tiết hơn mức điểm về mặt hình dạng)
- ❖ Đặc điểm:
  - ❖ Vùng được mô tả bằng tập các đường và điểm
  - ❖ Vùng được bao bởi nhiều đường
  - ❖ Có phép tính chu vi và diện tích cho đa giác



# Mối quan hệ không gian

---

- ❖ Chỉ ra mối quan hệ giữa các đối tượng trong không gian
- ❖ Các mối quan hệ không gian phổ biến:
  - ❖ Quan hệ topo: như liền kề, phân chia... và các phép biến đổi topo như phép dịch chuyển, phép xoay...
  - ❖ Quan hệ định hướng
    - ❖ Quan hệ định hướng bên trong: chỉ ra một đối tượng được đặt bên trong một đối tượng tham chiếu
    - ❖ Quan hệ định hướng bên ngoài: chỉ ra đối tượng được đặt bên ngoài một đối tượng tham chiếu
  - ❖ Quan hệ khoảng cách: Chỉ ra khoảng cách từ đối tượng cụ thể đến đối tượng tham chiếu

# Kết hợp hình học và mô hình dữ liệu DBMS

- ❖ Ý tưởng chính: Kết hợp kiểu dữ liệu thuộc tính (như integer, string...) và hình học (như Point, Line...) để thể hiện các “đối tượng không gian”- các đối tượng có thể là dòng sông, đất nước, thành phố...
- ❖ VD: Mô tả đặc điểm của sông, hay mô tả đặc điểm của thành phố ta có các bảng dữ liệu:
  - ❖ Rivers (rname: STRING, route : LINE)
  - ❖ Cities (cname: STRING, center : POINT, ext : POLYGON)
- ❖ Để biểu diễn các đối tượng không gian trong mô hình 2 chiều dùng cách biểu diễn hệ tọa độ
- ❖ VD:
  - ❖ Biểu diễn một điểm POINT (0,0): điểm nằm tại tọa độ (0,0)
  - ❖ Biểu diễn một đường LINE (0 0, 1 1, 1 2): đường nối 3 điểm nằm lần lượt tại các tọa độ (0,0) -> (1,1) -> (1,2)

# PostgreSQL và POSTGIS

---

- ❖ PostgreSQL
- ❖ Cài đặt PostgreSQL
- ❖ Quản trị PostgreSQL với pgAdmin
- ❖ PostGIS là gì?
- ❖ Đặc điểm của PostGIS

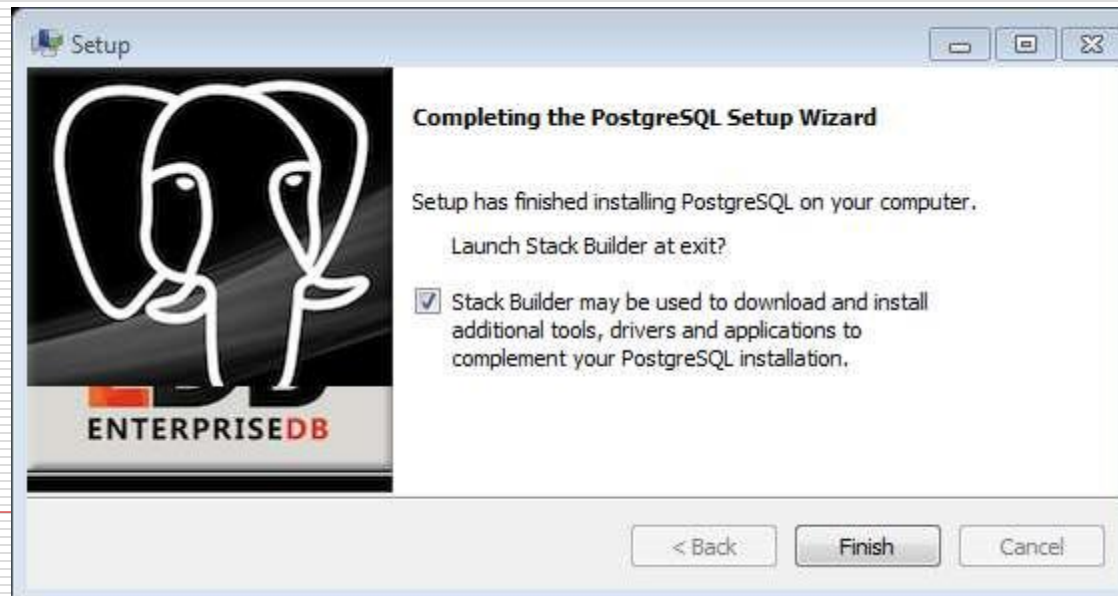
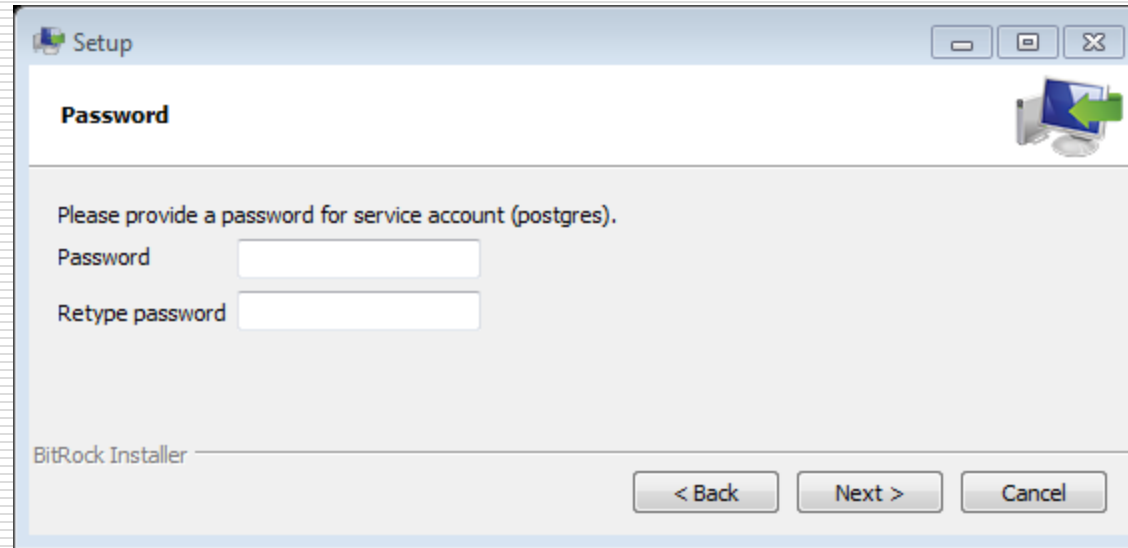
# PostgreSQL

---

- ❖ Là hệ thống quản trị cơ sở dữ liệu quan hệ đối tượng dựa trên POSTGRES, được phát triển tại trường đại học California tại phòng nghiên cứu máy tính Berkeley
- ❖ Hỗ trợ một phần rất lớn cho SQL chuẩn và cung cấp nhiều tính năng hiện đại như:
  - ❖ Các truy vấn phức tạp
  - ❖ Khóa ngoài
  - ❖ Trigger
  - ❖ Khung nhìn
  - ❖ Tính toàn vẹn của các giao dịch
  - ❖ Kiểm tra truy cập đồng thời đa phiên bản
- ❖ Người dùng có thể thêm kiểu dữ liệu, hàm, toán tử, hàm tập hợp, phương thức đánh chỉ mục và ngôn ngữ thủ tục

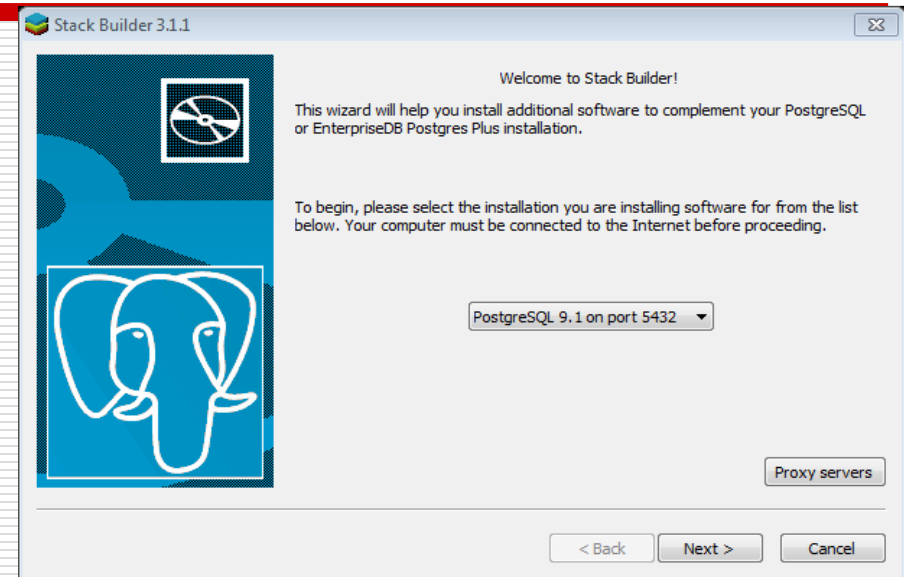
# Cài đặt PostGreSQL

- ❖ Tải PostgreSQL: postgresql-9.5.18-3-windows
- ❖ Các bạn tiến hành cài đặt bình thường như các chương trình bình thường
- ❖ Nhập và nhớ password
- ❖ Tick vào tục cài thêm spatial extension: Postgis trong stack builder

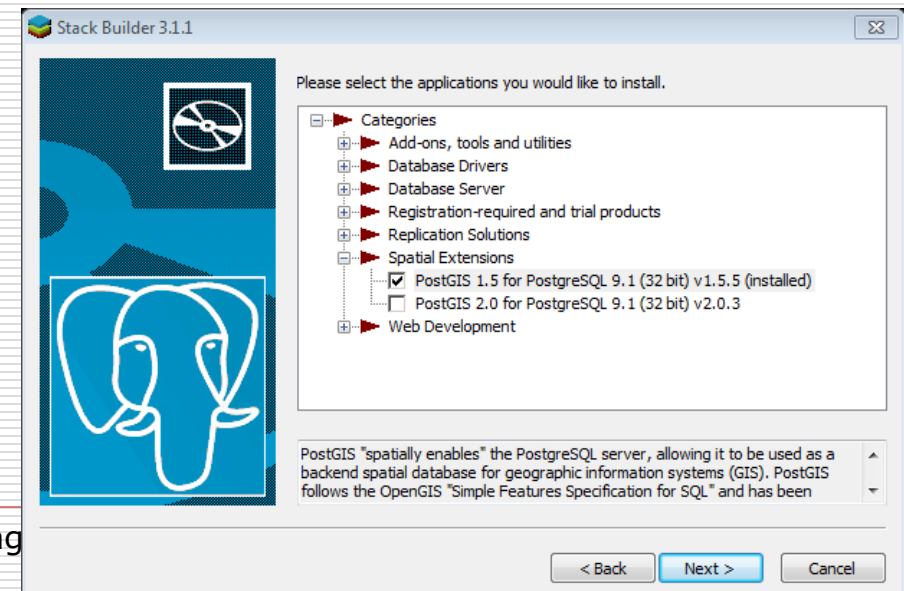


# Cài đặt PostGreSQL

- ❖ Chọn PostGreSQL ứng với phiên bản cài đặt: 9.5
- ❖ Cổng mặc định là 5432

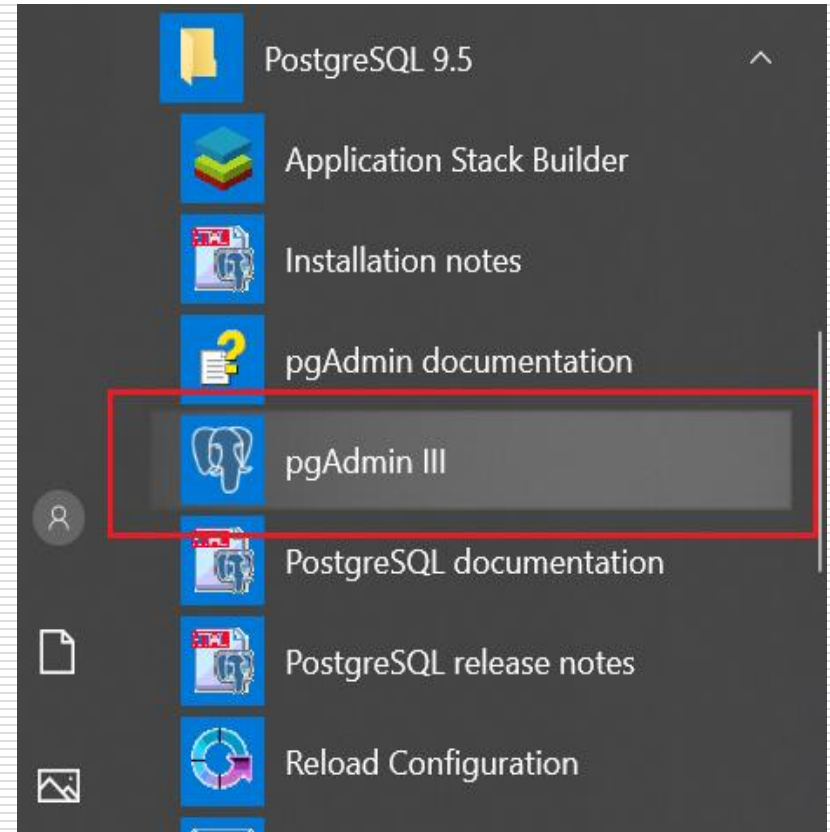


- ❖ Trong spatial extension bạn có thể chọn cài bản PostGIS ứng với PostgresSQL 9.5



# Quản trị PostgreSQL với pgAdmin

- ❖ pgAdmin miễn phí và là công cụ quản trị giao diện đồ họa mà nguồn mở cho PostgreSQL, hỗ trợ tất cả các tính năng của PostgreSQL
- ❖ Có thể dùng trên Linux, FreeBSD, Solaris, Mac OSX và Window
- ❖ Khởi động pgAdmin III

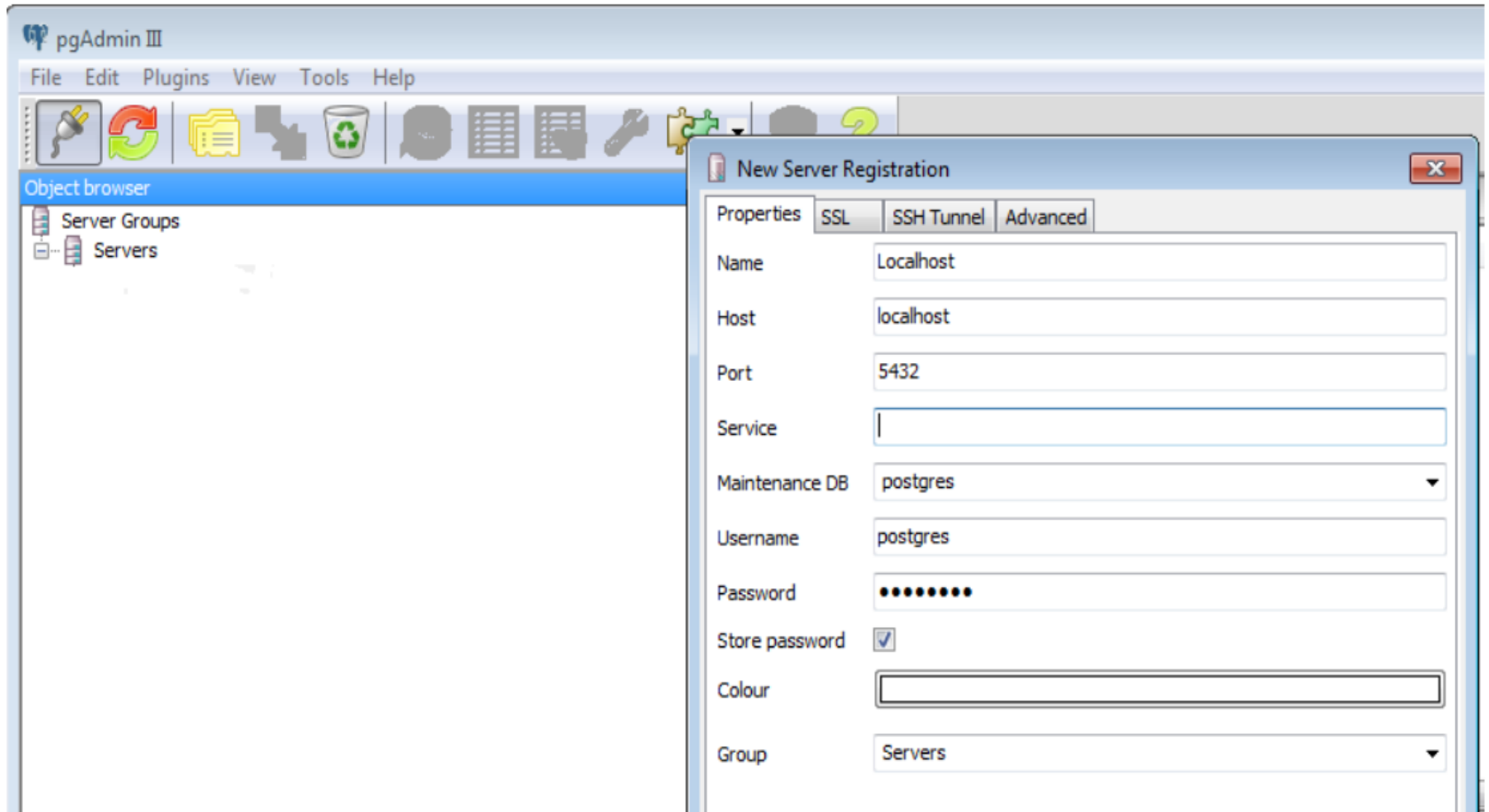




1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

# Quản trị PostgreSQL với pgAdmin

- ❖ Tạo kết nối mới đến localhost
- ❖ Nhập các thông tin liên quan để kết nối



# Quản trị PostgreSQL với pgAdmin

## ❖ Truy vấn CSDL dễ dàng

The screenshot displays the pgAdmin 4 interface. The top window is titled "Query - TestCSDL on postgres@localhost:5432 - [D:\GIS\PostgreSQL\Sql\_Spatial\_Exa...". The menu bar includes File, Edit, Query, Favourites, Macros, View, and Help. The toolbar contains various icons for file operations, editing, and execution. The main pane is the "SQL Editor", which has tabs for "SQL Editor" and "Graphical Query Builder". Below the tabs is a "Previous queries" list with "Delete" and "Delete All" buttons. The SQL Editor contains the following queries:

```
Select ST_AsText(ST_Intersection('POINT(1 1)::geometry, 'LINESTRING(2 0,0 2)::ge ^
Select ST_AsText(ST_Intersection('POLYGON((0 3,3 5,5 5,0 3))::geometry, 'POLYGON,
Select ST_AsText(geom) from "CMR_adml";
Select ST_AsGeoJson(geom) from "CMR_adml";

SELECT ST_AsGeoJson(geom) as geo from "CMR_adml" where ST_Within('SRID=4326;POINT
select distinct(ST_SRID(geom)) from "CMR_adml";

Select 'SRID=4326;POINT(12 5)::geometry;
```

The "Output pane" at the bottom shows the "Data Output" tab. It displays a table with the following data:

	st_astext text
1	MULTIPOLYGON(((11.2236 6.260200000000011,11.27516 6.250609999999994,11.28955
2	MULTIPOLYGON(((12.57034 3.379459999999994,12.66651000000001 3.289300000000008,
3	MULTIPOLYGON(((9.259584000000007 3.958471999999997,9.259584000000007 3.9581940
4	MULTIPOLYGON(((12.48615000000001 7.939979999999999,12.56689 7.921390000000003,

The status bar at the bottom shows "OK.", "DOS", "Ln 30, Col 1, Ch 1910", "39 chars", "10 rows.", and "153 msec".

# PostGIS là gì ?

---

- ❖ Được Refraction Research Inc phát triển, như một dự án nghiên cứu công nghệ CSDL không gian
- ❖ Hỗ trợ đối tượng địa lý cho CSDL đối tượng quan hệ PostgreSQL
- ❖ “Kích hoạt khả năng không gian” cho PostgreSQL, cho phép PostgreSQL sử dụng như một CSDL không gian phụ trợ cho các hệ thống thông tin địa lý (GIS)

# Đặc điểm của PostGIS

---

- ❖ Bao gồm tất cả các đặc điểm của CSDL không gian
- ❖ Các kiểu dữ liệu hình học:
  - ❖ Point, Linestring, Polygon, Multipoint, multilinestring, Multipolygons và Geometrycollection
- ❖ Các toán tử không gian cho phép xác định các phép đo không gian địa lý:
  - ❖ Tính diện tích, khoảng cách, độ dài, chu vi: ST\_Area(), ST\_Length(), ST\_Perimeter(), ST\_Distance()...
- ❖ Các toán tử không gian cho phép xác định không gian địa lý:
  - ❖ Phép hợp, so sánh sự khác nhau giữa các đối tượng hình học: ST\_Difference(), ST\_Buffer()...
- ❖ Đánh chỉ mục không gian tốc độ cao sử dụng GiST hoặc R-tree
- ❖ Kết hợp giữa truy vấn không gian hoặc truy vấn không có không gian

# Chuẩn đối tượng không gian mở WKT

---

- ❖ Đối tượng GIS hỗ trợ bởi PostGIS được định nghĩa bởi OpenGIS Consortium (OGC)
- ❖ Đặc tả OpenGIS định nghĩa cách thể hiện chuẩn của đối tượng không gian đó là dạng Well-Know Text (WKT)
- ❖ WKT bao gồm các thông tin về kiểu của đối tượng và các tọa độ đang đối tượng
- ❖ VD về hiển thị dạng WKT của đối tượng không gian:
  - ❖ POINT(0 0)
  - ❖ LINESTRING(0 0,1 1,1 2)
  - ❖ POLYGON((0 0,4 0,4 4,0 4,0 0))
  - ❖ MULTIPOINT(0 0,1 2)
  - ❖ MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
  - ❖ MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0)),(1 1,2 1,2 2,1 2,1 1)),((-1 -1,-1 -2,-2 -2,-2 -1,-1 -1))

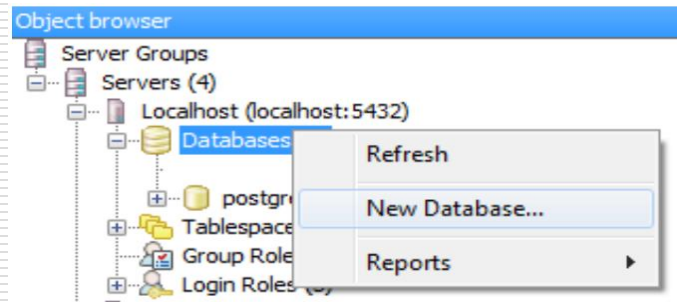
# Làm việc với CSDL không gian

---

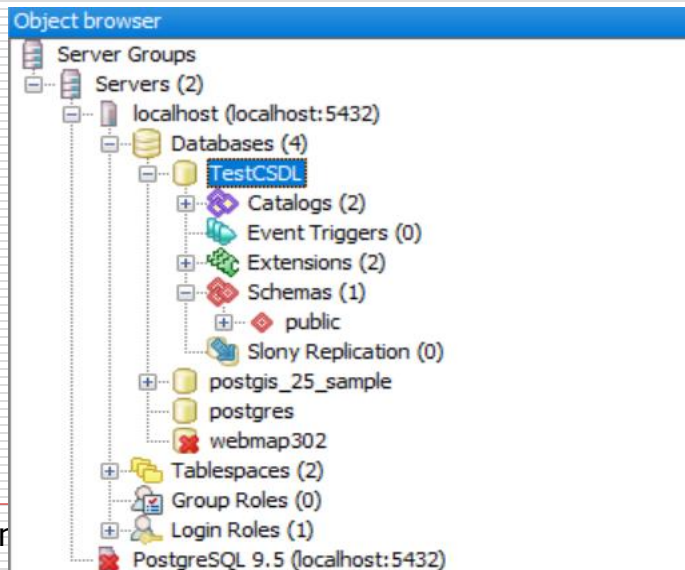
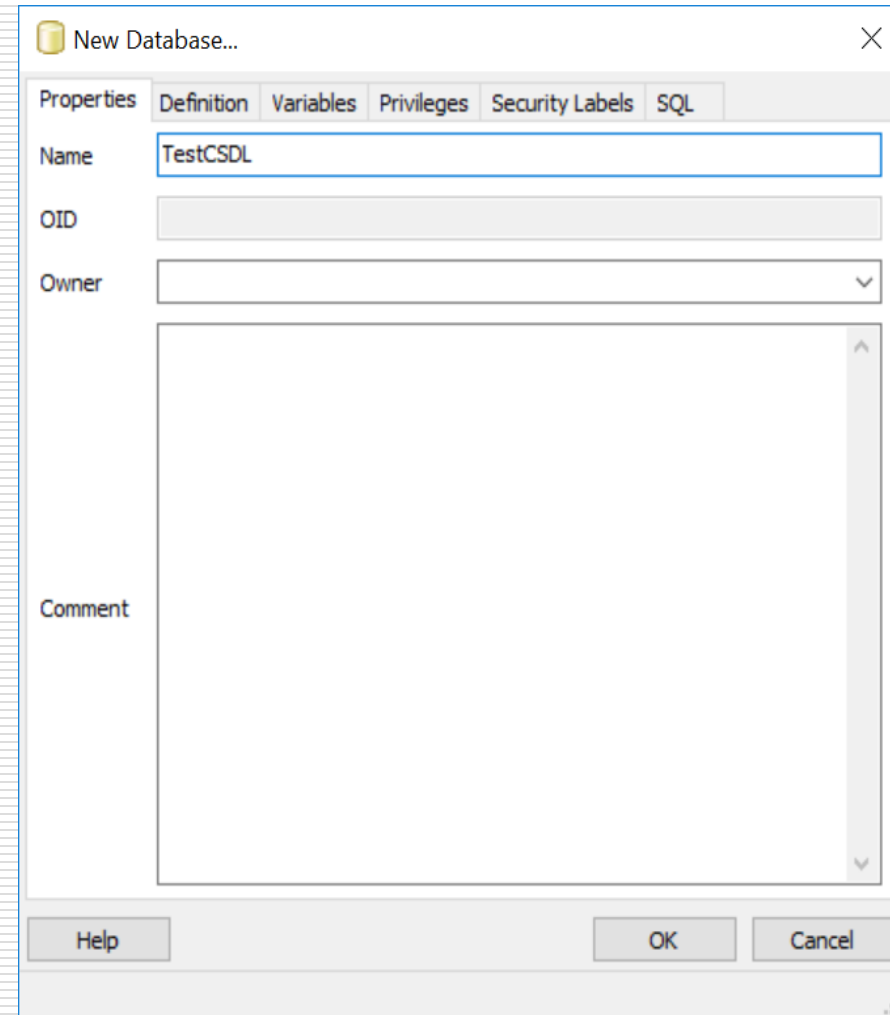
- ❖ Tạo CSDL không gian
- ❖ Thêm dữ liệu shapefile vào CSDL không gian
- ❖ Kiểm tra mã SRID
- ❖ Bảng siêu dữ liệu GEOMETRY\_COLUMNS
- ❖ Bảng siêu dữ liệu SPATIAL\_REF\_SYS
- ❖ Bảng dữ liệu không gian
- ❖ Một số hàm trong POSTGIS

# Tạo CSDL không gian

- ❖ B1: Click chuột phải mục Database, chọn New Database



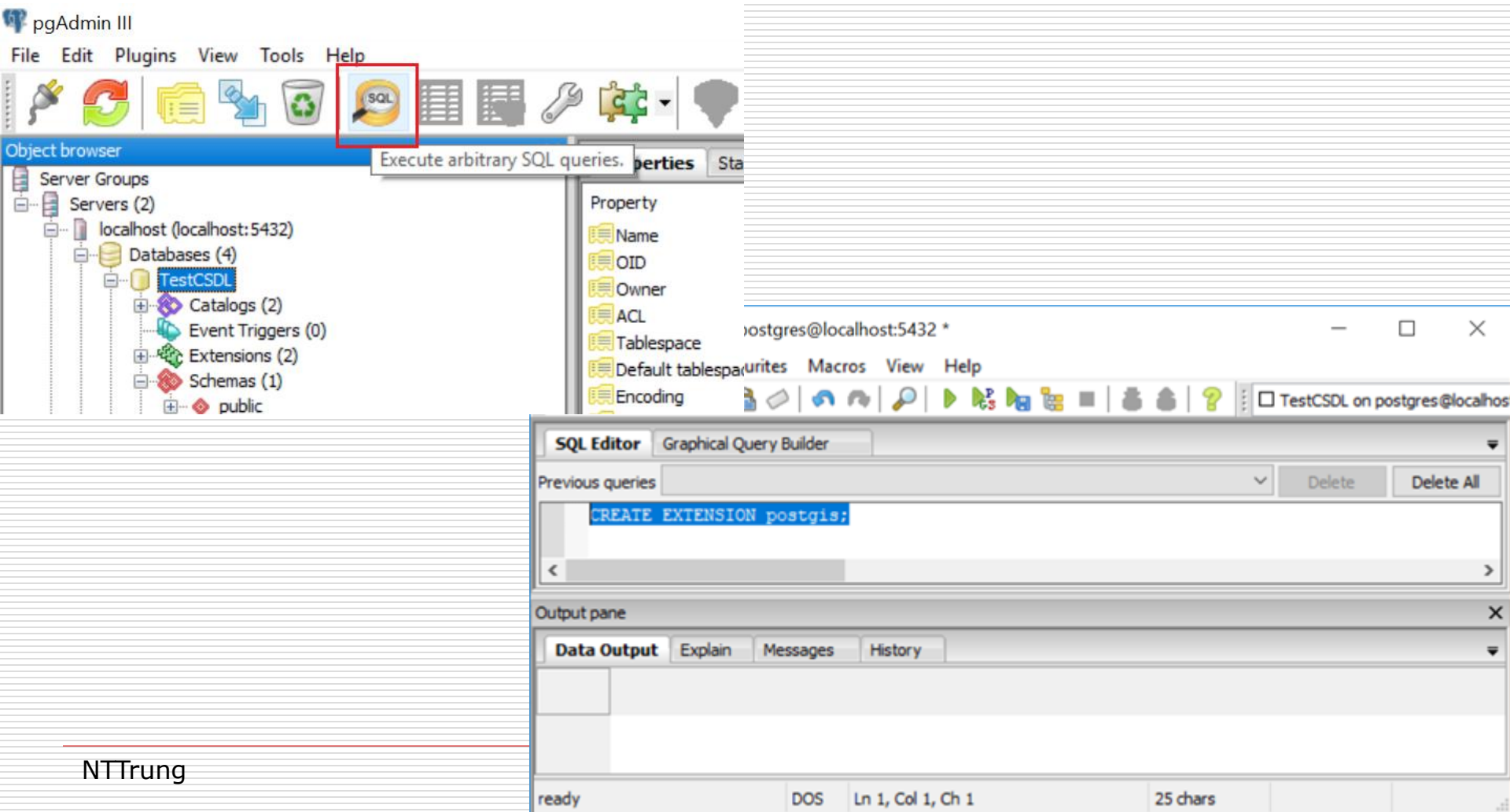
- ❖ B2: Nhập tên Database
- ❖ B3: Click OK





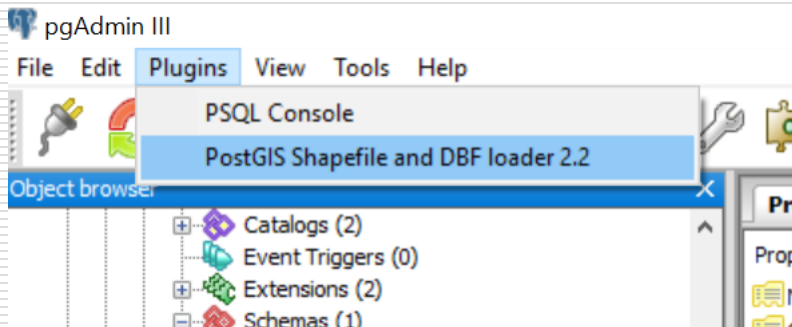
# Tạo CSDL không gian

- ❖ B4: Để CSDL có thể lưu trữ dữ liệu không gian, mở cửa sổ SQL và thực thi lệnh: **CREATE EXTENSION postgis;**

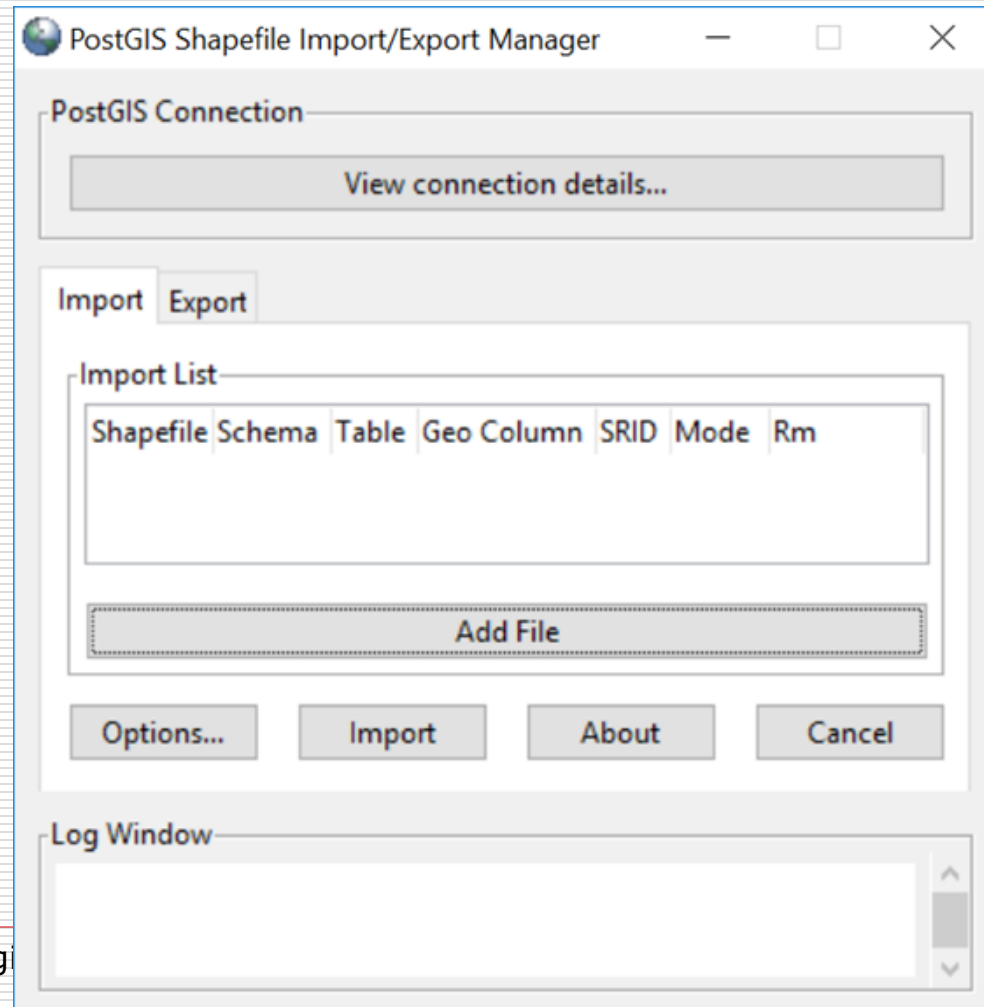


# Thêm dữ liệu shapefile vào CSDL không gian

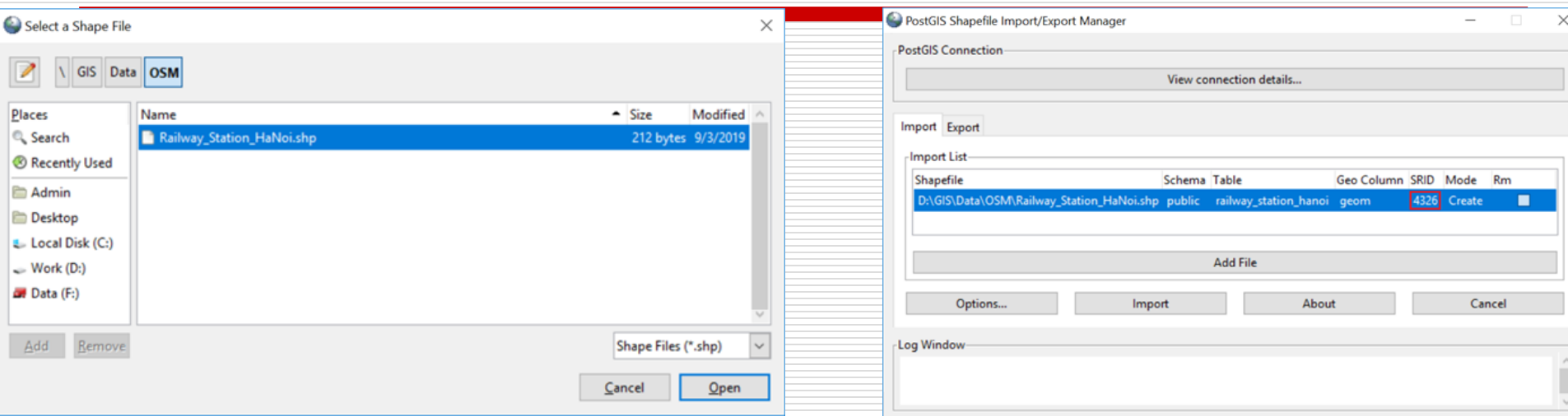
- ❖ B1: Click menu Plugins -> Chọn PostGIS Shapefile and DBF loader 2.2 mở hộp thoại PostGIS Shapefile Import/Export



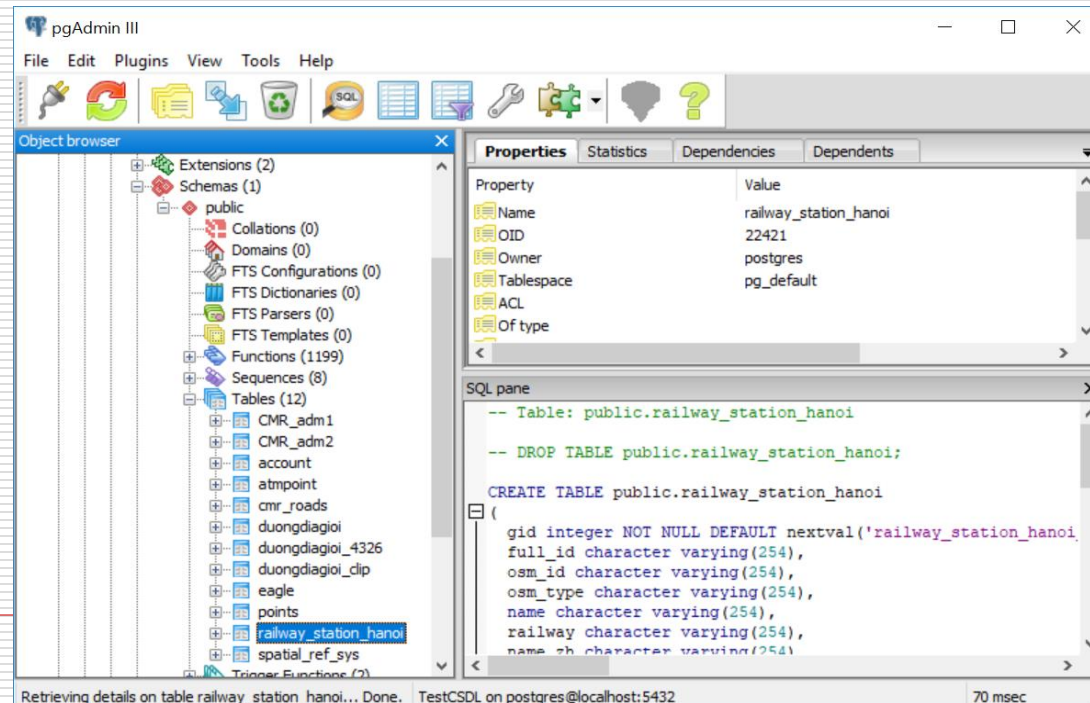
- ❖ B2: Click Add File mở hộp thoại tìm shapfile



# Thêm dữ liệu shapefile vào CSDL không gian

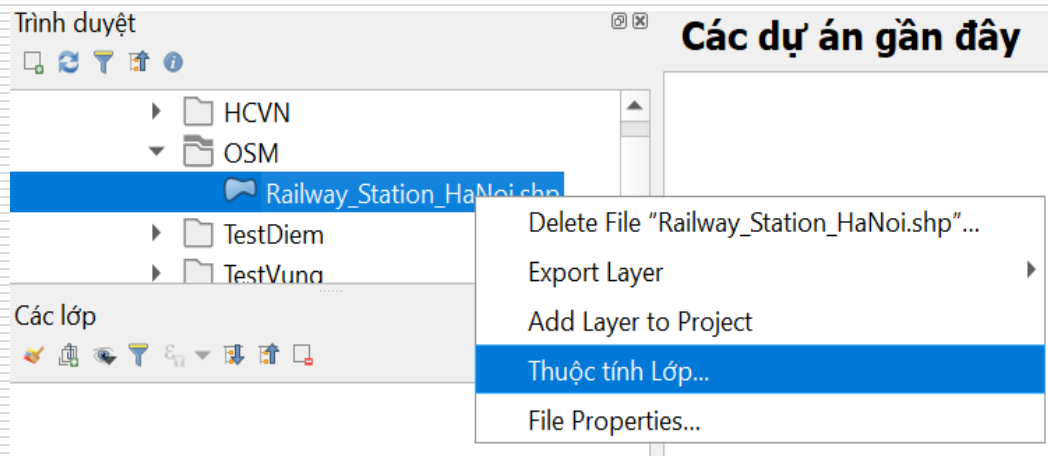


- ❖ B3: Tìm shapefile muốn nạp vào CSDL và click Open
- ❖ B4: Nhập mã SRID
- ❖ B5: Click Import

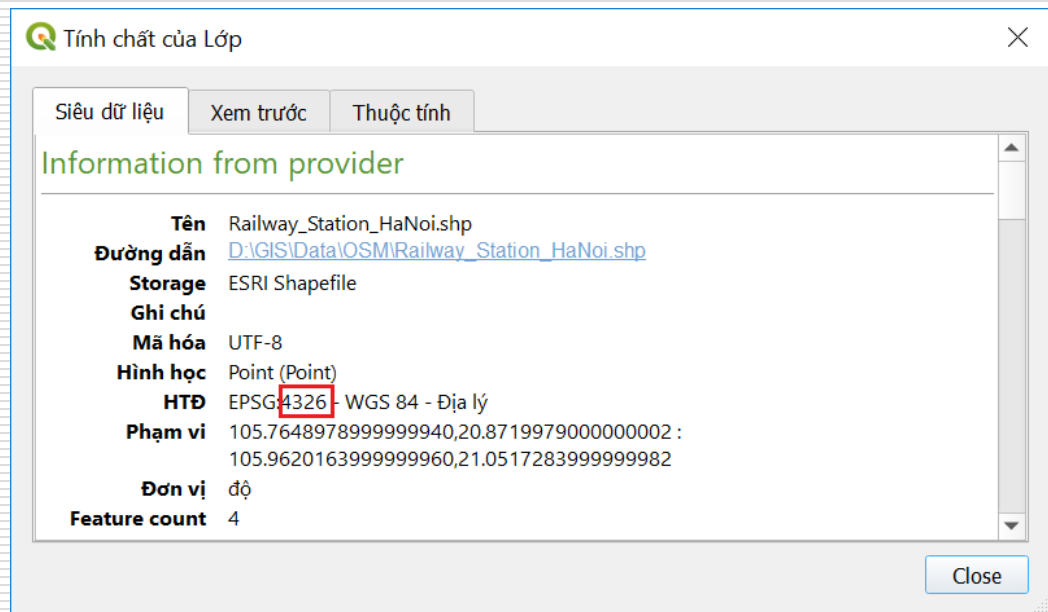


# Kiểm tra mã SRID

❖ B1: Click chuột phải vào file dữ liệu -> chọn Thuộc tính lớp để mở hộp thoại Thuộc tính lớp



❖ B2: Xem mã SRID tại dòng HTĐ, sau chữ EPSG



# Bảng siêu dữ liệu GEOMETRY\_COLUMNS

---

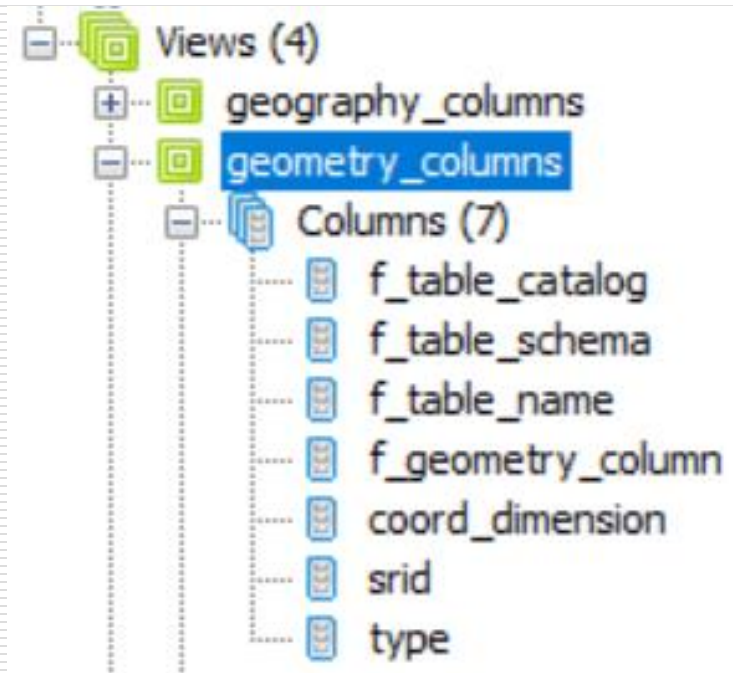
- ❖ Khi một cơ sở dữ liệu không gian được kích hoạt với PostGIS, có 2 hàng siêu dữ liệu được tạo ra: GEOMETRY\_COLUMNS và SPATIAL\_REF\_SYS
- ❖ Bảng GEOMETRY\_COLUMNS mô tả về bảng đã tồn tại được kích hoạt không gian trong cơ sở dữ liệu
- ❖ Khi thực thi hàm AddGeometryColumn() có thể dùng để thêm đồng thời một cột không gian vào bảng phi không gian và cập nhật bảng GEOMETRY\_COLUMNS
- ❖ Cấu trúc bảng GEOMETRY\_COLUMNS
  - ❖ f\_table\_catalog: tên CSDL
  - ❖ f\_table\_schema: tên schema
  - ❖ f\_table\_name: tên bảng
  - ❖ f\_geometry\_column: tên cột không gian

# Bảng siêu dữ liệu GEOMETRY\_COLUMNS

## ❖ Cấu trúc bảng GEOMETRY\_COLUMNS (tiếp)

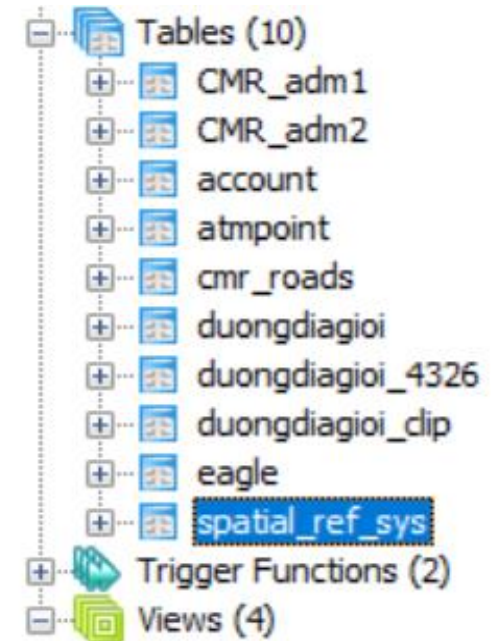
- ❖ coord\_dimension: số chiều không gian của dữ liệu
- ❖ srid: mã hệ thống tham chiếu không gian, là khóa ngoài tham chiếu đến bảng SPATIAL\_REF\_SYS

- ❖ Type: mô tả kiểu hình học được mô tả trong bảng, sử dụng một trong các kiểu sau: POINT, LINESTRING, POLYGON, MULTIPOINT, MULTILINESTRING, MULTIPOLYGON, GEOMETRYCOLLECTION



# Bảng siêu dữ liệu SPATIAL\_REF\_SYS

- ❖ Mô tả hệ thống tham chiếu không gian
- ❖ Mỗi kiểu hình học trong cơ sở dữ liệu không gian có liên quan đến số SRID
- ❖ Cấu trúc bảng SPATIAL\_REF\_SYS
  - ❖ SRID: là định danh duy nhất (có thể hiểu như khóa chính của bảng dữ liệu), là tham số nhận diện tham chiếu không gian
  - ❖ auth\_name: mô tả cơ quan hoặc tổ chức định nghĩa và sử dụng hệ thống tham chiếu
  - ❖ auth\_srid: là số nguyên được gán bởi cơ quan hoặc tổ chức
  - ❖ srtext, proj4text: hiển thị WKT, thông tin phép chiếu của hệ thống tham chiếu không gian





# Bảng không gian

---

- ❖ Bảng không gian là một bảng chứa một hoặc nhiều cột không gian
- ❖ Bao gồm những cột có kiểu dữ liệu thông thường và những cột kiểu dữ liệu không gian
- ❖ Cột không gian chỉ có thể chấp nhận kiểu dữ liệu hình học, bao gồm: Point, Multipoint, Linestring, MultiLinestring, Polygon, Multipolygon
- ❖ Tạo bảng không gian:
  - ❖ Cách thông thường
  - ❖ Dùng hàm AddGeometryColumn



# Cách tạo bảng không gian thông thường

---

- ❖ Tạo bảng với câu lệnh CREATE TABLE, và một thuộc tính của bảng sẽ có kiểu dữ liệu đang “geometry”
  - ❖ Ví dụ: CREATE TABLE points (name varchar, point geometry);
- ❖ Chú ý: khi chèn dữ liệu vào bảng không gian cần chú ý đến trường có kiểu dữ liệu dạng “geometry”, định dạng dữ liệu nhập phải theo chuẩn định dạng WKT, ví dụ:
  - ❖ Để nhập dữ liệu cho đối tượng POINT có tọa độ (0,0), chúng ta dùng dạng POINT(0 0)
  - ❖ Để nhập dữ liệu cho đối tượng LINESTRING nối tọa độ (0,0) và (3, 4), chúng ta dùng dạng LINESTRING (0 0,3 4);
- ❖ Ví dụ thêm điểm vào bảng points:
  - ❖ INSERT INTO points VALUES ('a', 'POINT(0 0)');

# Tạo bảng dùng AddGeometryColumn

---

- ❖ B1: Tạo bảng thông thường (không phải bảng dữ liệu không gian)
- ❖ B2: Thêm cột không gian vào bảng sử dụng hàm “AddGeometryColumn”:  
AddGeometryColumn(<schema\_name>,<table\_name>,<column\_name>,<srid>,<type>,<dimension>)
  - ❖ <schema\_name>: tên schema của bảng cần thêm cột không gian
  - ❖ <table\_name>: tên bảng cần thêm cột không gian
  - ❖ <column\_name>: tên cột không gian cần thêm
  - ❖ <srid>: mã duy nhất của hệ thống tham chiếu không gian trong phạm vi của CSDL
  - ❖ <type>: xác định kiểu hình học cho cột
  - ❖ <dimension>: số chiều dữ liệu

# Tạo bảng dùng AddGeometryColumn

---

- ❖ Nếu đang ở sơ đồ hiện tại thì bỏ qua thông số <schema\_name>: AddGeometryColumn (<table\_name>,<column\_name>,<srid>,<type>,<dimension>)
- ❖ Ví dụ: Tạo bảng points\_funccol có một cột không gian
  - ❖ Tạo bảng points\_funccol: Create table points\_funccol (id int, name varchar);
  - ❖ Thêm trường the\_geom: Select AddGeometryColumn('public','points\_funccol','the\_geom',-1,'POINT',2);
  - ❖ Thêm dữ liệu: INSERT INTO points\_funccol VALUES (0, 'a', 'POINT(0 0)');

# Một số hàm trong POSTGIS

---

- ❖ Nhóm hàm điều khiển
  - ❖ AddGeometryColumn
  - ❖ DropGeometryColumn
  - ❖ DropGeometryTable
- ❖ Hàm khởi tạo hình học
- ❖ Hàm trả về kiểu hình học ở đầu ra

# AddGeometryColumn

---

## ❖ Chức năng:

- ❖ Thêm một cột hình học vào hàng đã tồn tại. Hàm này rất quan trọng trong việc tạo bảng trong CSDL không gian

## ❖ Cú pháp:

- ❖ `text AddGeometryColumn(varchar table_name, varchar column_name, integer srid, varchar type, integer dimension);`
- ❖ `text AddGeometryColumn(varchar schema_name, varchar table_name, varchar column_name, integer srid, varchar type, integer dimension);`

# DropGeometryColumn

---

- ❖ Chức năng:
  - ❖ Loại bỏ một cột hình học từ bảng không gian
- ❖ Cú pháp:
  - ❖ text DropGeometryColumn(varchar table\_name, varchar column\_name);
  - ❖ text DropGeometryColumn(varchar schema\_name, varchar table\_name, varchar column\_name);
- ❖ Ví dụ:
  - ❖ `SELECT DropGeometryColumn('public','points_funccol','the_geom');`

# DropGeometryTable

---

- ❖ Chức năng:
  - ❖ Loại bỏ bảng và tất cả những gì tham chiếu trong cột hình học
- ❖ Cú pháp:
  - ❖ `text DropGeometryTable(varchar table_name);`
  - ❖ `text DropGeometryTable(varchar schema_name, varchar table_name);`
- ❖ Ví dụ:
  - ❖ `SELECT DropGeometryTable('public','points_funccol');`

# Hàm khởi tạo hình học

---

- ❖ Tên hàm:
  - ❖ `ST_GeometryFromText`
- ❖ Chức năng:
  - ❖ Trả về giá trị được chỉ định `ST_Geometry` từ hiển thị WKT
- ❖ Cú pháp:
  - ❖ `geometry ST_GeometryFromText(text WKT);`
- ❖ Ví dụ:
  - ❖ `Select ST_GeomFromText('POINT(1 0)');`



# Hàm trả về kiểu hình học ở đầu ra

---

- ❖ Tên hàm:
  - ❖ ST\_AsText
- ❖ Chức năng:
  - ❖ Trả về hiển thị dạng WKT của hình
- ❖ Cú pháp:
  - ❖ text ST\_AsText(geometry g);
- ❖ Ví dụ:
  - ❖ **Select**  
ST\_AsText('010100000000000000000000F03F0000000000000000');  
trả về POINT(1 0)
  - ❖ **Select ST\_AsText(ST\_GeomFromText('POINT(1 0)'));** trả về  
POINT(1 0)

