BÁO CÁO TASK 2

# Các khái niệm cơ bản:

* **Cơ sở dữ liệu (Database)** là một tập hợp các dữ liệu có mối quan hệ với nhau chứa thông tin về một tổ chức nào đó (như một trường học, một công ty, một nhà máy,…) được lưu trữ trên các thiết bị thứ cấp (như băng từ, đĩa từ) để đáp ứng nhu cầu khai thác thông tin của nhiều người sử dụng với nhiều mục đích khác nhau.
* **Hệ quản trị cơ sở dữ liệu (Database Management System – DBMS)** là một tập các chương trình cho phép người sử dụng định nghĩa, tạo lập và duy trì cơ sở dữ liệu.
* **Cơ sở dữ liệu quan hệ** là một kiểu cơ sở dữ liệu mà tại đó dữ liệu có liên quan tới các thông tin khác trên cơ sở dữ liệu. Chúng được dùng để lưu trữ và tổ chức dữ liệu với các tham chiếu giữa hai hoặc nhiều nguồn.
* **Bảng dữ liệu** là thành phần chính trong cơ sở dữ liệu quan hệ. Trong đó:
* ***Cột/Trường (Field)***: thể hiện thuộc tính của bảng dữ liệu ví dụ như tên, địa chỉ…
* ***Dòng (Row)***: là dòng dữ liệu gồm các dữ liệu liên quan với nhau.
* ***Ô (Cell)***: giao giữa dòng và cột, đây là nơi chứa dữ liệu.
* ***Khóa chính (Primary Key)***: là một hoặc nhiều trường gộp lại được dùng để định nghĩa bảng ghi. Khóa chính có 2 thuộc tính là không được rỗng và không được trùng.
* ***Khóa ngoại (Foreign Key)***: là trường ở bảng này nhưng có trường tương ứng làm khóa chính ở bảng khác để tạo mối quan hệ giữa hai bảng.
* ***Mối quan hệ (Relationship)***: tạo ra mối liên kết giữa hai bảng với mục đích xác định mối liên quan giữa các trường dữ liệu của hai bảng.

Trong cơ sở dữ liệu quan hệ, mối quan hệ thường thể hiện ở 3 dạng sau:

* ***Quan hệ 1-1***: mỗi bảng chỉ có một bản ghi tương ứng. Ví dụ quan hệ vợ – chồng, quan hệ thông tin cơ bản – thông tin chi tiết…
* ***Quan hệ 1-n***: quan hệ phổ biến nhất trong cơ sở dữ liệu. Trong hệ này 1 bảng ghi ở bảng này có nhiều bảng ghi tương ứng ở bảng khác.
* ***Quan hệ n-n***: một bảng ghi trong bảng này tương ứng với nhiều bảng ghi trong bảng kia và ngược lại.
* **Thực thể (Entity)** không có một định nghĩa hình thức, thực thể là đối tượng mà ta cần quan tâm trong thế giới khách quan.
* **Kiểu thực thể (Entity type)** là tập hợp các đối tượng cùng loại, hình thành kiểu thực thể, nói cách khác kiểu thực thể chính là những thực thể cùng được mô tả bằng những đặc trưng giống nhau.
* **Thuộc tính (attribute)** được hiểu là dữ liệu dùng để mô tả một đặc trưng của thực thể.

Ví dụ: thuộc tính tên, tuổi, giới tính … là tính chất đặc trưng của thực thể Sinh viên.

# Các câu lệnh cơ bản của ngôn ngữ SQL

## Lệnh tạo bảng:

**CREATE TABLE** **<tên\_bảng>**

**(**  <tên cột 1> <kiểu dữ liệu> [null | not null],

<tên cột 2> <kiểu dữ liệu> [null | not null],

…

<tên cột n> <kiểu dữ liệu> [null | not null],

[constraint <tên ràng buộc khóa chính> primary key (khóa chính)],

[constraint <tên ràng buộc khóa ngoại> foreign key (khóa ngọai) references <bảng>]**)**

## Lệnh sửa cấu trúc bảng:

Sử dụng lệnh **ALTER TABLE** để thay đổi cấu trúc bảng.

## Lệnh thêm dữ liệu:

**INSERT INTO** TênBảng[(Danh sách cột)]

**VALUES**(Danh sách giá trị)

## Lệnh xóa dữ liệu:

**DELETE FROM** TênBảng

[**WHERE** <Điều kiện xoá>]

## Lệnh sửa dữ liệu:

**UPDATE** <Tên Bảng>

**SET** <Tên cột>=<giá trị>

[**WHERE** <Điều kiện sửa>]

## Truy vấn dữ liệu:

**SELECT** [\* | **DISTINCT**] <danh\_sách\_cột>

**FROM** <danh\_sách\_bảng>

[**WHERE** <biểu\_thức\_điều\_kiện>]

[**GROUP BY** <danh\_sách\_tên\_nhóm>]

[**HAVING** <biểu\_thức\_điều\_kiện\_nhóm>]

[**ORDER BY** {tên\_cột\_thứ\_tự | số\_thứ\_tự\_cột | biểu thức] } [**ASC** | **DESC**]

Select: Xác định nội dung của các cột cần đưa ra kết quả.

From: Xác định các bảng cần lấy thông tin ra.

Where: Xác định các bảng ghi thỏa yêu cầu chọn lọc để đưa ra kết quả.

## Kết hợp các kết quả với toán tử UNION

UNION cho phép kết hợp hai hoặc nhiều bộ kết quả từ nhiều bảng với nhau.

**SELECT** <danh sách cột>

**UNION** [**DISTINCT** | **ALL**]

**SELECT** <danh sách cột>

**UNION** [**DISTINCT** | **ALL**]

….

## Truy vấn INTERSECT

Intersect kết hợp tập kết quả của hai hoặc nhiều truy vấn và trả về các bản ghi xuất hiện trong cả hai truy vấn.

**QUERY 1**

**INTERSECT**

**QUERY 2**

## Truy vấn EXCEPT

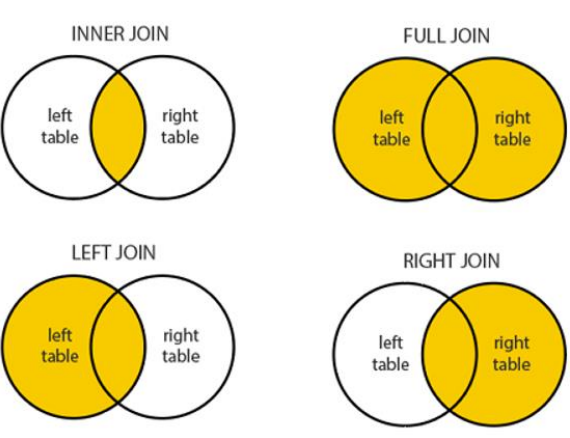
Except so sánh tập hợp kết quả của hai truy vấn và trả về các bản ghi của truy vấn thứ nhất không xuất hiện trong các bản ghi của truy vấn thứ hai.

**QUERY 1**

**EXCEPT**

**QUERY 2**

## Truy vấn với phép kết ngoài (outer join)



### Inner Join

**SELECT** <danh sách cột>

**FROM** <tên bảng>

**INNER** **JOIN** <tên bảng quan hệ>

**ON** <điều kiện kết nối>;

### Left outer Join

**SELECT** <danh sách cột>

**FROM** <tên bảng>

**LEFT [OUTER]** **JOIN** <tên bảng quan hệ>

**ON** <điều kiện kết nối>;

### Right outer Join

**SELECT** <danh sách cột>

**FROM** <tên bảng>

**RIGHT [OUTER]** **JOIN** <tên bảng quan hệ>

**ON** <điều kiện kết nối>;

### Full outer Join

**SELECT** <danh sách cột>

**FROM** <tên bảng>

**FULL OUTER** **JOIN** <tên bảng quan hệ>

**ON** <điều kiện kết nối>;