

# Advance query

**Giảng viên :Cao Le Thanh**

## Mục tiêu bài học

- ❖ Nắm được cú pháp các câu lệnh join
- ❖ Hiểu được ý nghĩa của câu lệnh nối kết và khóa
- ❖ Sử dụng thành thạo các câu lệnh join trong MySQL



# Nội dung

- ❖ Nối kết và khóa
- ❖ Câu lệnh inner join
- ❖ Câu lệnh left join
- ❖ Câu lệnh right join



# NỐI KẾT VÀ KHÓA

- ❖ Đôi khi chúng ta phải lấy dữ liệu từ hai bảng cùng một lúc, chúng ta thực hiện một kết nối.
- ❖ Các bảng trong **CSDL** có thể quan hệ ràng buộc với nhau thông qua các khoá. Một **khoá chính** (primary key) là một cột mà trong đó mỗi giá trị của hàng phải là duy nhất. Mục đích của khoá là kết nối dữ liệu lại với nhau, từ nhiều bảng khác nhau mà không gây trùng lặp dữ liệu giữa các bảng.

# PHÉP JOIN TRONG MySQL

❖ Phép **JOIN** là phép kết nối dữ liệu từ hai hay nhiều bảng lại với nhau. Khi bạn cần truy vấn lấy các cột dữ liệu từ hai hay nhiều bảng khác nhau để trả về trong cùng một kết quả, bạn cần dùng **JOIN**. Trong MySQL cung cấp các loại **JOIN** là **INNER JOIN**, **LEFT JOIN**, **RIGHT JOIN**, **OUTER JOIN**, **CROSS JOIN**.

# LỆNH INNER JOIN TRONG MYSQL

- ❖ Lệnh trả về kết quả là các bản ghi mà trường được JOIN ở hai bảng khớp nhau, các bản ghi chỉ xuất hiện ở một trong hai bảng sẽ không xuất hiện ở trong kết quả. Để truy xuất dữ liệu theo các điều kiện nào đó, một mệnh đề **WHERE** có thể được thêm vào câu lệnh **SELECT**

## Lưu ý:

Mệnh đề MySQL INNER JOIN là một phần tùy chọn của câu lệnh SELECT. Nó xuất hiện ngay sau mệnh đề FROM của câu lệnh SELECT.

# LỆNH INNER JOIN TRONG MYSQL

## ❖ Cú pháp :

```
SELECT cột_1, cột_2, cột_3  
FROM bảng_1  
INNER JOIN bảng_2  
ON bảng_1.khoá_chính = bảng_2.khoá_ngoại
```

## ❖ Ví dụ:

```
SELECT Employees.Name, Orders.Product  
FROM Employees  
INNER JOIN Orders  
ON Employees.Employee_ID = Orders.Employee_ID
```

# LỆNH LEFT JOIN TRONG MYSQL

❖ Lấy về các **records** có mặt trong **bảng bên trái( bảng 1)** và cả các **records** phù hợp với điều kiện khớp nối trong **bảng bên phải ( bảng 2)**. Với các bảng ghi không phù hợp xẩy hiện trong **bảng bên phải** sẽ được điền giá trị **NULL**

❖ Cú pháp

```
SELECT cột_1, cột_2, cột_3  
FROM bảng_1  
LEFT JOIN bảng_2  
ON bảng_1.khoá_chính = bảng_2.khoá_ngoại
```



# LỆNH RIGHT JOIN TRONG MYSQL

❖ Lấy về các **records** có mặt trong **bảng bên phải (bảng 2)** và cả các **records** phù hợp với điều kiện khớp nối trong **bảng bên trái (bảng 1)**. Với các bảng ghi không phù hợp xuất hiện trong **bảng bên trái** sẽ được điền giá trị **NULL**

❖ Cú pháp

```
SELECT cột_1, cột_2, cột_3  
FROM bảng_1  
RIGHT JOIN bảng_2  
ON bảng_1.khoá_chính = bảng_2.khoá_ngoại
```

# Tổng kết

- ❖ Phép JOIN khá quan trọng trong MySQL
- ❖ Học viên cần nắm vững cú pháp của các câu lệnh **join**

