

THỰC HÀNH LAB02

Nội dung thực hành:

Truy xuất dữ liệu với:

- Các hàm tổng hợp: COUNT, MIN, MAX, AVG, SUM
 - Hàm xử lý chuỗi
 - GROUP BY clause
 - HAVING clause
 - JOIN
-

1. ROLE

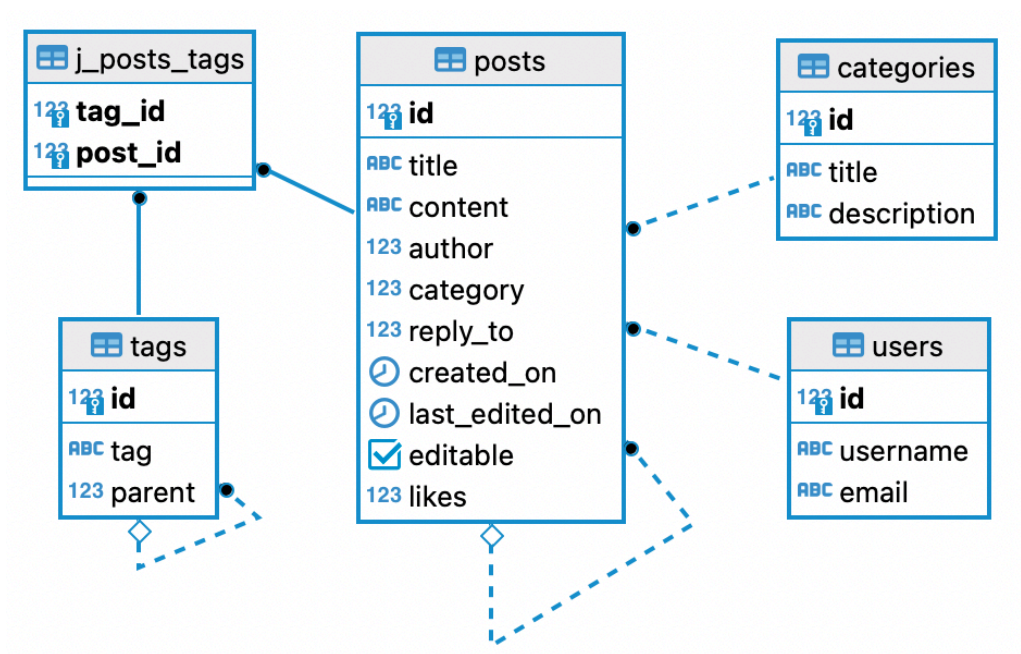
- Hiển thị ROLE đang được sử dụng: `SELECT current_user; SELECT current_role;`
- Tạo ROLE: `CREATE ROLE admin WITH LOGIN PASSWORD '123';`
- Hiển thị các ROLE hiện có: `\du`
- Tạo một database tên hr: `CREATE DATABASE hrdb WITH OWNER admin;`
- Kết nối vào database hr: `\c hrdb;`
- Chuyển sang role admin: `SET ROLE TO admin;`
- Kiểm tra role hiện tại: `SELECT current_role;`
- Cấp quyền cho role: `ALTER ROLE role_name WITH CREATEDB;`
- Loại bỏ quyền của role: `ALTER ROLE role_name WITH NOCREATEDB;`

2. SCHEMA

- Kiểm tra schema hiện tại: `SHOW SEARCH_PATH`
- Xem các schema hiện có: `\dn`
- Tạo một schema được cấp quyền cho admin : `CREATE SCHEMA schema_name AUTHORIZATION admin;`
- Set schema 'example' là schema mặc định: `SET SEARCH_PATH = schema_name;`
- Thay đổi schema của database:
`ALTER DATABASE db_name SET search_path TO schema_name;`
- Thiết lập role mặc định của schema

`ALTER ROLE role_name SET search_path TO schema_name;`

`\i forum_db.sql` → chạy file.sql trong command line



Editable: T, F -> boolean

3. LIMIT và OFFSET clause

```
SELECT * FROM categories ORDER BY id LIMIT 2;
```

```
SELECT * FROM categories ORDER BY id OFFSET 1 LIMIT 2;
```

LIMIT còn được dùng trong việc tạo 1 bảng có cấu trúc với 1 bảng đã có

```
CREATE TABLE new_categories AS select * from categories LIMIT 0;
```

4. Hàm tổng hợp: `COUNT()`, `AVG()`, `MIN()`, `MAX()`, `SUM()`

1.1 Hiện thị tổng likes của tác giả có id =1.

```
SELECT SUM(likes) FROM posts WHERE author=1;
```

1.2 Có tất cả bao nhiêu dòng dữ liệu trong bảng posts?

1.3 Giá trị likes lớn nhất và nhỏ nhất?

5. Hàm xử lý chuỗi: **LENGTH()**, **UPPER()**, **LOWER()**, **TRIM()**, **CONCAT()**, **SUBSTRING()**

```
SELECT LENGTH('Hello PostgreSQL');
SELECT UPPER('POSTgreSQL'), LOWER('POSTgreSQL');
SELECT TRIM(' PostgreSQL ');
SELECT CONCAT('Hello', ' ', 'World'), 'Hello' || ' ' || 'World';
SELECT SUBSTRING('PostgreSQL Database' FROM 1 FOR 10);
```

6. **GROUP BY, HAVING**

```
SELECT column1, aggregate_function (column2)
    FROM table_name
    WHERE condition
    GROUP BY column1
    HAVING condition;
```

```
SELECT DISTINCT category FROM posts;
SELECT category FROM posts GROUP BY category;
```

Ở bảng posts, mỗi category có bao nhiêu dòng dữ liệu?

```
SELECT category, COUNT(*) AS cat_count FROM posts GROUP BY category;
SELECT category, COUNT(*) AS cat_count FROM posts GROUP BY 1;
```

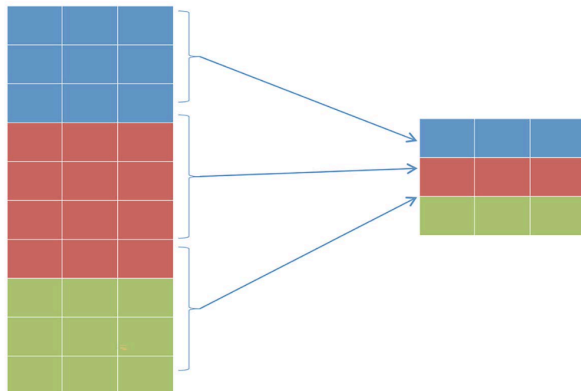
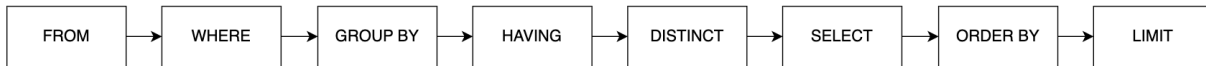
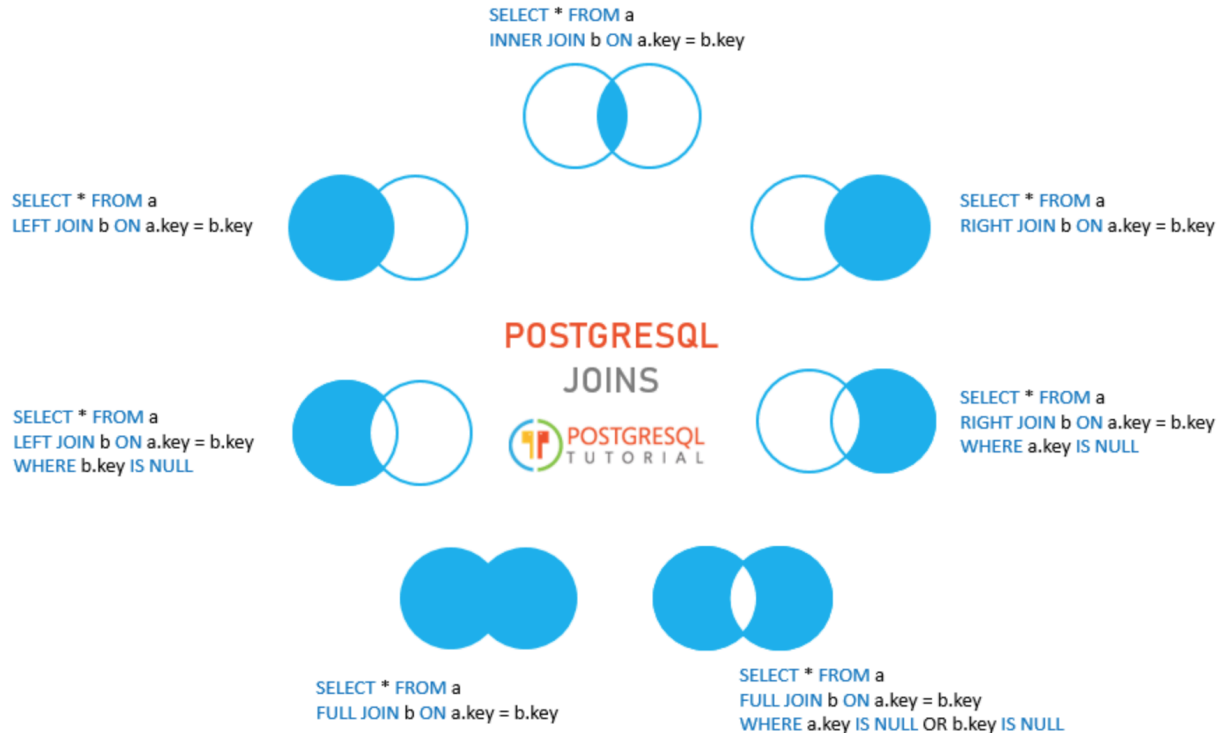


Figure 5.6: Group by aggregation

```
SELECT category, COUNT(*) AS cat_count FROM posts
    GROUP BY 1
    HAVING COUNT(*) > 1;
```



7. JOINS



Nguồn: <https://www.postgresqltutorial.com/>

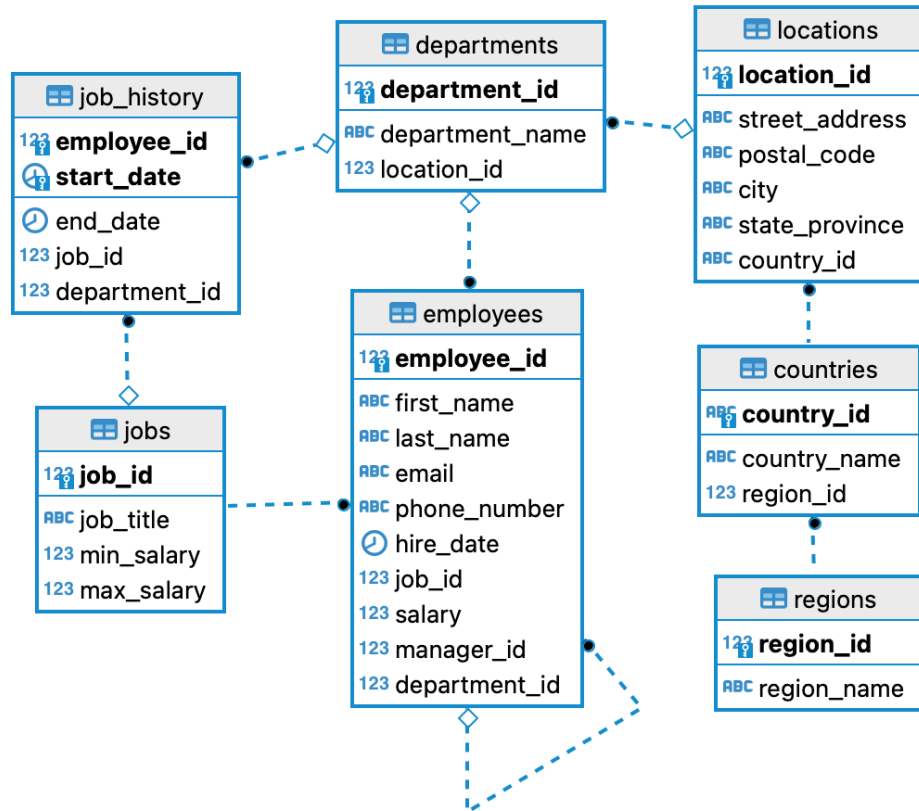
Sử dụng JOIN viết lại các câu truy vấn sau:

- SELECT c.id, c.title, p.id, p.category, p.title FROM categories c, posts p;
- SELECT c.id, c.title, p.id, p.category, p.title FROM categories c, posts p
WHERE c.id=p.category;
- SELECT c.id,c.title,p.id,p.category,p.title FROM categories c,posts p
WHERE c.id=p.category AND c.title='Database';
- SELECT * FROM categories c
WHERE c.id NOT IN (SELECT category FROM posts);

Bài tập:

1. Tạo ROLE 'hadmin'

2. Tạo database 'hr' với owner là 'hradmin'
3. Trong db 'hr', tạo một schema 'hrschema' AUTHORIZATION hradmin
4. Trong schema 'hrschema', tạo các bảng theo mô hình thực thể dưới đây



5. INSERT dữ liệu vào các bảng trên từ hr_data.sql
6. Thực hiện các câu truy vấn sau:
 - a. Hiển thị 3 ký tự đầu tiên của first_name dưới dạng chữ hoa
 - b. Hiển thị first_name của tất cả nhân viên sau khi xóa tất cả khoảng trắng đầu và cuối.
 - c. Hiển thị first_name, last_name và độ dài của name(first_name+last_name)
 - d. Giả sử lương trong bảng employees là lương theo năm, hiển thị first_name, last_name, lương theo tháng của từng nhân viên (làm tròn 2 chữ số thập phân)
 - e. Hiển thị họ tên, mức lương của những nhân viên có mức lương từ \$10,000 đến \$15,000.
 - f. Hiển thị họ tên, department ID của những nhân viên thuộc phòng ban có ID 3 hoặc 10, kết quả sắp xếp theo department ID với thứ tự tăng dần.

- g. Hiển thị họ tên, department ID, salary của những nhân viên thuộc phòng ban có ID 3 hoặc 10, và mức lương không nằm trong khoảng từ \$10,000 đến \$15,000
- h. Hiển thị tên của những nhân viên có chứa chữ cái 'c' và 'e' trong firstname
- i. Hiển thị last_name của những nhân viên có số lượng ký tự của lastname bằng 6.
- j. Hiển thị last_name của nhân viên có ký tự 'e' nằm ở vị trí thứ 3 trong lastname.
- k. Hiển thị name (first_name, last_name), salary và 15% salary cho tất cả nhân viên.
- l. Tổng số tiền lương phải trả cho nhân viên là bao nhiêu?
- m. Mức lương tối đa, tối thiểu, mức lương trung bình và số lượng nhân viên của công ty là bao nhiêu? (làm tròn sau dấu , 2 chữ số thập phân)
- n. Hiển thị các job_id và job_title khác nhau hiện có trong bảng employees, sắp xếp theo job_id?
- o. Mức lương tối đa của nhân viên ở vị trí Programmer?
- p. Chênh lệch giữa mức lương tối đa và mức lương tối thiểu của tất cả nhân viên là bao nhiêu?
- q. Hiển thị id, first_name, last_name của tất cả manager
- r. Hiển thị manager ID và mức lương thấp nhất của nhân viên chịu sự quản lý của manager ID tương ứng.
- s. Hiển thị department ID, tên department và tổng số lương phải trả tương ứng của từng phòng ban, chỉ hiển thị những phòng ban có tổng lương lớn hơn 30000 và sắp xếp theo department_id
- t. Hiển thị name, job_title, salary của những nhân viên không phải làm ở vị trí Programmer hoặc Shipping Clerk, và không có mức lương là \$4,500, \$10,000, hoặc \$15,000
- u. Mức lương trung bình của các phòng ban có trên 5 nhân viên?
- v. Hiển thị job title và mức lương trung bình tương ứng
- w. Hiển thị tên manager, tên phòng ban và thành phố tương ứng
- x. Hiển thị job title, tên employee và sự khác biệt giữa lương của từng nhân viên với mức lương thấp nhất trong công ty và chỉ hiển thị 3 kết quả có mức lương chênh lệch nhiều nhất.

