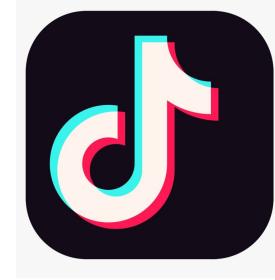


A large red square with a white border, centered on the page. Inside the square, the text "SQL for Beginner" is written in white.

SQL for Beginner



Các ứng dụng cung cấp các thông tin cho mục đích sử dụng



Các ứng dụng lưu trữ dữ liệu

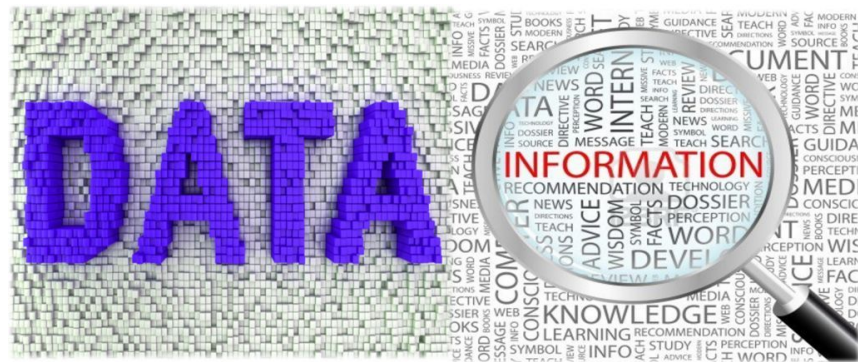
Mục tiêu bài học

- Cơ sở dữ liệu (Database)
- Hệ quản trị cơ sở dữ liệu (DBMS)
- Hệ quản trị cơ sở dữ liệu quan hệ (RDBMS)
- Giới thiệu về SQL Server, T-SQL
- Thiết kế cơ sở dữ liệu

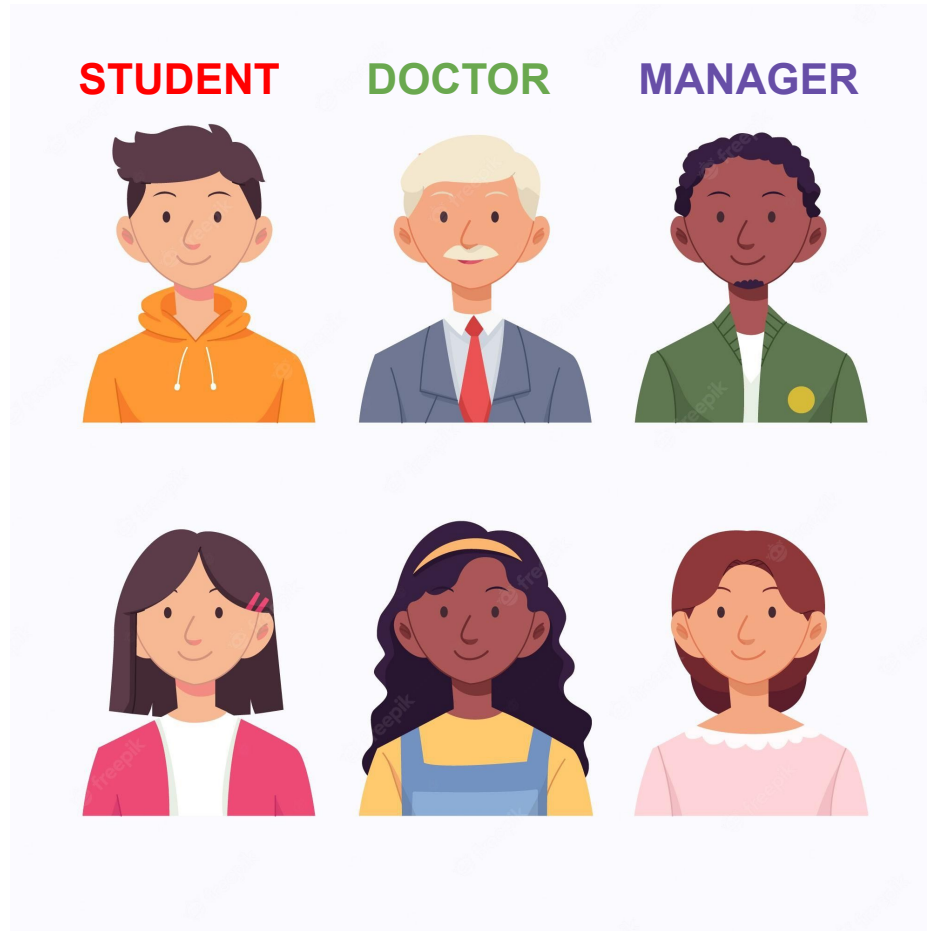


Dữ liệu (Data)?

- ❑ Là các thông tin của đối tượng (ví dụ: con người, đồ vật, sự việc, khái niệm,...) được lưu trữ trên thiết bị (server, máy tính, tệp tin,...)
- ❑ Có thể truy cập để trích xuất ra các thông tin phục vụ nhu cầu khai thác, quản lý.
- ❑ Dữ liệu có thể được miêu tả dưới nhiều dạng/kiểu dữ liệu khác nhau (ví dụ: hình ảnh, chuỗi ký tự, số học, ngày giờ, ký hiệu, âm thanh, tệp, ...)

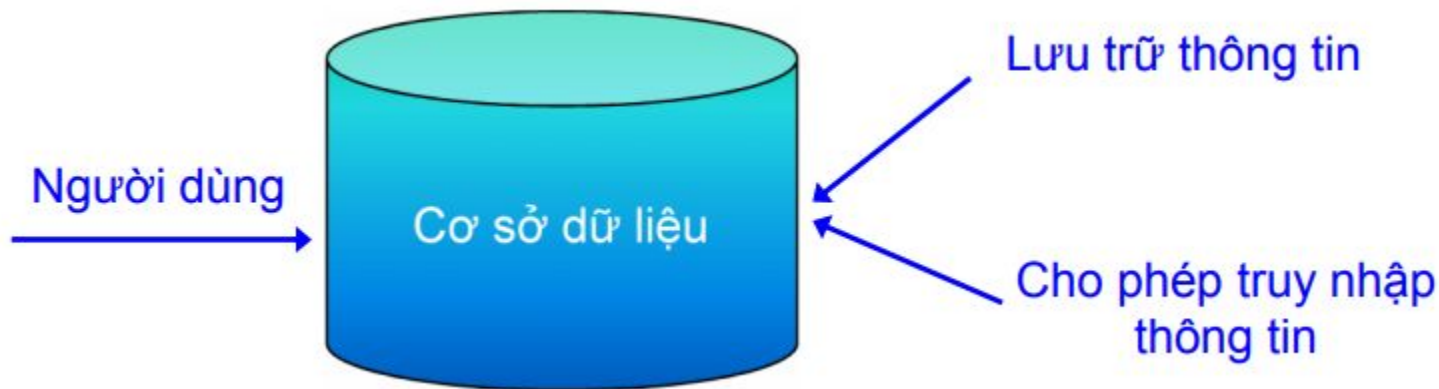


- ❑ Dữ liệu về đối tượng có thể khác nhau, tùy thuộc vào ngữ cảnh và mục đích sử dụng, quản lý.



Cơ sở dữ liệu - CSDL (Database) ?

- ❑ CSDL (Database): Là tập hợp dữ liệu được tổ chức có cấu trúc, có liên quan với nhau và được lưu trữ.
- ❑ CSDL được thiết kế, xây dựng nhằm mục đích cho phép người sử dụng lưu trữ, truy xuất, cập nhật và quản lý dữ liệu.



Cơ sở dữ liệu - CSDL (Database) ?

❏ CSDL được tổ chức có cấu trúc:

- Cấu trúc thành các bản ghi (record), các trường dữ liệu (field)
- Các dữ liệu lưu trữ có mối quan hệ (relational) với nhau

MaHV	HoTen	QueQuan	DienThoai	NgaySinh
HV01	Phan Dinh Phu...	Da Nang	0123456789	1990-01-01
HV02	Nguyen Van Anh	Quang Nam	0123456789	1990-01-01
HV03	Nguyen Van Na...	Ha Noi	0123456789	1990-01-01
HV04	Le Van Nam	HCM	0123456789	1990-01-01
HV05	Nguyen Thi Loan	Da Nang	0123456789	1990-01-01

Record

1

0..n

Relational

Field

MaDiem	MaHV	MaMon	NgayThi	DiemThi
DT01	HV01	MH01	2018-01-01	7
DT02	HV01	MH02	2018-02-01	9
DT03	HV02	MH03	2018-04-01	3
DT04	HV02	MH03	2018-04-15	6
DT05	HV03	MH04	2018-04-20	10
DT06	HV05	MH04	2018-04-20	10

Cơ sở dữ liệu - CSDL (Database) ?



Hệ quản trị CSDL (DBMS) ?

DBMS (Database Management System)

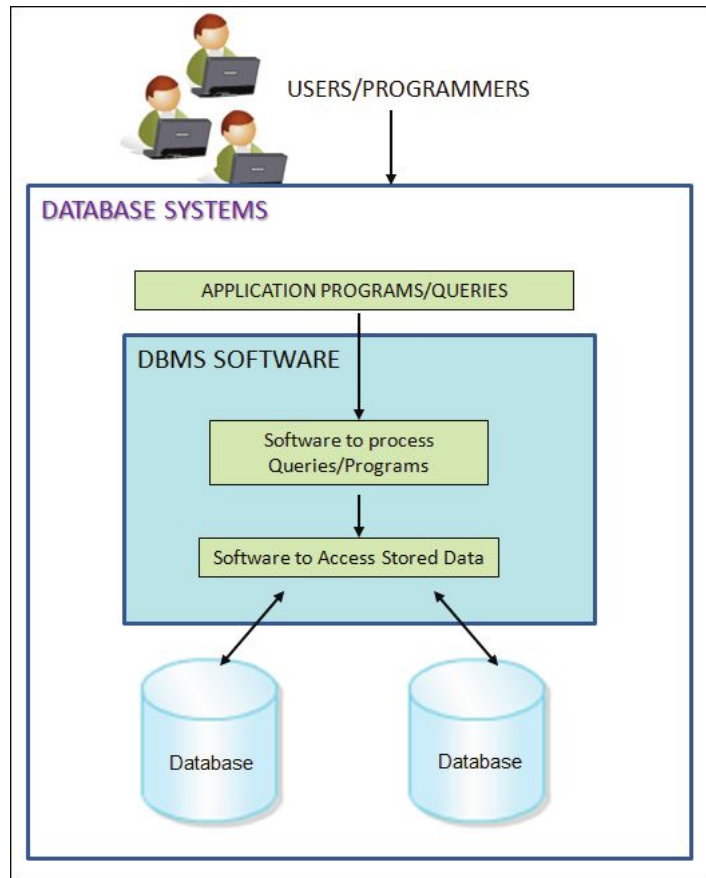
Là tập hợp các chương trình truy cập và thao tác CSDL hay chính là các phần mềm giúp tạo và cung cấp cơ chế lưu trữ, truy xuất theo mô hình cấu trúc CSDL



Hệ quản trị CSDL (DBMS) ?

Lợi ích của DBMS

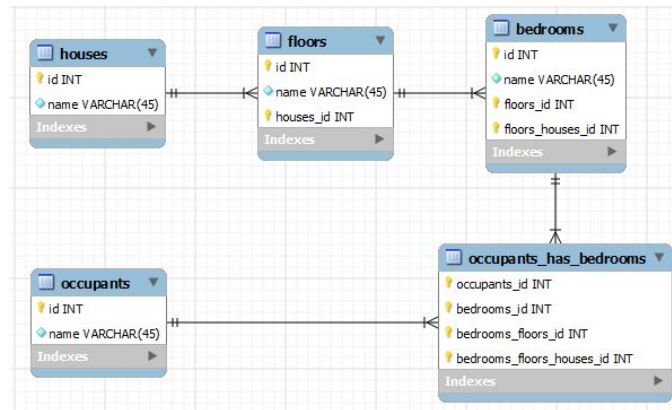
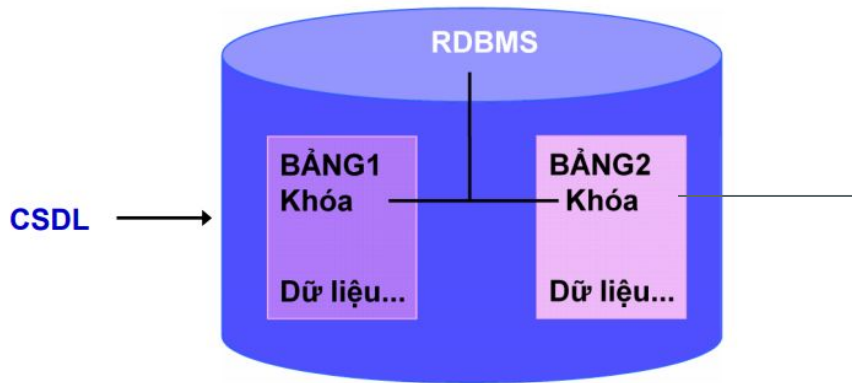
- ❑ Lưu trữ dữ liệu.
- ❑ Định nghĩa dữ liệu
- ❑ Thao tác dữ liệu
- ❑ Bảo mật và toàn vẹn dữ liệu
- ❑ Phục hồi và truy cập đồng thời dữ liệu
- ❑ Cải thiện hiệu suất
- ❑ Kiểm soát truy cập nhiều người dùng
- ❑ Cung cấp ngôn ngữ và giao diện truy cập



Hệ quản trị CSDL quan hệ (RDBMS) ?

RDBMS (Relational Database Management System)

- ❑ Là một loại phổ biến trong DBMS, dựa trên mô hình dữ liệu quan hệ. Trong đó tất cả dữ liệu được tổ chức chặt chẽ dưới dạng các bảng (table) dữ liệu.
- ❑ Tất cả thao tác với database đều diễn ra trên các table, giữa các table có mối liên hệ chặt chẽ thông qua các thuộc tính liên kết (khóa)



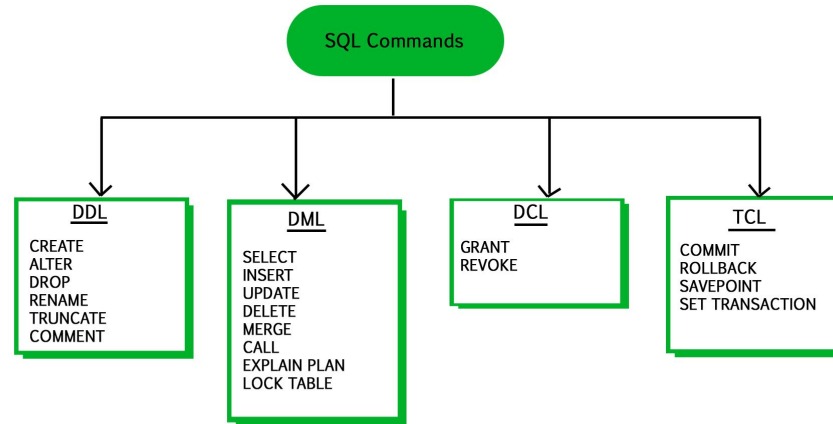
Tổng quan về SQL Server



- Là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) sử dụng câu lệnh Transact SQL (T-SQL) để trao đổi dữ liệu giữa Client và máy cài SQL Server.

Ngôn ngữ T-SQL (Transact SQL)

- Transact SQL (T-SQL) là ngôn ngữ phát triển nâng cao của ngôn ngữ SQL chuẩn. Dùng để giao tiếp giữa ứng dụng và SQL Server.
- T-SQL cung cấp ngôn ngữ định nghĩa dữ liệu (DDL) và ngôn ngữ thao tác dữ liệu (DML) cho phép lập trình trên SQL Server được linh động hơn.



Cài đặt SQL SERVER

<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>



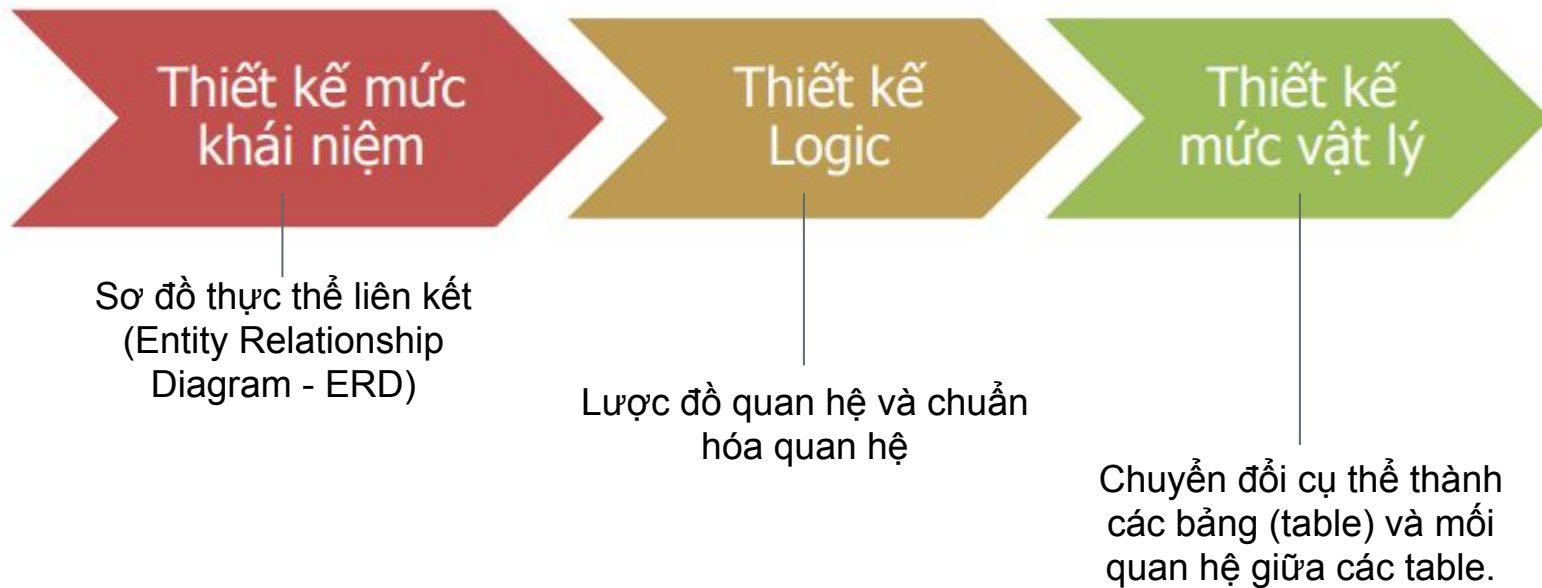
Developer

SQL Server 2019 Developer is a full-featured free edition, licensed for use as a development and test database in a non-production environment.

Download now >

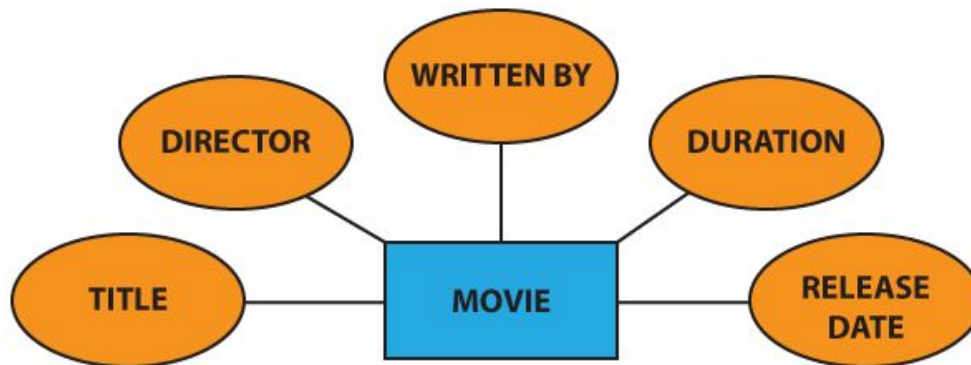
Các bước thiết kế CSDL quan hệ

Thiết kế một CSDL được phân thành các mức khác nhau:



Thực thể và thuộc tính

- ❑ Thực thể (Entity) là một đối tượng, địa điểm, sự vật, sự việc,... trong thế giới thực được lưu trữ thông tin trong Database
- ❑ Mỗi thực thể bao gồm một hoặc nhiều thuộc tính đặc trưng cho thực thể đó.

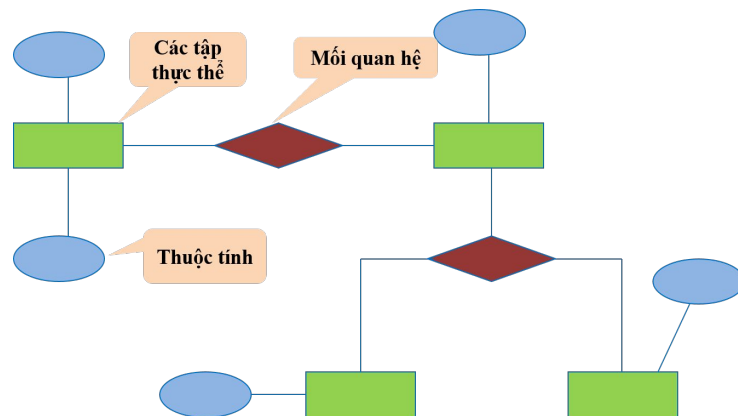


CSDL mức khái niệm


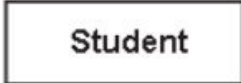








- ❑ Biểu đồ thực thể - quan hệ (Entity-Relationship): là một biểu diễn bằng đồ họa của mô hình E-R, dùng để thể hiện các thành phần khác nhau của mô hình E-R.

Các thành phần cơ bản mức khái niệm:

- ❑ Các thực thể (Entity).
- ❑ Các thuộc tính (Attribute)
- ❑ Các mối quan hệ (Relationship)



Các biểu tượng được sử dụng

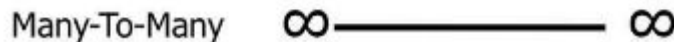
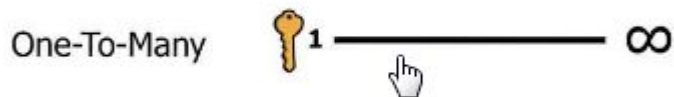
Thành phần	Ký hiệu	Ví dụ
Thực thể		
Thực thể yếu		
Thuộc tính		
Mối quan hệ		
Thuộc tính chính		

Mối quan hệ (Relationship)

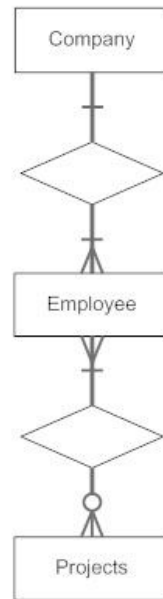
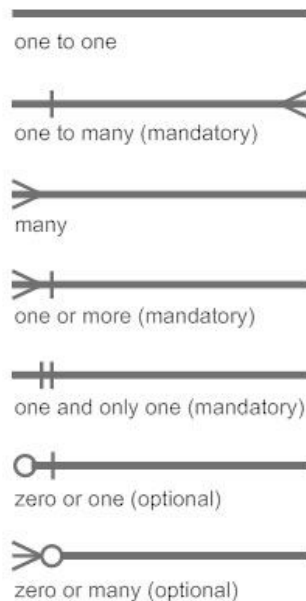
❖ Relationship là mối liên kết giữa các tập thực thể (Entity).

❖ Phân loại:

- ❑ Quan hệ một - một (1 - 1).
- ❑ Quan hệ một - nhiều (1 - n)
- ❑ Quan hệ nhiều - nhiều (n - n)

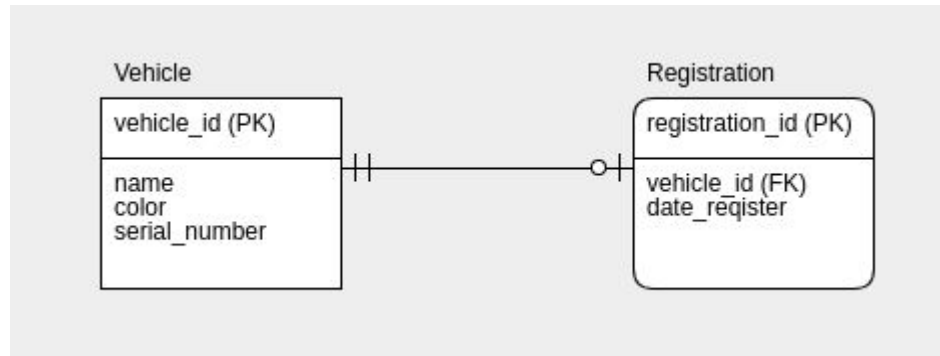


Information Engineering Style



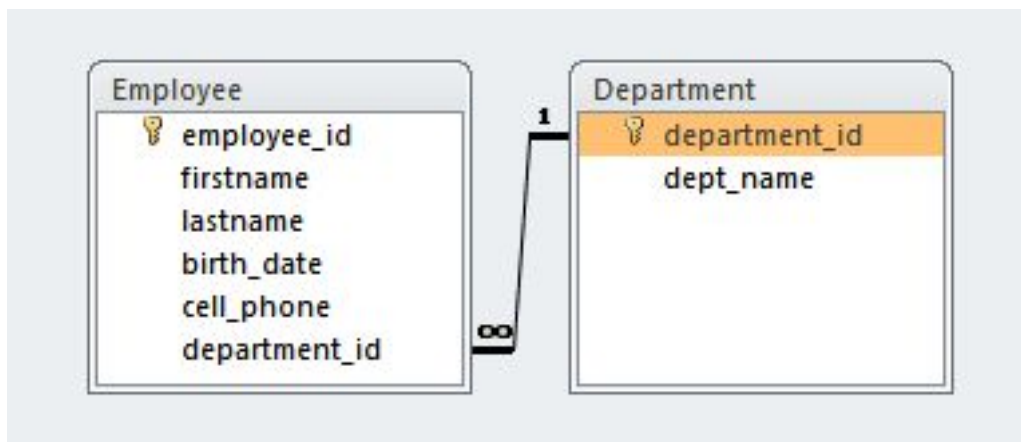
Quan hệ 1 - 1 (one-to-one)

- ❖ Là quan hệ giữa hai tập thực thể trong đó mỗi thực thể của tập này chỉ có thể liên kết với nhiều nhất một thực thể của tập kia, và ngược lại
- ❖ Ví dụ: Mối quan hệ giữa **Vehicle** (Phương tiện) và **Registration** (Giấy đăng ký)



Quan hệ 1 - n (one-to-many)

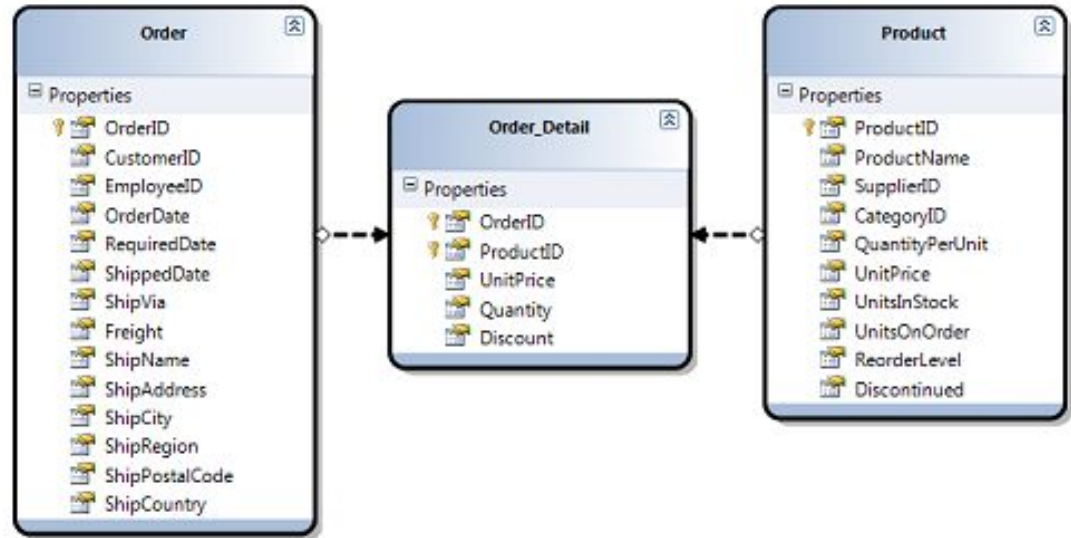
- ❖ Là quan hệ giữa hai tập thực thể trong đó một thực thể của tập này có thể liên kết với nhiều thực thể của tập còn lại.
- ❖ Ví dụ: Mối quan hệ giữa **Employee** (Nhân viên) và **Department** (Phòng ban)



Quan hệ n - n (many-to-many)

- ❖ Là quan hệ giữa hai tập thực thể trong đó một thực thể của tập này có thể liên kết với 0, 1 hoặc nhiều thực thể của tập kia, và ngược lại.
- ❖ Thường quan hệ n - n có thêm phần dữ liệu giao nhau để thêm thông tin cụ thể cho mỗi quan hệ.

- ❖ **Ví dụ:** Mỗi quan hệ giữa **Order** (Đơn hàng) và **Product** (Sản phẩm)



Các bước xây dựng một sơ đồ E-R

- ❖ Bước 1: Thu thập tất cả các dữ liệu cần được lên mô hình
- ❖ Bước 2: Xác định dữ liệu có thể được mô hình hóa như các thực thể trong thế giới thực.
- ❖ Bước 3: Nhận dạng các thuộc tính cho từng thực thể.
- ❖ Bước 4: Sắp xếp các tập hợp thực thể thành các tập hợp thực thể yếu và mạnh
- ❖ Bước 5: Sắp xếp các thuộc tính thực thể như các thuộc tính khóa, thuộc tính..
- ❖ Bước 6: Nhận dạng các quan hệ giữa các thực thể khác nhau.

Chuẩn hóa dữ liệu (Mức Logic)

Khái niệm: Chuẩn hóa dữ liệu là là quá trình loại bỏ các sự dư thừa và các phần phụ thuộc không mong muốn.

Các dạng chuẩn hóa dữ liệu:

- ❖ Chuẩn 1NF
- ❖ Chuẩn 2NF
- ❖ Chuẩn 3NF

Chuẩn hóa dữ liệu

Dạng chuẩn 1NF:

- ❖ Tạo các bảng riêng cho từng nhóm dữ liệu có liên quan
- ❖ Các cột của bảng phải có các giá trị nguyên tử (giá trị đơn)
- ❖ Tất cả các thuộc tính chính phải được nhận dạng

Xem ví dụ: Bảng chi tiết dự án của nhân viên

Emp_no	Project_id	Project_name	Emp_name	Grade	Salary
142	113, 124	BLUE STAR, MAGNUM	John	A	20.000
168	113	BLUE STAR	James	B	15.000
263	113	BLUE STAR	Andrew	C	10.000
109	124	MAGNUM	Bob	C	10.000

Chuẩn hóa dữ liệu

Sau chuẩn hóa 1NF:

- Tách ra 2 bảng: Project Detail và Employee Detail

Project_id	Project_name
113	BLUE STAR
124	MAGNUM

Emp_no	Emp_name	Grade	Salary
142	John	A	20.000
168	James	B	15.000
263	Andrew	C	10.000
109	Bob	C	10.000

- Xác định thuộc tính khóa chính: Project_id và Emp_no

Chuẩn hóa dữ liệu

Dạng chuẩn 2NF:

- ❖ Đáp ứng các yêu cầu của dạng chuẩn 1NF
- ❖ Không có các phần phụ thuộc một phần trong các bảng
- ❖ Các bảng được liên kết qua các khóa ngoại

Phụ thuộc một phần có nghĩa là thuộc tính không phải khóa nhưng phụ thuộc vào nhiều hơn một thuộc tính khóa.

Chuẩn hóa dữ liệu

Xem xét ví dụ: Chuẩn 1NF gồm 2 bảng Project Detail và Employee Detail sau khi chuẩn hóa 2NF.

Tách thành 3 bảng: Project Detail, Employee Detail và Project Employee Detail

Project_id	Project_name
113	BLUE STAR
124	MAGNUM

Emp_no	Emp_name	Grade	Salary
142	John	A	20.000
168	James	B	15.000
263	Andrew	C	10.000
109	Bob	C	10.000

Emp_no	Project_id
142	113
142	124
168	113
263	113
109	124

Chuẩn hóa dữ liệu

Dạng chuẩn 3NF:

- ❖ Đáp ứng các yêu cầu của dạng chuẩn 2NF
- ❖ Các bảng không được có các phần phụ thuộc bắc cầu trong chúng

Nếu thuộc tính có thể xác định được bằng thuộc tính khác không phải khóa, nó được gọi là phần phụ thuộc bắc cầu. Nếu một thuộc tính không khóa có thể được xác định bằng một thuộc tính không khóa khác, nó cần phải đưa vào bảng khác

Chuẩn hóa dữ liệu

Xem xét ví dụ: Chuẩn 2NF có bảng Employee Detail. Có thuộc tính Salary đang phụ thuộc vào thuộc tính **Grade** chứ không phải là Khóa chính Emp_no.

--> Tách thành bảng khác: Employee Detail và Grade Salary Detail

Emp_no	Emp_name	Grade
142	John	A
168	James	B
263	Andrew	C
109	Bob	C

Grade	Salary
A	20.000
B	15.000
C	10.000

Chuẩn hóa dữ liệu

Sau khi chuẩn hóa dữ liệu 3NF ta thu được 4 bảng dữ liệu sau:

Project_id	Project_name
113	BLUE STAR
124	MAGNUM

Emp_no	Project_id
142	113
142	124
168	113
263	113
109	124

Emp_no	Emp_name	Grade
142	John	A
168	James	B
263	Andrew	C
109	Bob	C


Grade	Salary
A	20.000
B	15.000
C	10.000

Các khái niệm mức vật lý

- ❖ **Mỗi thực thể được biểu diễn bởi một bảng (table). Bảng bao gồm các cột (column), các hàng/bản ghi (record) dữ liệu.**
 - Mỗi column biểu diễn một thuộc tính và có kiểu dữ liệu (Data type) nhất định.
 - Mỗi record thể hiện một thực thể
 - Mỗi table có một khóa chính (Primary key) - xác định duy nhất của một record trong tập dữ liệu của table. Khóa chính gồm có một hoặc một vài thuộc tính của table.

Kiểu dữ liệu (Data type)

- ❖ Mỗi column trong table được quy định bởi một kiểu dữ liệu nhất định.
- ❖ Kiểu dữ liệu cho phép xác định:
 - Loại dữ liệu của cột như dạng số nguyên, kiểu kí tự, ngày tháng,...
 - Giới hạn miền giá trị cho column

	Column Name	Data Type	Allow Nulls
	Date	datetime	<input checked="" type="checkbox"/>
	Name	varchar(50)	<input checked="" type="checkbox"/>
	Weight	float	<input checked="" type="checkbox"/>
	Position	numeric(18, 2)	<input checked="" type="checkbox"/>
	Effective_Maturity	float	<input checked="" type="checkbox"/>
	Yield_to_Worst	float	<input checked="" type="checkbox"/>
	Modified_Duration	float	<input checked="" type="checkbox"/>
	OAD	float	<input checked="" type="checkbox"/>
	OAC	float	<input checked="" type="checkbox"/>
	Coupon	float	<input checked="" type="checkbox"/>
	Credit_Rating	varchar(10)	<input type="checkbox"/>
	 CUSIP	varchar(10)	<input type="checkbox"/>
	Market_Value	numeric(18, 2)	<input checked="" type="checkbox"/>
	Principal	numeric(18, 2)	<input checked="" type="checkbox"/>
	Yield_to_Maturity	float	<input checked="" type="checkbox"/>
	CTD	float	<input checked="" type="checkbox"/>
	OAS	float	<input type="checkbox"/>

Kiểu dữ liệu (Data type)

Kiểu dữ liệu	Ý nghĩa
bigint	Số nguyên 8 byte.
binary[(n)]	Dữ liệu nhị phân có kích thước cố định n byte. Kích thước lưu trữ là n+4 byte. Với $1 < n < 8000$
<u>bit</u>	Dữ liệu số nguyên nhận các giá trị 0, 1 hoặc NULL
<u>char[(n)]</u>	Dữ liệu ký tự, có chiều dài n ký tự, không hỗ trợ Unicode. Với độ dài $1 < n < 8000$.
cursor	Tham chiếu tới một con trỏ. Chỉ dùng cho các biến và tham số trong stored procedure.
<u>datetime</u>	Dữ liệu ngày giờ. Giá trị nhận từ 1/1/1753 đến 31/12/9999
decimal[(p,[s])] <u>numeric[(p,[s])]</u>	Dữ liệu số thập phân. p là tổng số ký tự số có thể được lưu, s là số chữ số thập phân.
<u>float[(n)]</u>	Dữ liệu số động có phạm vi từ $-1.79E+308$ đến $1.79E+308$
image	Dữ liệu nhị phân có chiều dài thay đổi, dài hơn 8000 byte và tối đa là $2^{31}-1$ byte.

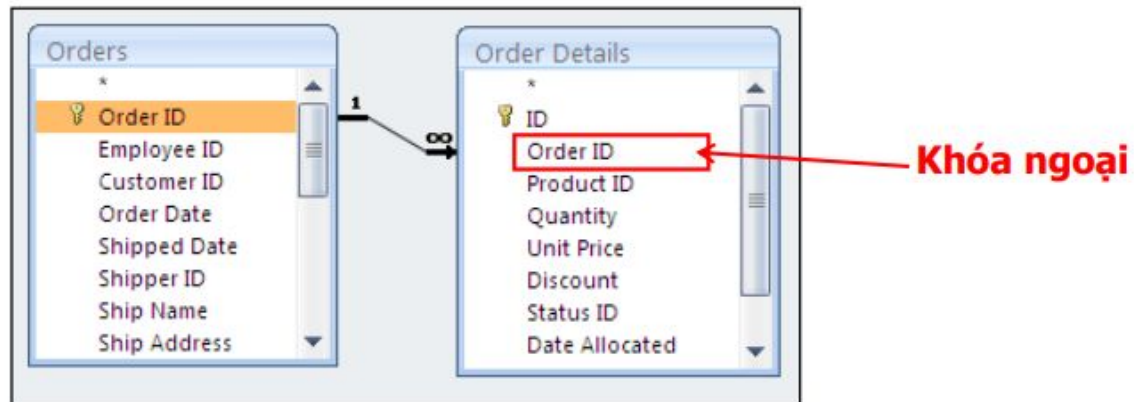
<u>integer hoặc int</u>	Dữ liệu số nguyên từ -2^{31} đến $2^{31}-1$. Chiếm 4 byte.
money	Dữ liệu kiểu tiền tệ từ -2^{63} đến 2^{63} . Chiếm 8 byte.
nchar[(n)]	Dữ liệu ký tự, có chiều dài cố định n ký tự. Có hỗ trợ Unicode. Với $1 < n < 4000$. Chiếm 2n byte.
<u>ntext</u>	Dữ liệu ký tự có chiều dài thay đổi, với chiều dài tối đa là $2^{30}-1$ ký tự. Hỗ trợ unicode.
<u>nvarchar[(n)]</u>	Dữ liệu ký tự có chiều dài thay đổi với n ký tự. Có hỗ trợ Unicode. Với $1 < n < 4000$.
real	Dữ liệu số thực động, phạm vi từ $-3.4E+38$ đến $3.4E+38$. Chiếm 4 byte.
smalldatetime	Dữ liệu ngày giờ từ 1/1/1900 đến 6/6/2079. Chiếm 4 byte.
smallint	Dữ liệu số nguyên từ -2^{15} đến $2^{15}-1$. Chiếm 2 byte.
smallmoney	Dữ liệu kiểu tiền tệ từ -2^{31} đến 2^{31} . Chiếm 4 byte.
sql_variant	Cho phép giữ các giá trị của các kiểu dữ liệu khác nhau (tất cả các kiểu dữ liệu hệ thống khác).

Kiểu dữ liệu (Data type)

sysname	Đây là kiểu dữ liệu đặc biệt, do SQL Server định nghĩa là kiểu nvarchar(128). Chiếm 256 byte.
table	Tương tự như bảng tạm, khai báo gồm danh sách các cột và các kiểu dữ liệu.
<u>text</u>	Dữ liệu ký tự có chiều dài thay đổi, dài hơn 8000byte. Có thể lưu tới 2^{31} ký tự. Không hỗ trợ Unicode.
timestamp	Cột timestamp được cập nhật tự động mỗi khi dòng được thêm hoặc được cập nhật. Mỗi bảng chỉ có thể có 1 cột timestamp. Kích thước lưu trữ là 8 byte.
tinyint	Dữ liệu số nguyên từ 0 đến 255. Chiếm 1 byte.
unique-identifier	Giá trị nhị phân 16 byte, là số định danh duy nhất toàn cục.
varbinary[(n)]	Dữ liệu nhị phân n có chiều dài thay đổi. Với $1 < n < 8000$
<u>varchar[(n)]</u>	Dữ liệu ký tự có chiều dài thay đổi với n ký tự. Không hỗ trợ Unicode. Với $1 < n < 4000$.

Khái niệm khóa (Key)

- ❖ **Khóa chính (Primary key):** Một hoặc một số thuộc tính (column) dùng để phân biệt một bộ (record) dữ liệu duy nhất trong một thực thể (table).
- ❖ **Khóa ngoại (Foreign key):** Một hoặc một số thuộc tính của một thực thể (quan hệ) R1 có quan hệ với thực thể (quan hệ) R2. Các thuộc tính khóa ngoại của R1 phải chứa giá trị trùng khớp với những giá trị trong R2.



Tổng kết

- Database
- RDBMS với SQL Server
- Mô hình E-R
- Kiểu dữ liệu
- Khóa chính, khóa ngoại

