

Các mô hình dữ liệu

# Nội dung

- Các mô hình dữ liệu
  - Mô hình mạng
  - Mô hình phân cấp
  - Mô hình quan hệ
  - Mô hình thực thể - liên kết
  - Mô hình hướng đối tượng
- Thiết kế CSDL
- Mô hình Thực thể - liên kết

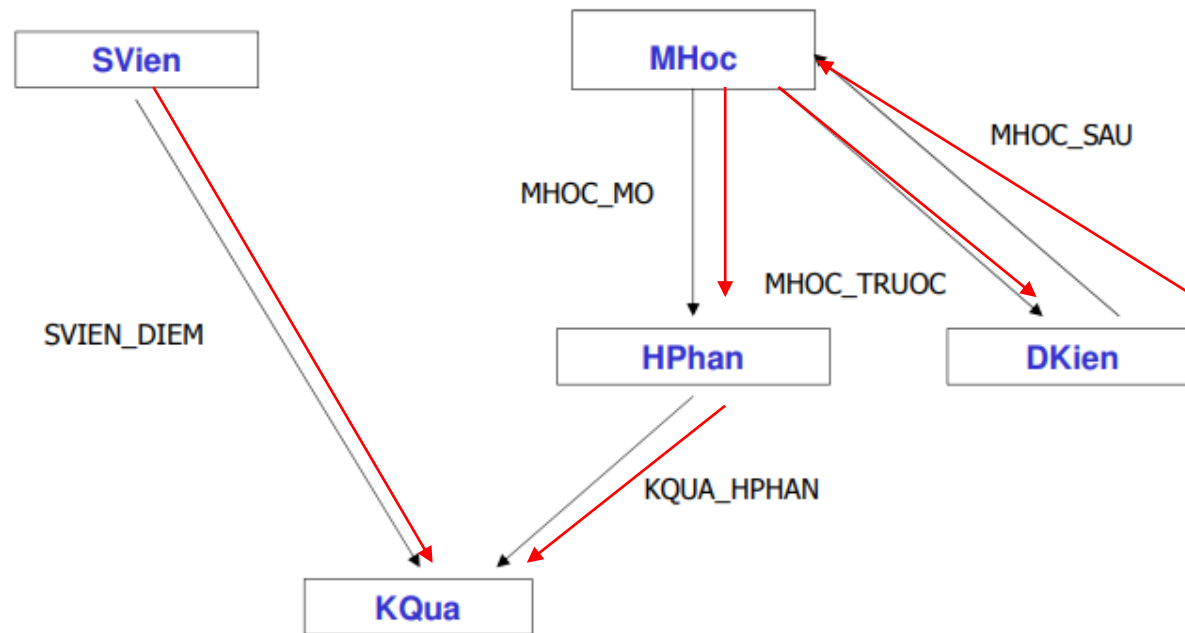
# Các mô hình dữ liệu

- Vấn đề?
  - Đặc điểm của các mô hình dữ liệu?
  - Sự khác nhau giữa các mô hình dữ liệu?
  - Các mô hình dữ liệu phổ biến ngày nay

# Các mô hình dữ liệu

- **Mô hình mạng (Network data model)**

- Biểu diễn bởi một đồ thị có hướng
- Quan hệ cha/con
- Các khái niệm bản ghi
- Móc nối



- Kiểu liên kết đi từ mẫu tin chủ (owner record) đến mẫu tin thành viên (member)

- **Ưu điểm : Đơn giản**

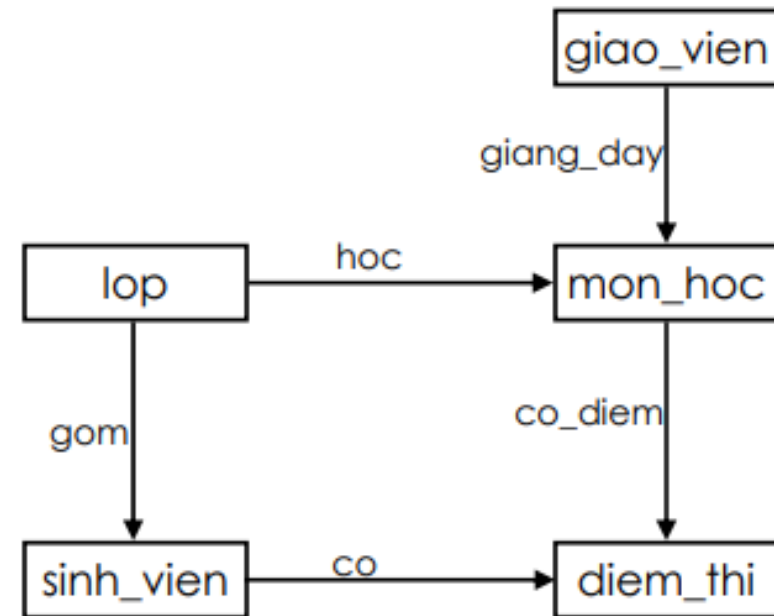
- Có thể biểu diễn các ngữ nghĩa đa dạng với kiểu bản ghi và kiểu móc nối
- Truy vấn thông qua phép duyệt đồ thị (navigation)

- **Nhược điểm**

- Số lượng các con trỏ lớn
- Hạn chế trong biểu diễn ngữ nghĩa của các móc nối giữa các bản ghi

# Các mô hình dữ liệu

- **Mô hình phân cấp (Hierarchical Data Model)**
  - Mô hình là một cây (Tree), trong đó mỗi nút của cây biểu diễn một thực thể, giữa nút con và nút cha được liên hệ với nhau theo một mối quan hệ xác định.
- Mô hình dữ liệu phân cấp sử dụng các khái niệm sau:
  - - *Loại mẫu tin*: giống khái niệm mẫu tin trc mô hình dữ liệu mạng.
  - - *Loại mối liên hệ*: Kiểu liên hệ là phân cấp
  - - Các đối tượng ngang hàng quan hệ 1-1
  - Cha/con 1-N



# Các mô hình dữ liệu

Mức 1:

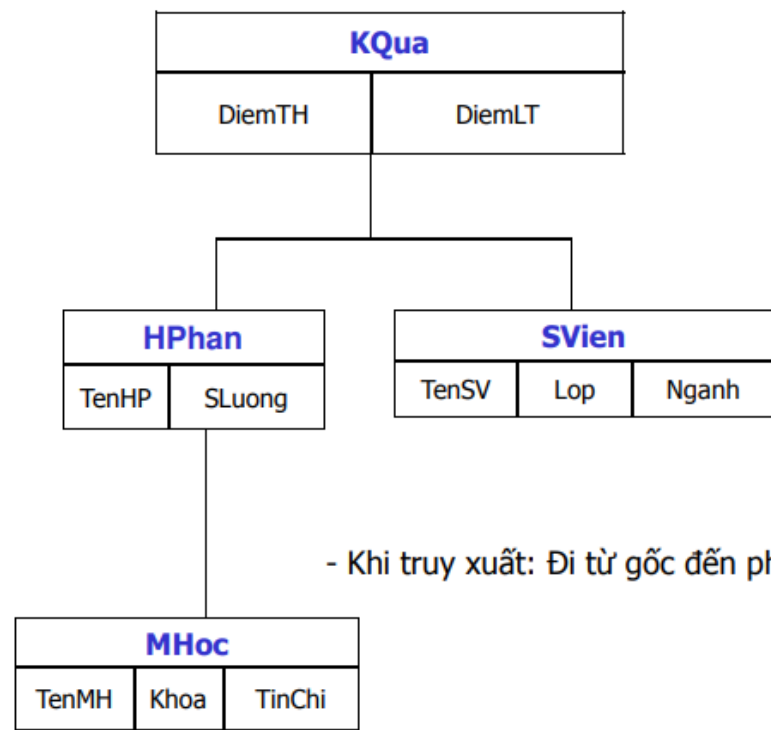
- **Mô hình phân cấp (Hierarchical Data Model)**

Mức 2:

- **Ưu điểm**
  - Dễ xây dựng và thao tác
  - Tương thích với các lĩnh vực tổ chức phân cấp
  - Ví dụ: tổ chức nhân sự trong các đơn vị, ...)
  - Ngôn ngữ thao tác đơn giản (duyệt cây)

Mức 3:

- **Nhược điểm**
  - Dư thừa dữ liệu và dữ liệu không nhất quán
  - Hạn chế trong biểu diễn ngữ nghĩa của các mối nối giữa các bản ghi (chỉ cho phép quan hệ 1-n)



- Khi truy xuất: Đi từ gốc đến phần tử cần xét

# Các mô hình dữ liệu

Student		Major		Class	
PK	<u>StudentID</u>	PK	<u>MajorName</u>	PK	<u>ClassName</u>
	GivenName Surname GraduationYear Hometown			PK	<u>ClassTimeSlot</u>
				PK	<u>ClassLocation</u>
Minor					
PK	<u>MinorName</u>				

- **Mô hình dữ liệu quan hệ (relational data model)**
- Vào năm 1970 [Codd, 1970]
- Nền tảng cơ bản của nó là khái niệm lý thuyết tập hợp trên các quan hệ, tức là tập của các bộ giá trị (Value Tuples).
  - Trong mô hình dữ liệu này những khái niệm sẽ được sử dụng bao gồm thuộc tính (Attribute), quan hệ (Relation), lược đồ quan hệ (Relation Schema), bộ (Tuple), khóa (Key).
- Chúng ta sẽ học mô hình trong môn học này.

Suppliers			
SupplID	SuppName	SuppStreet	SuppCity
085	Lomar Inc.	183 S. Main	Ma
086	Elgin Co.	1900 Ave. B	Bo
087	Drake Co.	206 Merl	Sa

Products			
ProdID	SupplID	ProdName	ProdCategory
93877	085	Alpha tool	En
93878	086	Beta tool	De
93879	087	Delta tool	Ch

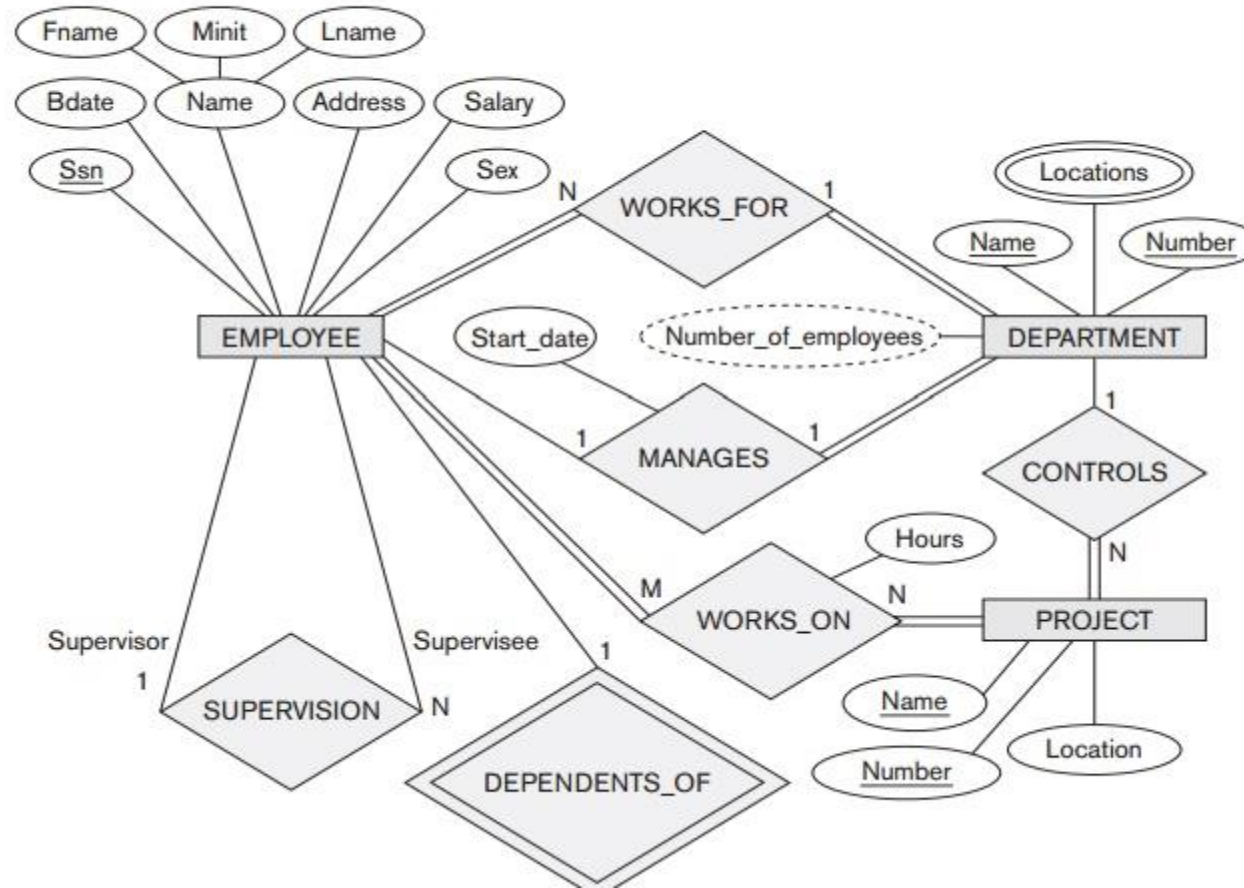
# Các mô hình dữ liệu

- Mô hình dữ liệu thực thể - liên kết (Entity-Relational data model)
- Sự ra đời
  - Xuất phát từ nhu cầu mô hình hoá ngữ nghĩa dữ liệu và phát triển phần mềm z đề xuất 1975 [Chen, 1976] [Chen, 2002]
  - Biểu diễn: bằng sơ đồ thực thể - liên kết
  - Các khái niệm cơ bản
    - Thực thể: một đối tượng trong thế giới thực
    - Thuộc tính: một đặc tính của một tập thực thể
    - Khoá: xác định sự duy nhất của 1 thực thể
    - Liên kết: mối liên hệ có nghĩa giữa nhiều thực thể
    - Mỗi liên kết có thể có các thuộc tính
      - 1-1, 1-n, n-m, đệ quy



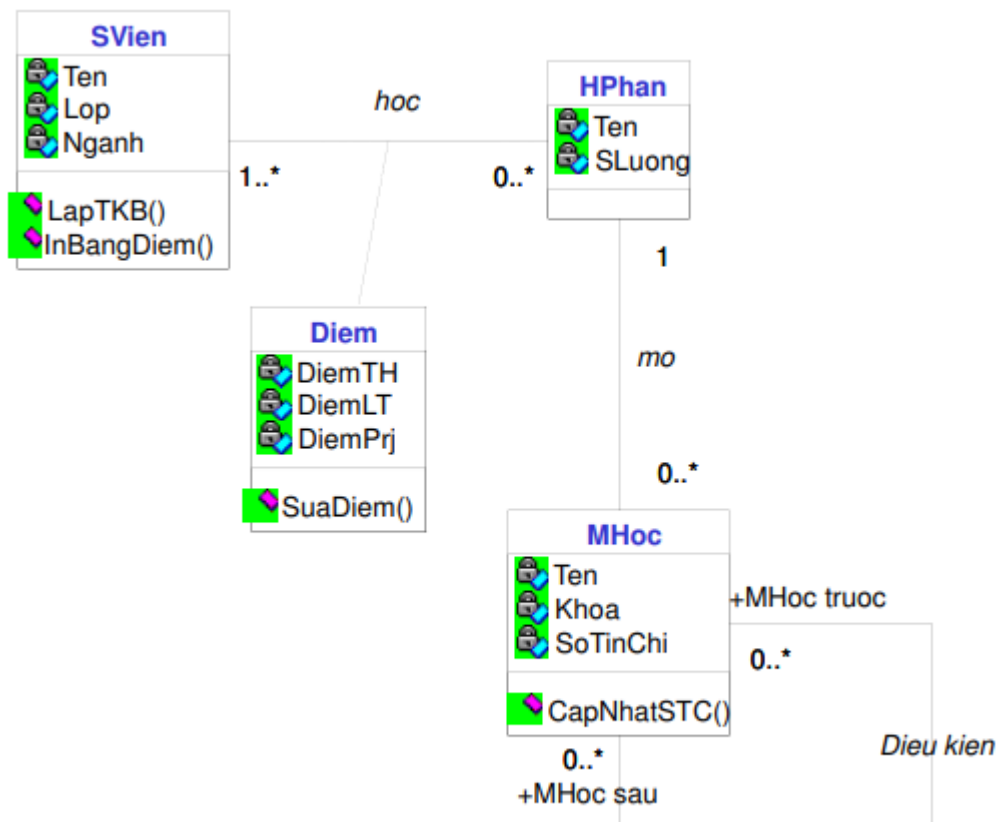
**Figure 9.1**

The ER conceptual schema diagram for the COMPANY database.



# Các mô hình dữ liệu

- Mô hình dữ liệu hướng đối tượng (Object-oriented data model)
- Sự ra đời
  - Khoảng đầu những năm 90
  - Biểu diễn: sơ đồ lớp
  - Các khái niệm cơ bản
    - Đối tượng: một đối tượng trong thế giới thực, được xác định bởi một định danh duy nhất
    - Thuộc tính: biểu diễn một đặc tính của đối tượng
    - Phương thức : thao tác được thực hiện trên đối tượng.
      - Tất cả các truy nhập vào thuộc tính của đối tượng đều phải được thực hiện thông qua các phương thức này.
    - Lớp: một cách thức để khai báo một tập các đối tượng có chung một tập thuộc tính và phương thức



```

public class CacheEntry {
    //private Document[] doc = new Document[10];
    private String [] Results;
    private int frequency;

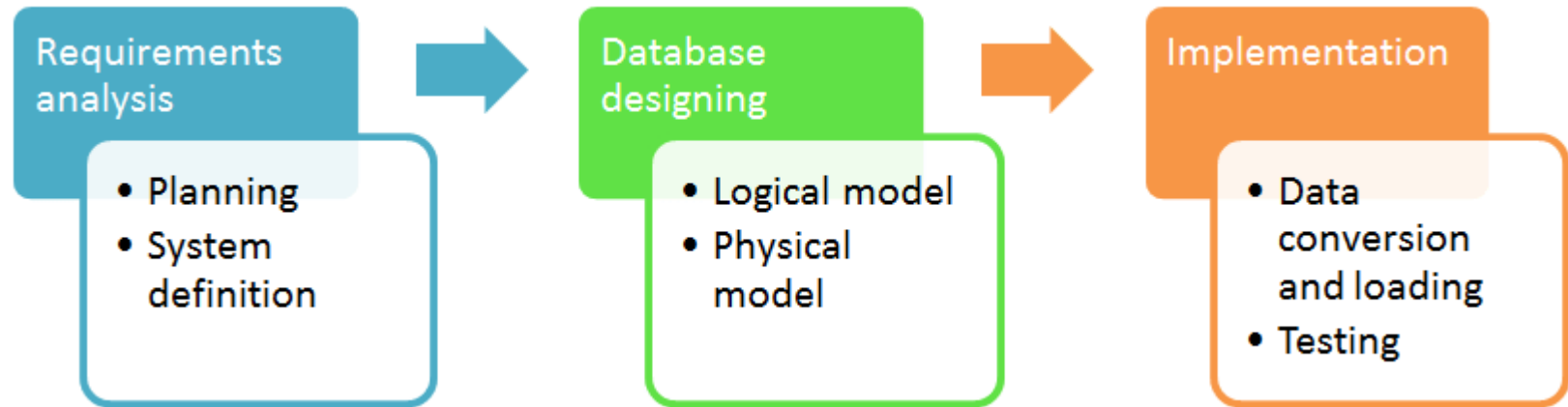
    // default constructor
    public CacheEntry(String [] Results , int frequency)
    {
        this.Results =Results;
        this.frequency = frequency;
    }

    public String[] getResults() {
        return Results;
    }
    public void setResults(String[] Results) {
        this.Results = Results;
    }

    public int getFrequency() {
        return frequency;
    }
    public void setFrequency(int frequency) {
        this.frequency = frequency;
    }
}

```

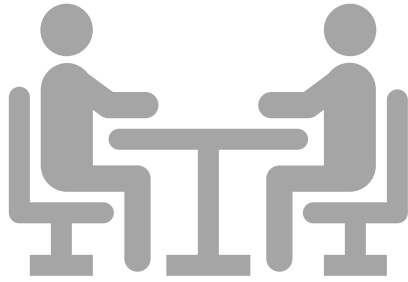
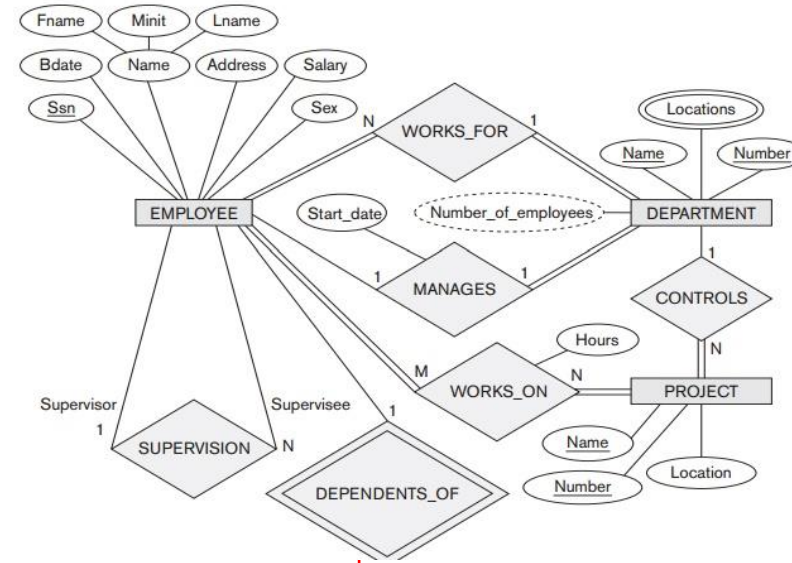
# Thiết kế CSDL



# Mô hình hoá dữ liệu

Figure 9.1

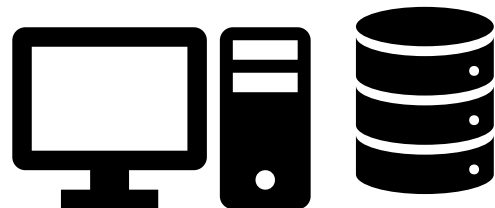
The ER conceptual schema diagram for the COMPANY database.



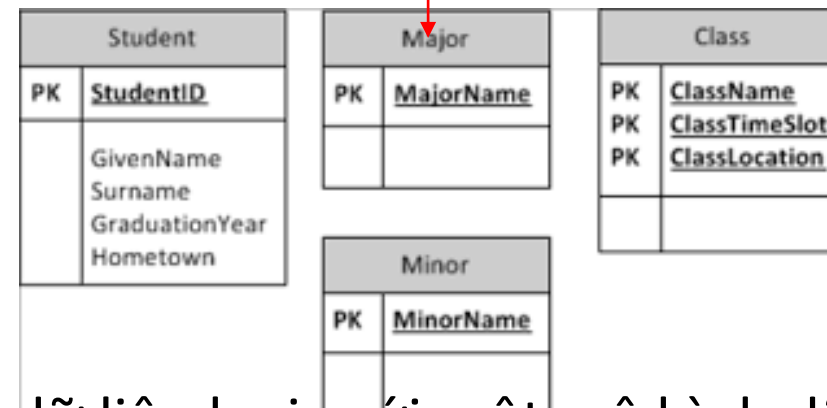
Mô tả công việc

Phân tích

Thiết kế



Triển khai



Mô tả dữ liệu logic với một mô hình dữ liệu (Sơ đồ quan hệ)

# Thiết kế CSDL

- Thu thập và phân tích yêu cầu
  - Các yêu cầu về CSDL
  - Các yêu cầu về chức năng thao tác (Thao tác trên CSDL)
- Thiết kế quan niệm và phân tích chức năng
  - Tạo một sơ đồ quan niệm (cấp cao), ví dụ thực thể liên kết
  - Đặc tả giao tác cấp cao tương ứng với các thao tác trên CDL
- Thiết kế logic
  - Ánh xạ lược đồ quan niệm thành lược đồ logic (mô hình quan hệ)
- Thiết kế và triển khai các ứng dụng.

# Mô hình dữ liệu bằng lược đồ thực thể-liên kết

- Được dùng để thiết kế CSDL ở mức quan niệm
- Biểu diễn trừu tượng cấu trúc của CSDL

- Thích hợp mô hình hoá dữ liệu cho CSDL
- Dễ phân tích với trực quan của con người
- Dễ chuyển đổi sang mô hình quan hệ
- Được biểu diễn bằng sơ đồ thực thể liên kết (entity-relationship diagram ERD)
- Và dựa vào một số định nghĩa chính:

- Thực thể
- Tập Thực thể
- Thuộc tính
- Khoá
- Liên kết
- Tập các liên kết

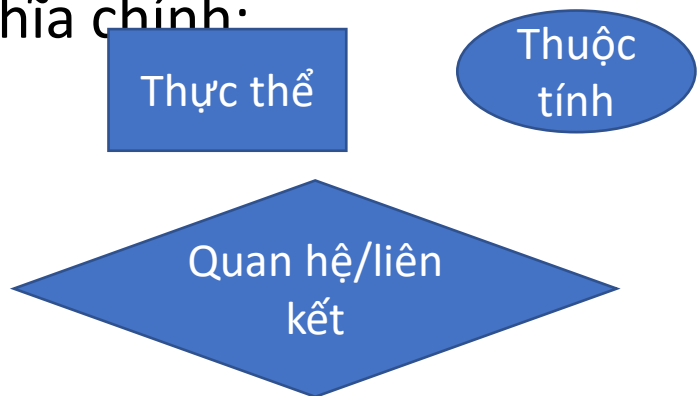
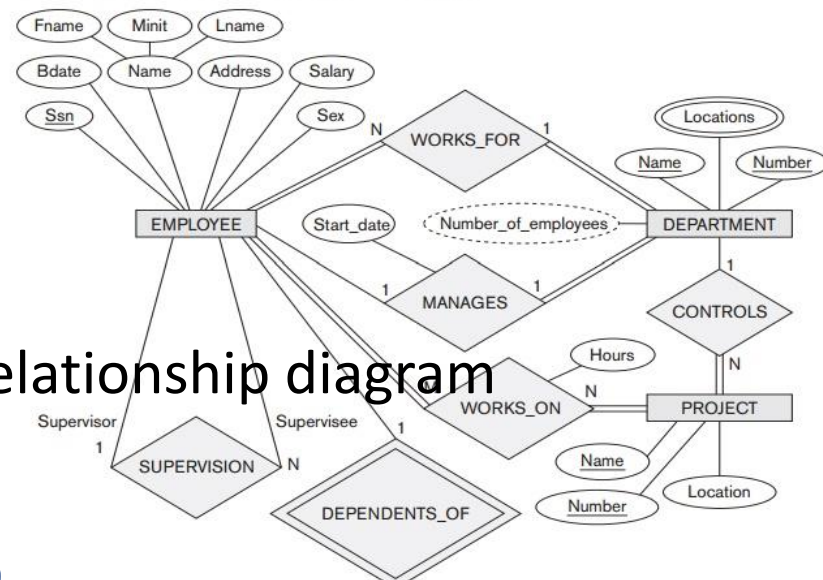


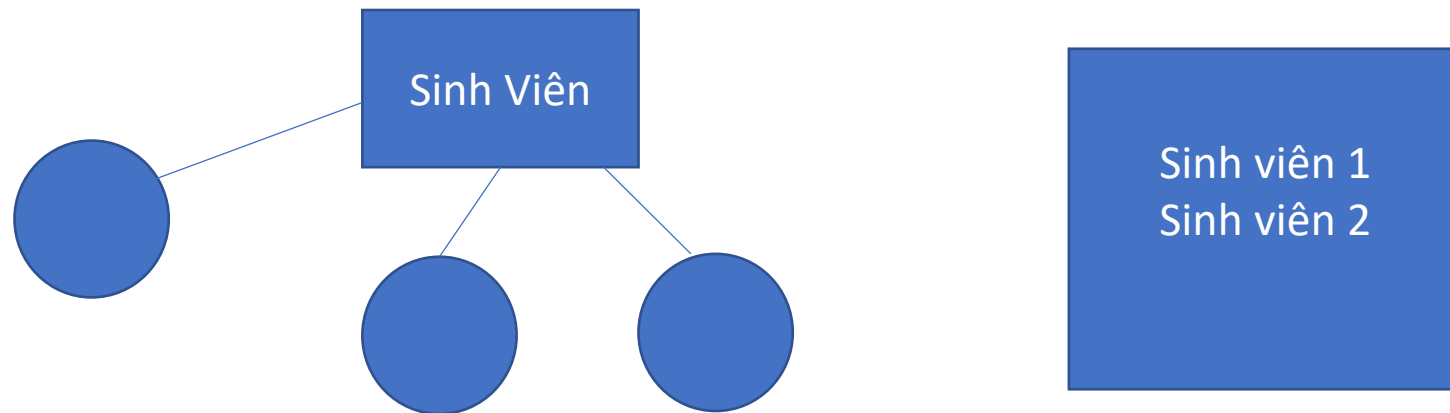
Figure 9.1

The ER conceptual schema diagram for the COMPANY database.



# Thực thể

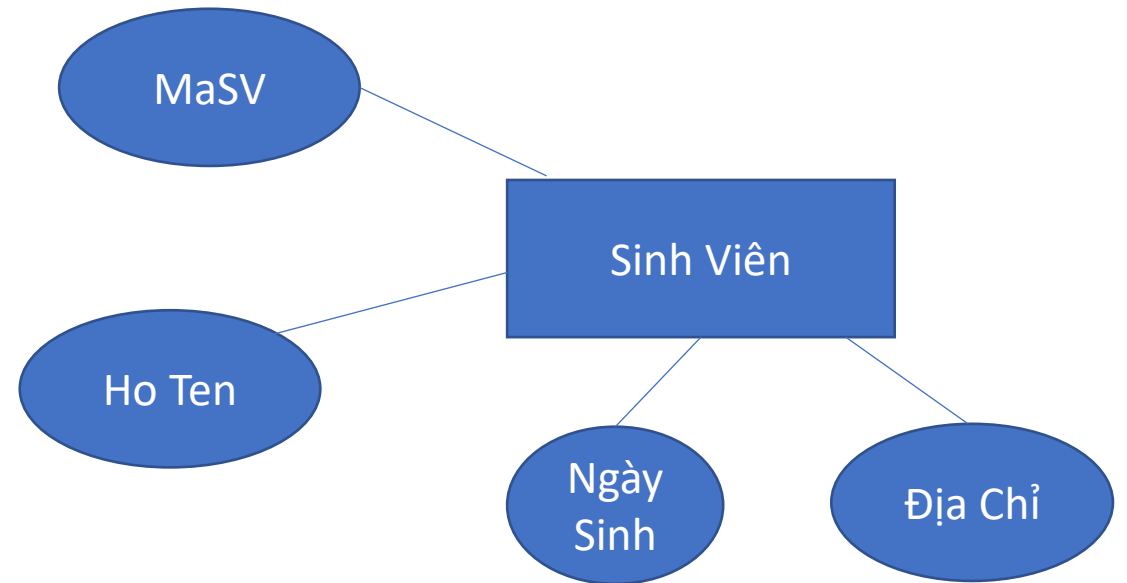
- Thực thể: Được xem như là một đối tượng của thế giới thực
  - Ví dụ: Thực thể Sinh Viên, Môn học
- Tập thực thể: gồm các thực thể có tính chất giống nhau
  - Ví dụ: Một tập hợp các Sinh Viên, hoặc các Môn học



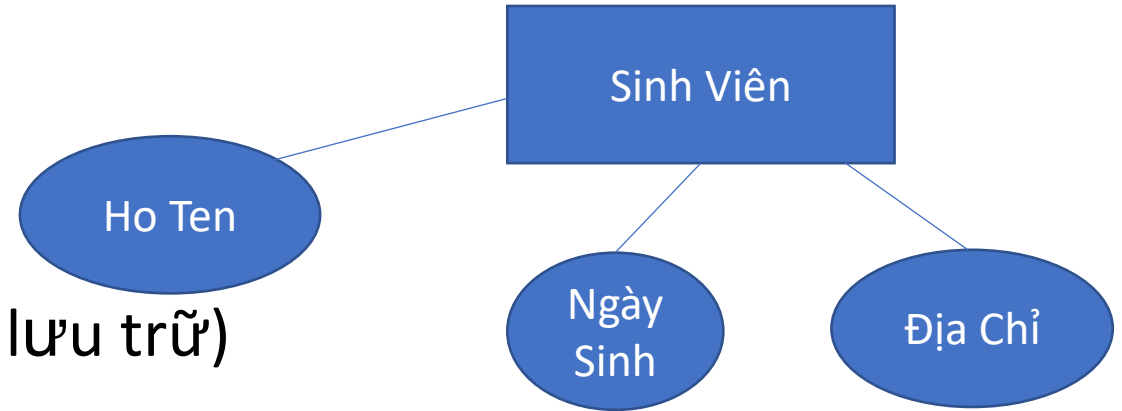


# Thuộc tính

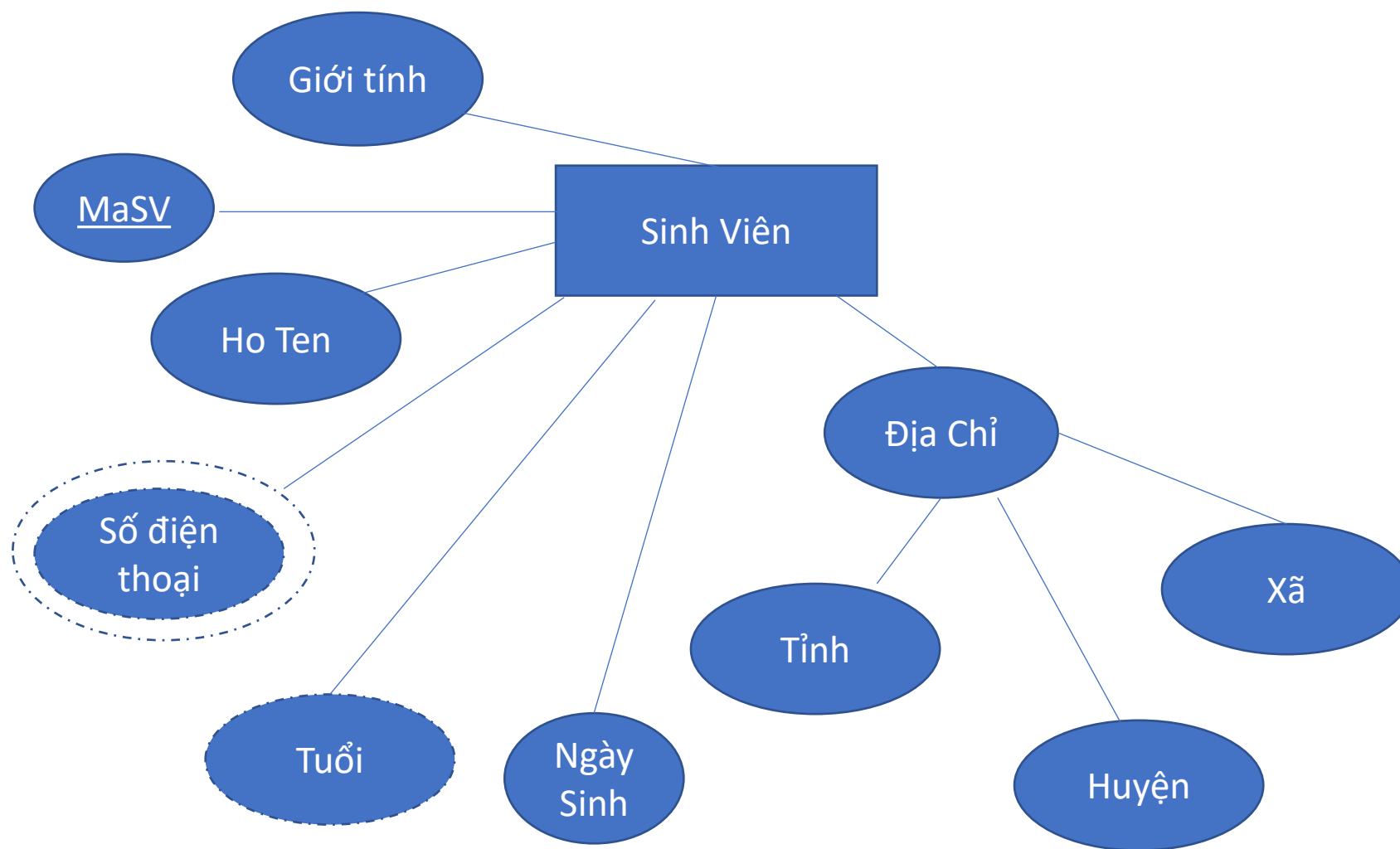
- Là những đặc tính riêng biệt của thực thể
  - Miền giá trị : Tập các giá trị có thể
  - Khoá: xác định sự duy nhất của 1 thực thể
  - Có thể có nhiều Khoá, hoặc dùng nhiều thuộc tính làm Khoá
- Ví dụ: Thực thể Sinh Viên
  - Thuộc tính:
    - Họ Tên
    - Ngày sinh,
    - Địa chỉ
- Kiểu giá trị:
  - Số
  - Ký tự
  - Text



# Thuộc tính

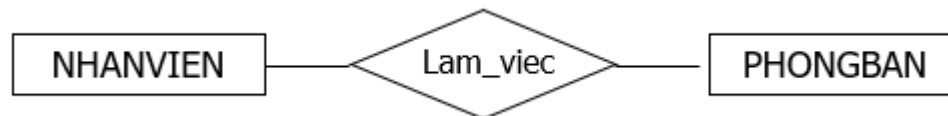
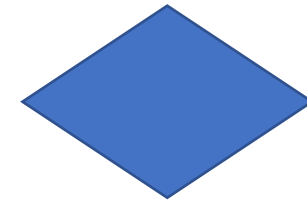


- Thuộc tính đơn – không thể tách nhỏ ra được (khi tách ra, không còn ý nghĩa để lưu trữ)
  - Ví dụ: Giới tính của SV: Nam hay Nữ
  - Tên Sinh Viên.
- Thuộc tính kết hợp – Có thể tách thành nhiều thuộc tính nhỏ hơn
  - Ví dụ: Địa chỉ, có thể tách ra là Tỉnh, Huyện, Xã,...
- Thuộc tính đơn trị – Có giá trị duy nhất cho một thực thể
  - VD: Mã số SV, Căn Cước CD
- Thuộc tính đa trị – Có thể nhiều giá trị khác nhau ở cùng 1 thực thể (vd: số đt, emails)
- Thuộc tính suy diễn – Giá trị của nó được suy ra từ thuộc tính khác
  - Ví dụ, từ Năm Sinh, có thể suy ra được Tuổi



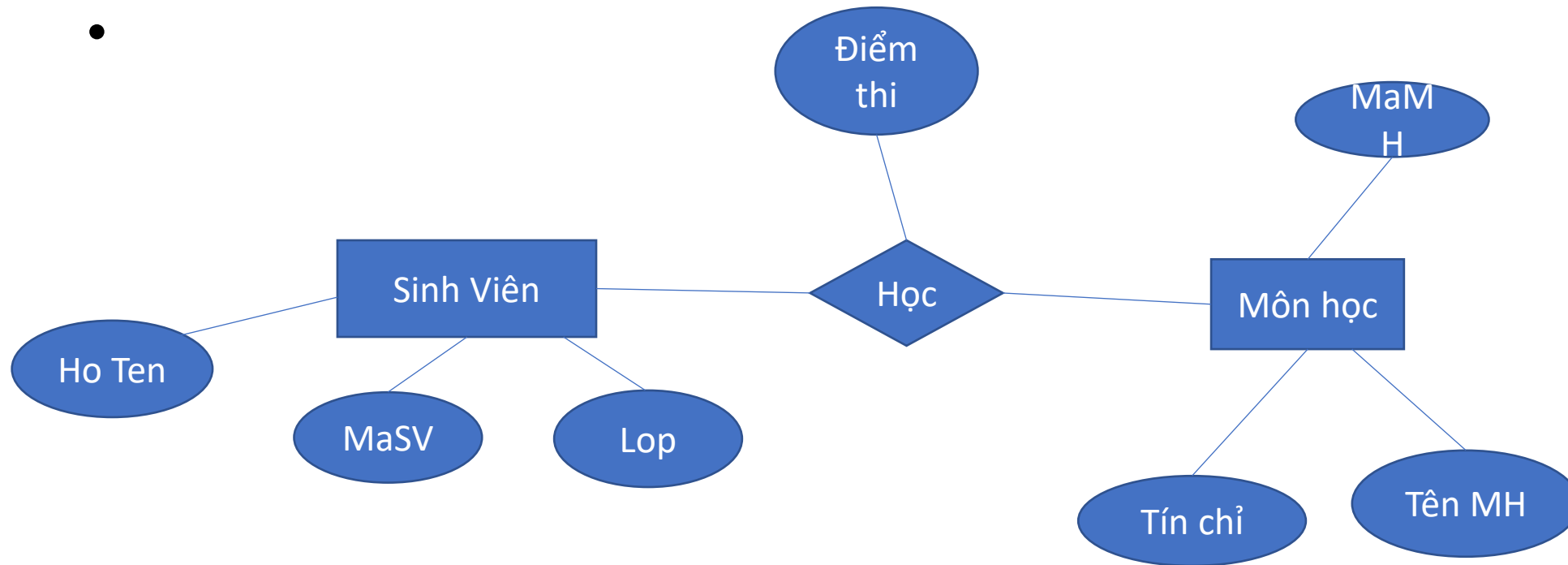
# Liên kết (Quan hệ)

- Liên kết : Là sự liên kết giữa 2 hay nhiều tập thực thể
- Ví dụ giữa tập thực thể NHANVIEN và PHONGBAN có các liên kết
  - Một nhân viên thuộc một phòng ban nào đó



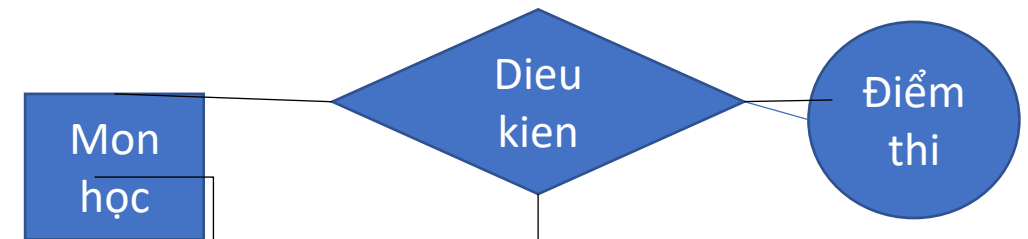
# Liên kết (Quan hệ)

- Liên kết cũng có thuộc tính.
- 

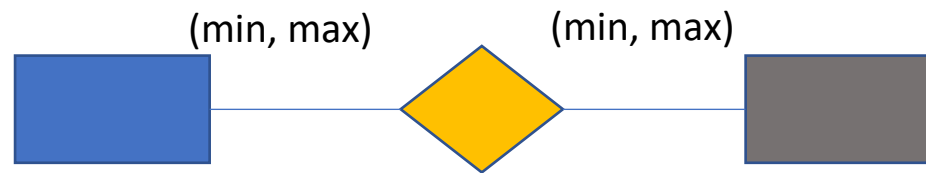


# Các kiểu Liên kết/Quan hệ - Ràng buộc của liên kết (multiplicity)

- 1 -1 : Liên kết giữa 1 thực thể (của một tập thực thể ) với duy nhất một thực thể (của một tập thực thể khác)
- 1-n : Liên kết 1 thực thể (trong tập thực thể) với nhiều thực thể (trong tập thực thể khác)
- n-m : Liên kết 1 thực thể (trong tập thực thể A với nhiều thực thể (trong tập thể B), và một thực thể (trong tập thể B) với nhiều thực thể (trong tập thực thể A).
- Đệ quy: Liên kết với chính thực thể dựa vào điều kiện



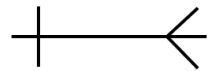
- Trong sơ đồ thiết kế: cũng có thể sẽ ký hiệu các kiểu liên kết dưới dạng:
- (min, max)
- (0,1) Không hoặc 1
- (1,1) duy nhất 1
- (0,n) Không hoặc nhiều
- (1,n) một hoặc nhiều



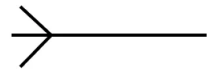
# Relationship Cardinality Notations



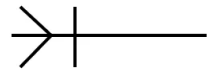
One to One



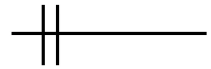
One to Many  
(Mandatory)



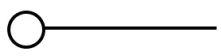
Many



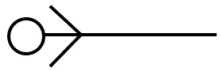
One of More  
(Mandatory)



One & only One  
(Mandatory)



Zero or One  
(Optional)



Zero or Many  
(Optional)

Company

Employee

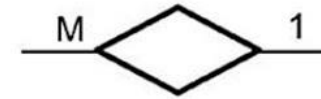
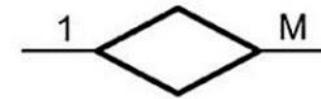
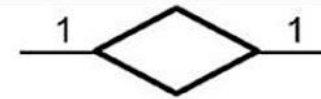
Projects

## Cardinality Relationship

**Cardinality** – The number of entities to which another entity can be associated through a relationship

The diagrams on the right show, in order:

one-to-one  
one-to-many  
many-to-one  
many-to-many



Relationship

Relationship

Relationship

Relationship



# Các bước tiến hành lập ERD

- Xác định các yêu cầu phân tích
- Xác định các đối tượng → thực thể
- Xác định các thuộc tính liên quan
- Xác định miền giá trị (kiểu dữ liệu) cho thuộc tính
- Xác định Khoá, thuộc tính Khoá
- Xác định mối liên kết giữa các thực thể
- Xác định kiểu liên kết cho giữa các thực thể (hoặc (min, max))

- **Ví dụ 1: Thiết kế hệ thống quản lý điểm của SV**

- Bao gồm các đối tượng?
- Thuộc tính?
- Liên kết?
- Vẽ sơ đồ?

- CACOO.com

- <https://www.flaticon.com/>

- **Ví dụ 2: Thiết kế hệ thống CSDL quản lý xe đạp**