

# SELECT TOP

 Cover



## ▼ CÔNG DỤNG

- Được sử dụng để lấy ra một số lượng **hạn chế** bản ghi hoặc số **phần trăm** nhất định từ một bảng.
- Nó tương tự như lệnh `LIMIT` trong MySQL và `ROWNUM` trong Oracle

## ▼ CÚ PHÁP

```
SELECT TOP number|percent column1, column2, ...  
FROM table_name  
WHERE condition;
```

- `TOP number|percent` : Chỉ định số lượng bản ghi hoặc phần trăm bản ghi muốn lấy.
- `column1, column2, ...` : Danh sách các cột cần truy vấn.
- `table_name` : Tên bảng chứa dữ liệu.
- `condition` : Điều kiện lọc (tùy chọn).

## ▼ VÍ DỤ

1. Lấy 10 bản ghi đầu tiên từ bảng `employees` .

```
SELECT TOP 10 *  
FROM employees;
```

2. Lấy 5 bản ghi đầu tiên từ bảng `employees` , chỉ với các cột `first_name` , `last_name` , và `salary` .

```
SELECT TOP 5 first_name, last_name, salary  
FROM employees;
```

3. Lấy 20% số bản ghi đầu tiên từ bảng `employees` .

```
SELECT TOP 20 PERCENT *  
FROM employees;
```

4. Lấy 10 bản ghi đầu tiên từ bảng `employees` với điều kiện `department_id` bằng 10.

```
SELECT TOP 10 first_name, last_name, salary
FROM employees
WHERE department_id = 10;
```

5. Lấy 5 nhân viên có mức lương cao nhất từ bảng `employees`.

```
SELECT TOP 5 first_name, last_name, salary
FROM employees
ORDER BY salary DESC;
```

## ▼ SỬ DỤNG TOP VỚI CÁC HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU KHÁC

**MySQL:** Sử dụng `LIMIT` thay cho `TOP`.

```
SELECT first_name, last_name, salary
FROM employees
ORDER BY salary DESC
LIMIT 5;
```

**PostgreSQL:** Cũng sử dụng `LIMIT` thay cho `TOP`.

```
SELECT first_name, last_name, salary
FROM employees
ORDER BY salary DESC
LIMIT 5;
```

**Oracle:** Sử dụng `ROWNUM` hoặc `FETCH FIRST`.

```
SELECT first_name, last_name, salary
FROM employees
WHERE ROWNUM <= 5
ORDER BY salary DESC;
```

```
SELECT first_name, last_name, salary
FROM employees
```

```
ORDER BY salary DESC
FETCH FIRST 5 ROWS ONLY;
```

## ▼ LƯU Ý

### Lưu ý:

- Mệnh đề **TOP** không được hỗ trợ bởi tất cả các hệ thống quản trị cơ sở dữ liệu SQL. Ví dụ: MySQL sử dụng lệnh **LIMIT**, Oracle sử dụng **ROWNUM**.
- Khi sử dụng **TOP** cùng với **ORDER BY**, hãy đảm bảo rằng cột sắp xếp được lập chỉ mục để truy vấn có hiệu suất tốt.
- Sử dụng **TOP** một cách cẩn thận, vì nó có thể ảnh hưởng đến hiệu suất truy vấn, đặc biệt là khi lấy một lượng lớn dữ liệu.

### Ngoài ra, TOP còn có một số cách sử dụng khác:

- Lấy một số bản ghi ngẫu nhiên: Sử dụng **ORDER BY RAND()** trước **TOP**.
- Phân trang kết quả: Sử dụng **TOP** cùng với biến để lấy các trang khác nhau của kết quả truy vấn.

Giả sử chúng ta có bảng **employees** với dữ liệu sau:

employee_id	first_name	last_name	department_id	salary
1	John	Doe	10	60000
2	Jane	Smith	20	75000
3	Larry	Page	10	65000
4	Sergey	Brin	30	80000
5	Alan	Turing	20	70000
6	Ada	Lovelace	30	90000
7	Charles	Babbage	10	72000
8	Grace	Hopper	20	80000
9	Margaret	Hamilton	30	85000
10	Elon	Musk	40	95000

Giả sử bảng **employees** có 10 bản ghi. 20% của 10 bản ghi là 2 bản ghi

## ▼ SQL Server

Trong SQL Server, bạn có thể sử dụng **SELECT TOP 20 PERCENT**:

```
SELECT TOP 20 PERCENT *
```

```
FROM employees;
```

Kết quả:

employee_id	first_name	last_name	department_id	salary
1	John	Doe	10	60000
2	Jane	Smith	20	75000

## ▼ MySQL

Trong MySQL, bạn có thể sử dụng **LIMIT** sau khi tính toán số lượng bản ghi cần lấy:

### 1. Tính tổng số bản ghi trong bảng:

```
SELECT COUNT(*) FROM employees;
```

Tổng số bản ghi là 10.

### 2. Sử dụng **LIMIT** để chọn 20% số bản ghi đầu tiên:

```
SELECT * FROM employees  
LIMIT 2;
```

Kết quả:

employee_id	first_name	last_name	department_id	salary
1	John	Doe	10	60000
2	Jane	Smith	20	75000

## ▼ PostgreSQL

Trong PostgreSQL, sử dụng **LIMIT** với số lượng tính toán:

### 1. Tính tổng số bản ghi trong bảng:

```
SELECT COUNT(*) FROM employees;
```

Tổng số bản ghi là 10.

### 2. Sử dụng **LIMIT** để chọn 20% số bản ghi đầu tiên:

```
SELECT * FROM employees  
LIMIT (SELECT ROUND(COUNT(*) * 0.2) FROM employees);
```

Kết quả:

employee_id	first_name	last_name	department_id	salary
1	John	Doe	10	60000
2	Jane	Smith	20	75000

#### ▼ Oracle

Trong Oracle, sử dụng `ROWNUM` với truy vấn lồng nhau:

##### 1. Tính tổng số bản ghi trong bảng:

```
SELECT COUNT(*) FROM employees;
```

Tổng số bản ghi là 10.

##### 1. Sử dụng `ROWNUM` để chọn 20% số bản ghi đầu tiên:

```
SELECT *  
FROM (  
    SELECT *, ROWNUM AS rnum  
    FROM employees  
    WHERE ROWNUM <= (SELECT ROUND(COUNT(*) * 0.2) FROM employ  
);
```

Kết quả:

employee_id	first_name	last_name	department_id	salary
1	John	Doe	10	60000
2	Jane	Smith	20	75000