



# Data Management Using Microsoft SQL Server

## Chương: 1

### Các khái niệm trong Hệ quản trị cơ sở dữ liệu

- Giải thích khái niệm dữ liệu và cơ sở dữ liệu
- Mô tả các phương pháp tiếp cận để quản lý dữ liệu
- Định nghĩa một Hệ thống Quản trị Cơ sở dữ liệu (Database Management System - DBMS) và liệt kê lợi ích của nó
- Giải thích các mô hình cơ sở dữ liệu khác nhau
- Định nghĩa và giải thích RDBMS
- Mô tả thực thể và bảng và liệt kê các đặc điểm của bảng
- Liệt kê sự khác biệt giữa một DBMS và RDBMS

- Các tổ chức thường xuyên phải duy trì một khối lượng dữ liệu lớn, chúng được phát sinh ra từ kết quả của các hoạt động trong công việc hằng ngày.
- Một cơ sở dữ liệu:
  - là một hình thức tổ chức các dữ liệu đó.
  - có thể bao gồm một hoặc nhiều dữ liệu có liên quan được gọi là bản ghi(record).
  - là một tập hợp dữ liệu để có thể có nhiều câu hỏi khác nhau được đặt ra trên đó.
- Ví dụ,
  - 'Số điện thoại và địa chỉ của năm bưu điện gần nhất là gì?'
  - hoặc
  - 'Chúng ta có bất kỳ cuốn sách nào trong thư viện mà có liên quan với thực phẩm sức khỏe?

# Dữ liệu(Data) và Cơ sở dữ liệu(Database)

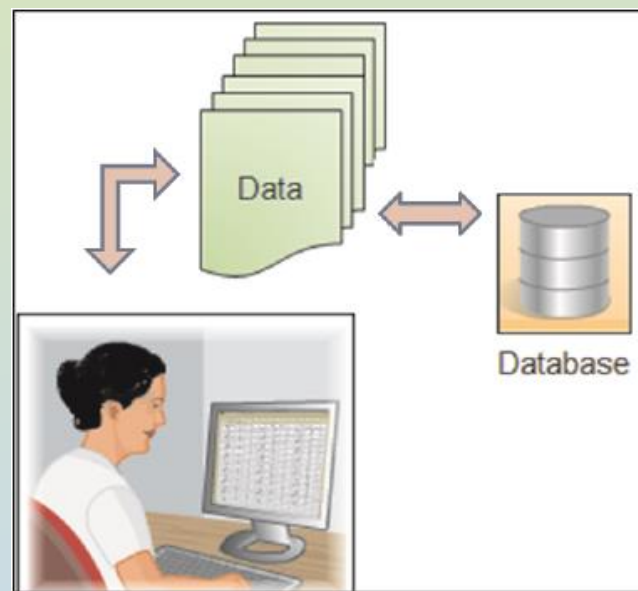
Khi dữ liệu được tổng hợp và phân tích, nó sinh ra thông tin. Việc biểu diễn dữ liệu thông minh sinh ra các thông tin.

Thông tin sẽ giúp cho việc dự đoán (forsee) và lên kế hoạch cho các sự kiện, chiến lược trong tương lai.

Một cơ sở dữ liệu là một tập hợp các dữ liệu về hoạt động (như mua bán, tuyển sinh, khám chữa bệnh, ...) của một tổ chức cụ thể được lưu trữ có tổ chức trên các thiết bị lưu trữ để các chương trình ứng dụng có thể truy cập, quản lý và cập nhật một cách dễ dàng

Một **cuốn sổ điện thoại** là một cơ sở dữ liệu bao gồm tên, địa chỉ, và số điện thoại.

- Hình dưới đây minh họa về khái niệm cơ sở dữ liệu:

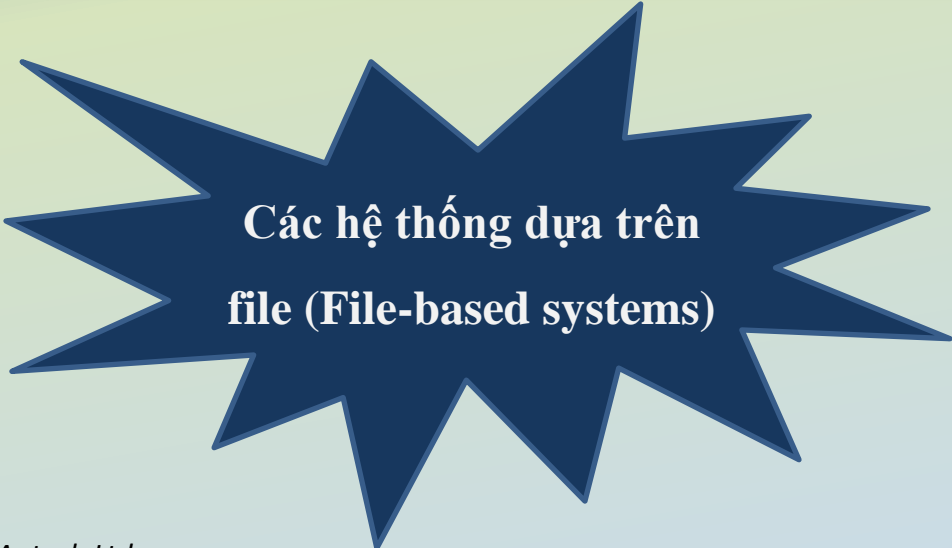


# Quản trị dữ liệu (Data Management)

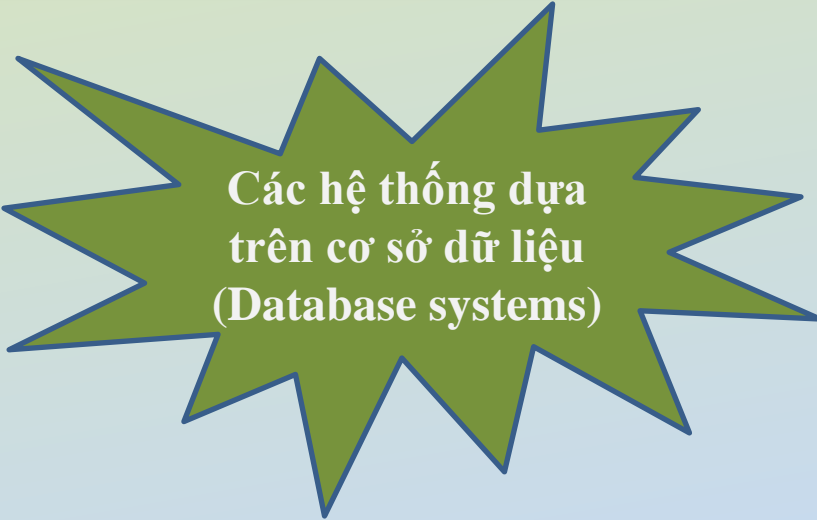
Quản trị dữ liệu là đề cập đến việc phải quản lý một khối lượng lớn thông tin, bao gồm:

- lưu trữ thông tin
- cung cấp các kỹ thuật để thao tác với thông tin
- cung cấp việc lưu trữ thông tin an toàn trong nhiều tình huống khác nhau

Có hai cách tiếp cận khác nhau để quản lý dữ liệu như sau:



**Các hệ thống dựa trên  
file (File-based systems)**



**Các hệ thống dựa  
trên cơ sở dữ liệu  
(Database systems)**

# Hệ thống dựa trên tập tin(File-based Systems)

Trong hệ thống dựa trên tập tin, dữ liệu được lưu trữ trong các tập tin rời rạc và một bộ sưu tập các tập tin như vậy được lưu trữ trên một máy tính.

Các tập tin lưu trữ dữ liệu được gọi là bảng vì chúng trông giống như bảng được sử dụng trong lưu giữ tập tin truyền thống.

Hàng trong bảng được gọi là bản ghi(record) và các cột được gọi là các trường(fields). Một ví dụ về hệ thống dựa trên tập tin được minh họa trong bảng sau:

First Name	Last Name	Address	Phone
Eric	David	ericd@eff.org	213-456-0987
Selena	Sol	selena@eff.org	987-765-4321
Jordan	Lim	nadroj@otherdomain.com	222-3456-123

# Nhược điểm của các hệ thống dựa trên file

Dữ liệu bị dư thừa và không nhất nhất quán

Những truy vấn dị thường

Dữ liệu cô lập

Dị thường khi đồng thời truy cập dữ liệu

Các vấn đề về bảo mật

Các vấn đề về toàn vẹn



# Các hệ thống CSDL(Database Systems)

Hệ thống cơ sở dữ liệu phát triển vào cuối những năm 1960 nhằm giải quyết các vấn đề phổ biến trong các ứng dụng xử lý khối lượng dữ liệu lớn, mà còn là dữ liệu chuyên sâu.

Các CSDL được dùng để lưu trữ dữ liệu một cách hiệu quả và có tổ chức sao cho quản lý được nhanh chóng và dễ dàng.

Tại một thời điểm bất kỳ, dữ liệu có thể được lấy, được thêm, và được tìm kiếm có điều kiện từ những các csdl.

Việc lưu trữ dữ liệu có thể đạt được bằng việc sử dụng các tập tin hướng dẫn sử dụng đơn giản.

Dữ liệu được lưu trữ ở dạng này không phải là vĩnh viễn. Các bản ghi trong các tập tin hướng dẫn sử dụng chỉ có thể được duy trì trong một vài tháng hoặc vài năm.



# Các thuận lợi của hệ thống CSDL

Giảm thiểu lưu trữ dữ liệu dư thừa

Đảm bảo dữ liệu nhất quán.

Dữ liệu lưu trữ có thể chia sẻ

Các tiêu chuẩn có thể được thiết lập và được tuân thủ

Toàn vẹn dữ liệu được duy trì

Có thể thực thi được bảo mật dữ liệu

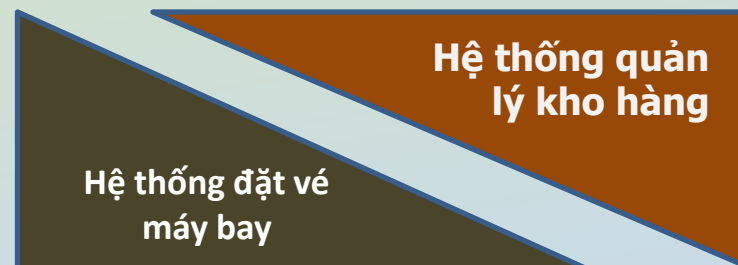
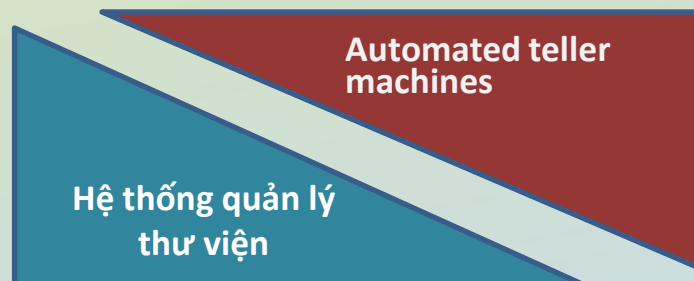
# Hệ Quản Trị Cơ Sở Dữ Liệu (DBMS) 1-2

Một Hệ QTCSDL (DBMS) là một tập các bản ghi có liên quan đến và một tập các chương trình truy cập và thao tác trên các bản ghi này và cho phép người dùng nhập vào, lưu trữ và quản lý dữ liệu.

Trong hệ thống csdl tập trung, csdl được lưu trữ ở vị trí trung tâm để mọi người có thể truy xuất từ máy tính của họ.

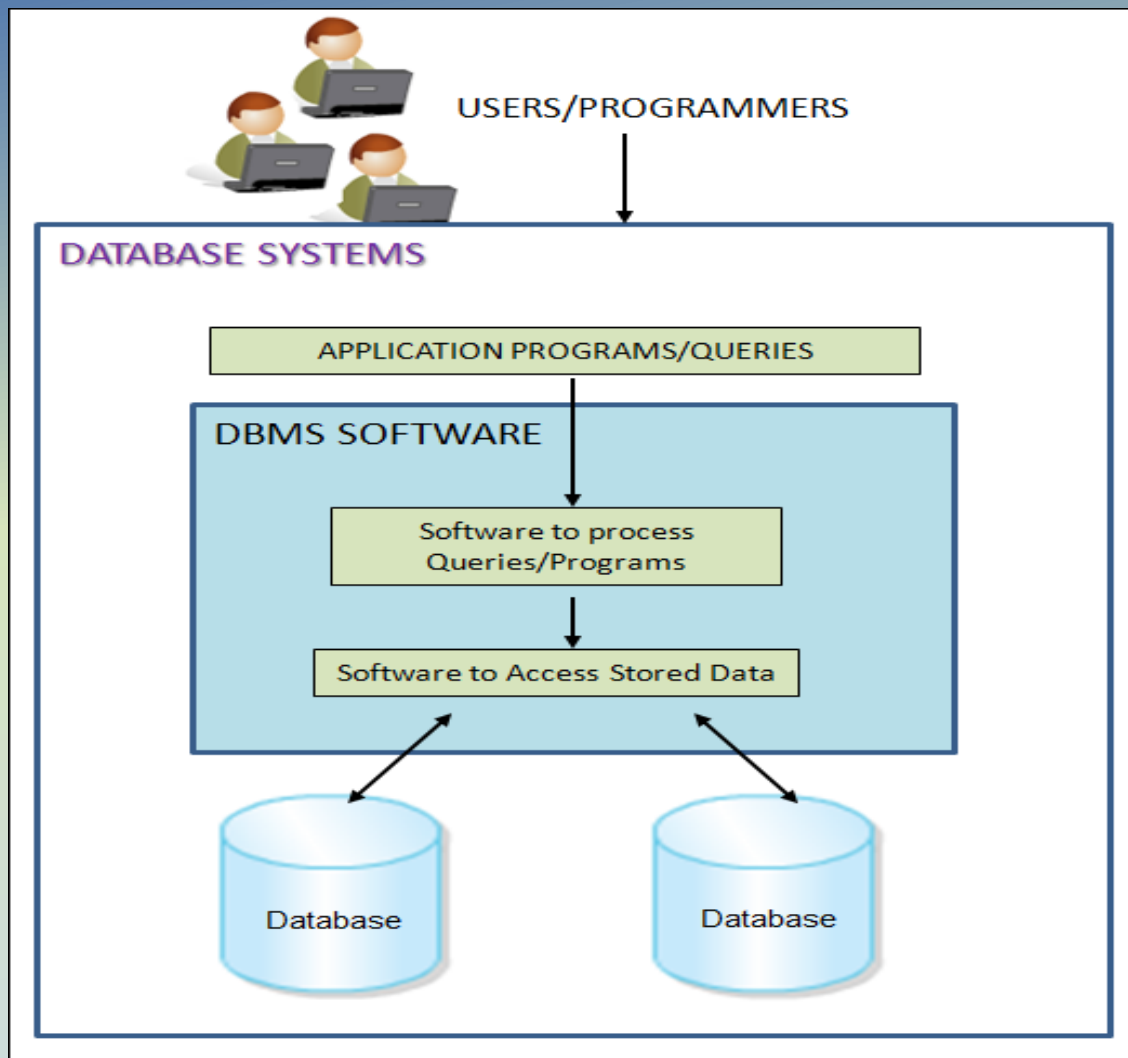
Một cơ sở dữ liệu là một tập hợp các dữ liệu liên quan đến nhau, và DBMS là một tập hợp các chương trình sử dụng để thêm hoặc sửa đổi dữ liệu này

Ví dụ về các ứng dụng cơ sở dữ liệu:



# Hệ Quản Trị Cơ Sở Dữ Liệu (DBMS) 2-2

- Dưới đây là hình minh họa về một hệ quản trị cơ sở dữ liệu





# Thuận lợi của Hệ QTCSDL 1-2

Hệ quản trị csdl có trách nhiệm xử lý dữ liệu và chuyển nó thành thông tin.

Cho phép thực hiện các thao tác như: truy vấn cơ sở dữ liệu để lấy ra dữ liệu cụ thể, cập nhật csdl, và cuối cùng là sinh ra các báo cáo.

Các báo cáo này là nguồn của thông tin là dữ liệu đã được xử lý.

Hệ quản trị csdl còn có trách nhiệm đảm tính toàn vẹn và bảo mật dữ liệu.

# Thuận lợi của Hệ QTCSDL 2-2

Lưu trữ dữ liệu

Định nghĩa dữ liệu

Thao tác dữ liệu

Toàn vẹn và bảo mật dữ liệu

Dữ liệu có thể được truy cập đồng thời và khả năng phục hồi

Tối ưu hóa hiệu suất

Kiểm soát nhiều người dùng truy cập

Các ngôn ngữ truy xuất csdl và giao diện lập trình ứng dụng(API)

# Các mô hình cơ sở dữ liệu

Tùy vào mục đích xây dựng hệ thống mà các CSDL có thể khác nhau về chức năng và mô hình dữ liệu.

Mô hình dữ liệu mô tả về container lưu trữ dữ liệu, quá trình lưu trữ và truy xuất dữ liệu từ container đó.

Phân tích và thiết kế các mô hình dữ liệu đã là sự phát triển cơ bản của cơ sở dữ liệu.

Mỗi mô hình đã phát triển từ mô hình trước đó. Mô hình cơ sở dữ liệu thường được sử dụng như sau:

**Mô hình dữ  
liệu flat-file**

**Mô hình dữ  
liệu phân cấp**

**Mô hình dữ  
liệu mạng**

**Mô hình dữ  
liệu quan hệ**

# Mô hình dữ liệu Flat-file

Trong mô hình này, csdl chỉ gồm có một bảng hoặc một file.

Mô hình này được sử dụng cho các csdl đơn giản - ví dụ, để lưu trữ các roll numbers, các họ tên, các môn học và các điểm của một nhóm sinh viên.

Không thể dùng mô hình này để quản lý dữ liệu quá phức tạp. Nó gây ra sự dư thừa dữ liệu khi dữ liệu được lặp đi lặp lại nhiều lần.

➤ Bảng sau đây mô tả cấu trúc một csdl flat file:

Roll Number	First Name	Last Name	Subject	Marks
45	Jones	Bill	Maths	84
45	Jones	Bill	Science	75
50	Mary	Mathew	Science	80



# Mô hình dữ liệu phân cấp 1-3

Trong mô hình này, các bản ghi khác nhau có mối quan hệ phân cấp, hoặc có cấu trúc giống như cây (tree).

Trong mô hình này, các mối quan hệ được suy nghĩ dưới dạng cha và con.

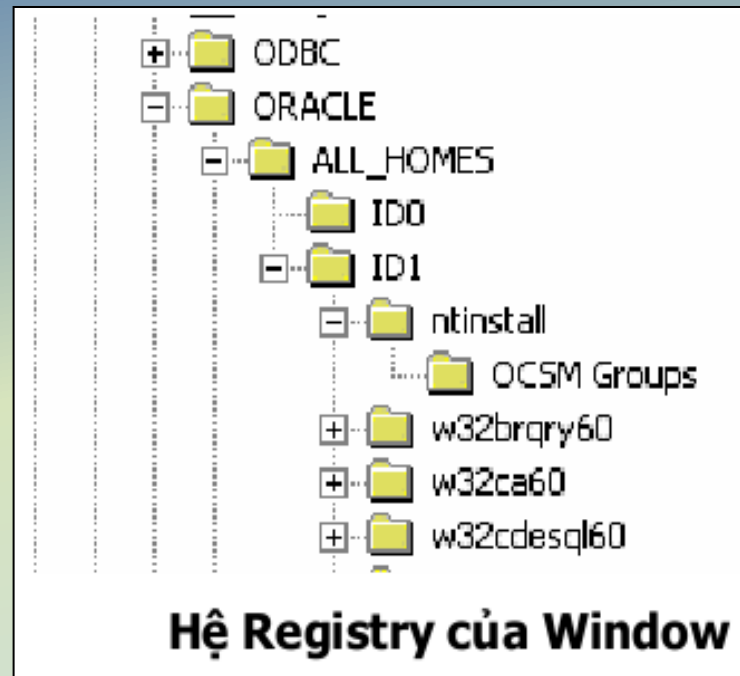
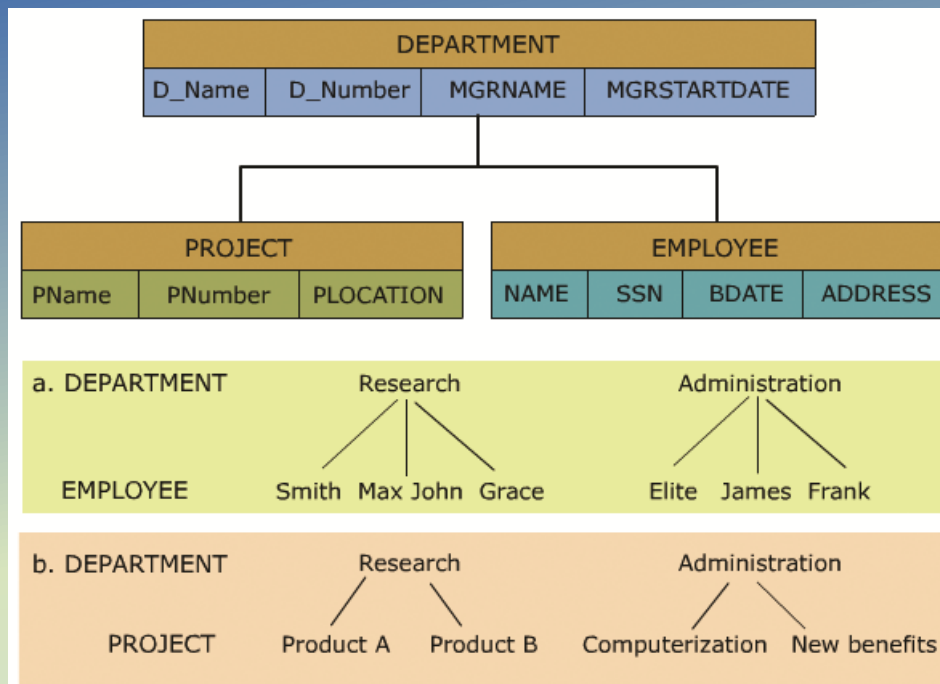
Một bản ghi cha có thể có nhiều bản ghi con, nhưng một bản ghi con chỉ có một bản ghi cha.

Để tìm dữ liệu lưu trữ trong mô hình này, người dùng cần phải biết cấu trúc của cây.

Windows Registry là một ví dụ của csdl có cấu trúc, được dùng để lưu trữ các thiết lập cấu hình và các tùy chọn của hệ điều hành Microsoft Windows.

# Mô hình dữ liệu phân cấp 2-3

- Hình minh họa về csdl theo mô hình phân cấp



- Bên trong mô hình phân cấp, Department(phòng ban) được xem như là cha của một phân đoạn (segment).
- Các bảng Project và Employee là cá bảng con.
- Một đường dẫn, mà điểm xuất phát từ phân đoạn cha rồi đi, định nghĩa ra một cây.
- Một dãy các phân đoạn có sắp xếp này chỉ ra cấu trúc phân cấp được gọi là đường dẫn phân cấp.

# Mô hình dữ liệu phân cấp 3-3

## ➤ Các thuận lợi của mô hình phân cấp như sau:

Dữ liệu được lưu giữ trong một csdl chung, vì vậy việc chia sẻ dữ liệu một cách dễ dàng, và đảm bảo bảo mật bởi hệ quản trị csdl.

Sự độc lập dữ liệu được cung cấp bởi hệ quản trị csdl giúp giảm được chi phí bảo trì chương trình.

Mô hình này rất hiệu quả khi một csdl có chứa một khối lượng dữ liệu lớn..

## ➤ Ví dụ, một hệ thống tài khoản khách hàng của ngân hàng rất phù hợp với mô hình phân cấp, do tài khoản khách hàng là chủ thể của rất nhiều giao dịch

# Mô hình dữ liệu mạng 1-4

Mô hình này tương tự như mô hình dữ liệu phân cấp, thực sự nó là tập con của mô hình phân cấp.

Trong mô hình mạng, dữ liệu được lưu trữ trong các bộ (sets), thay cho dạng cấu trúc phân cấp. Điều này nhằm để giải quyết vấn đề dư thừa dữ liệu.

Cho phép các bản ghi con có thể có nhiều bản ghi cha. Do vậy các bản ghi được liên kết vật lý thông qua danh sách liên kết (linked-lists).

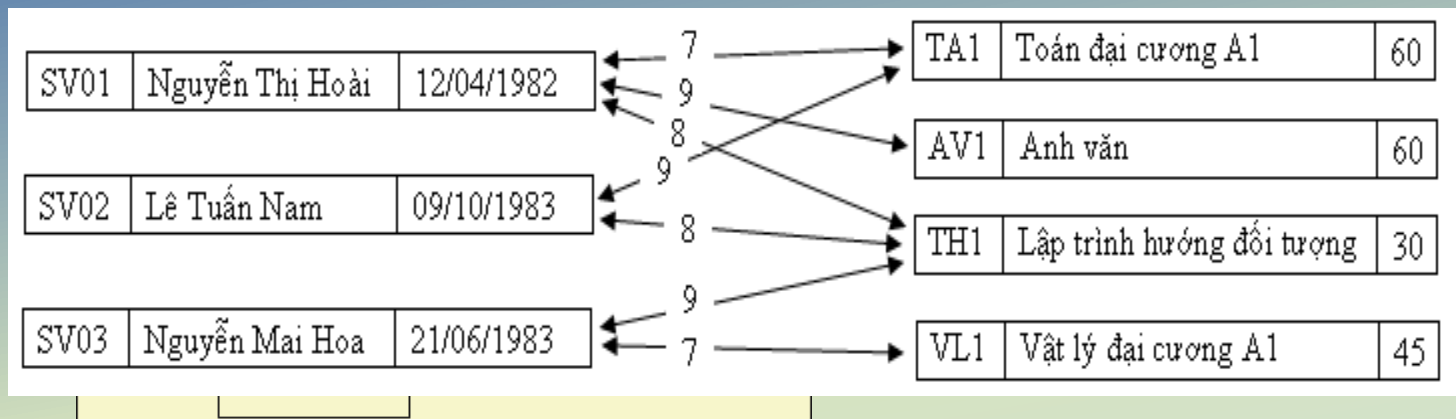
Với mỗi csdl, định nghĩa một tên csdl, kiểu bản ghi cho mỗi bản ghi, và các thành phần tạo nên các bản ghi đó được lưu trữ. Điều này được gọi là lược đồ mạng.

Khi các chương trình ứng dụng nhìn thấy một phần csdl mà sản xuất các thông tin mong muốn từ các dữ liệu được chứa trong cơ sở dữ liệu được gọi là tiểu giản đồ (sub-schema).

Nó cho phép các chương trình ứng dụng truy cập dữ liệu cần thiết từ cơ sở dữ liệu. Bộ quản lý cơ sở dữ liệu Raima (RDM- Raima Database Manager) Server Raima Inc là một ví dụ về một DBMS theo mô hình mạng.

## Mô hình dữ liệu mạng 2-4

- Mô hình mạng thể hiện trong hình dưới đây minh họa một loạt các mối quan hệ một-nhiều :



- Một nhân viên bán hàng có thể viết nhiều vé **hóa đơn(Invoice)**, nhưng mỗi **hóa đơn** được viết bởi duy nhất một nhân viên bán hàng (**Salesrep**).
- Một **khách hàng** có thể mua hàng vào những dịp khác nhau.
- Một **khách hàng** có thể có nhiều vé **hóa đơn**, nhưng mỗi **hóa đơn** chỉ thuộc về một khách hàng.
- Vé **Hóa đơn** có thể có nhiều **dòng Hóa đơn (Invline)**, nhưng mỗi **Invline** được tìm thấy trên một vé **hóa đơn** duy nhất.
- Một sản phẩm có thể xuất hiện trong **Invline** khác nhau, nhưng mỗi **Invline** chỉ chứa một sản phẩm duy nhất.

# Mô hình dữ liệu mạng 3-4

- Các thành phần của ngôn ngữ được sử dụng với mô hình mạng như sau:

## Data Definition Language(DDL)

- Được sử dụng để tạo, xóa cơ sở dữ liệu và các đối tượng của cơ sở dữ liệu. Nó cho phép người quản trị cơ sở dữ liệu để định nghĩa các thành phần lược đồ.

## Sub-schema DDL

- Cho phép người quản trị cơ sở dữ liệu để định nghĩa các thành phần cơ sở dữ liệu.

## Data Manipulation Language (DML)

- Được dùng để chèn, lấy, và sửa đổi thông tin cơ sở dữ liệu.

## Data Control Language (DCL)

- Được sử dụng để quản lý quyền truy cập vào cơ sở dữ liệu và các đối tượng cơ sở dữ liệu.

# Mô hình dữ liệu mạng 4-4

- Các thuận lợi của một cấu trúc như vậy được chỉ ra như sau:

Các mối quan hệ dễ dàng thực thi hơn trong mô hình cơ sở dữ liệu mạng so với mô hình phân cấp.

Mô hình này đảm bảo tính toàn vẹn cơ sở dữ liệu.

Mô hình này đạt được khả năng độc lập dữ liệu.

- Một số không thuận lợi được chỉ ra như sau:

Cơ sở dữ liệu theo mô hình này là khó khăn trong việc thiết kế.

Các lập trình viên phải biết rõ cấu trúc bên trong để truy cập vào cơ sở dữ liệu.

Mô hình này cung cấp một môi trường truy cập định vị dữ liệu.

- Mô hình này là khó thực thi và duy trì.
- Sử dụng mô hình này là các lập trình viên máy tính, chứ không phải là người dùng cuối.



# Mô hình dữ liệu quan hệ 1-5

Khi thông tin tăng trưởng và cơ sở dữ liệu thêm phức tạp, các ứng dụng được đòi hỏi, thiết kế cơ sở dữ liệu, quản trị và sử dụng trở nên quá cồng kềnh.

Điều này dẫn đến sự phát triển của mô hình cơ sở dữ liệu quan hệ.

Thuật ngữ "quan hệ" có nguồn gốc từ lý thuyết tập hợp của toán học. Trong mô hình quan hệ, không giống như các mô hình phân cấp và mạng, không có liên kết vật lý.

Tất cả các dữ liệu được duy trì ở dạng bảng bao gồm các hàng và cột. Dữ liệu trong hai bảng có quan hệ thông qua các cột chung chứ không liên kết vật lý.

Các thao tác được cung cấp thao tác trên các hàng trong bảng. Mô hình này biểu diễn cho các cơ sở dữ liệu như một tập hợp các mối quan hệ.

## Mô hình dữ liệu quan hệ 2-5

Một hàng được gọi là một bộ(tuple), một cột, một thuộc tính, và bảng được gọi là một quan hệ.

Danh sách các giá trị áp dụng cho một trường (field) cụ thể được gọi là tên miền.

Một số thuộc tính có thể thuộc về cùng một miền.

Số lượng các thuộc tính của một mối quan hệ được gọi là bậc(degree) của mối quan hệ.

Số lượng các bộ dữ liệu xác định số phần tử/lực lượng của các quan hệ.

# Mô hình dữ liệu quan hệ 3-5

- Để hiểu mô hình quan hệ, hãy xem xét các bảng **Students** và **Marks** sau đây:

Roll Number	Student Name
1	Sam Reiner
2	John Parkinson
3	Jenny Smith
4	Lisa Hayes
5	Penny Walker
6	Peter Jordan
7	Joe Wong

Students Table

Roll Number	Marks Obtained
1	34
2	87
3	45
4	90
5	36
6	65
7	89

Marks Table

- Bảng **Students** hiển thị **Roll Number**, **Student Name**, và bảng **Marks** hiển thị **Roll Number** và **Marks** (điểm) đạt được bởi các sinh viên.
- Xác định vị trí các học sinh có điểm trên 40:
- Đầu tiên, xác định vị trí roll numbers của các sinh viên đạt trên 50 điểm từ bảng **Marks**.
  - Thứ hai, tên của họ phải được xác định trong bảng **Students** bằng cách khớp(matching) với roll number.

# Mô hình dữ liệu quan hệ 4-5

- Kết quả được hiển thị như bảng sau:

Roll Number	Student Name	Marks Obtained
2	John	87
4	Lisa	90
6	Peter	65
7	Joe	89

- Đã có thể nhận được thông tin này bởi hai thứ:

Thứ nhất, có cột chung giữa hai bảng là cột - Roll Number.

Thứ hai, dựa trên cột này, các bản ghi từ hai bảng này có thể khớp nối, và thông tin yêu cầu có thể lấy được.

- Trong một mô hình quan hệ, dữ liệu được lưu trữ trong các bảng.
- Một bảng trong một cơ sở dữ liệu có một cái tên duy nhất mà chỉ ra nội dung của nó.
- Mỗi bảng có thể được định nghĩa như là một giao nhau của các hàng và các cột.

# Mô hình dữ liệu quan hệ 5-5

## Ưu điểm của mô hình quan hệ

- Giúp cho lập trình dành thời gian tập trung vào khung nhìn luận lý (logical view) của cơ sở dữ liệu hơn là bận tâm về khung nhìn vật lý.
- Cung cấp truy vấn một cách linh hoạt, do đó sự phổ biến của các cơ sở dữ liệu quan hệ.
- Dễ dàng để xử lý mô hình đến mức mà ngay cả những người chưa qua đào tạo tìm thấy nó dễ dàng để tạo các báo cáo và truy vấn thuận tiện, mà không đưa ra nhiều suy nghĩ đến sự cần thiết phải thiết kế một cơ sở dữ liệu thích hợp.

## Nhược điểm của mô hình quan hệ

- Giấu tất cả sự phức tạp của hệ thống và do đó nó có xu hướng chậm hơn so với các hệ thống cơ sở dữ liệu khác.

# Hệ quản trị CSDL quan hệ (RDBMS) 1-3

Mô hình quan hệ là một nỗ lực để đơn giản hóa các cấu trúc cơ sở dữ liệu.

Biểu diễn tất cả các dữ liệu trong cơ sở dữ liệu như là các bảng hàng-cột của giá trị dữ liệu.

Một Hệ quản trị CSDL là một chương trình phần mềm giúp tạo, duy trì và thao tác trên csdl quan hệ

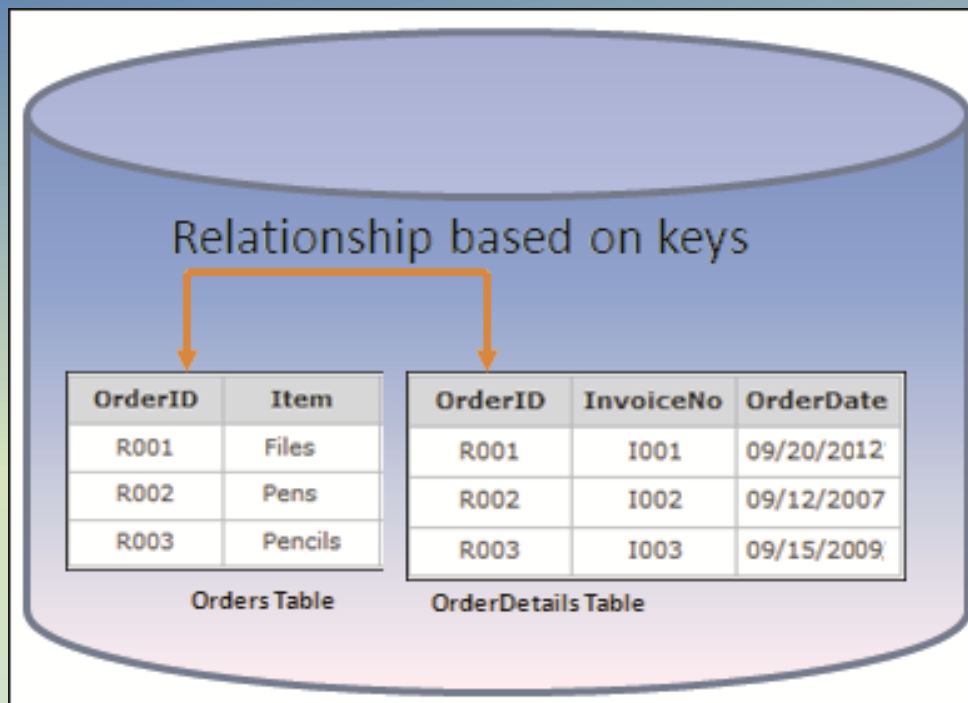
Một cơ sở dữ liệu quan hệ là một cơ sở dữ liệu chia thành các khối logic gọi là bảng, trong đó các bảng có quan hệ với nhau trong cơ sở dữ liệu.

Các bảng có quan hệ trong csdl quan hệ cho phép lấy đầy đủ dữ liệu trong một truy vấn duy nhất(cho dù các dữ liệu cần có thể tồn tại trong nhiều hơn một bảng.

Do có các khóa chung, hoặc các trường chung, giữa các bảng của cơ sở dữ liệu quan hệ, nên dữ liệu từ nhiều bảng có thể được ghép lại để tạo thành một tập kết quả lớn.

# Hệ quản trị CSDL quan hệ (RDBMS) 2-3

- Hình dưới đây cho thấy hai bảng quan hệ với nhau thông qua các khóa chung(giá trị dữ liệu) trong một cơ sở dữ liệu quan hệ:



- Do vậy, một cơ sở dữ liệu quan hệ là một cơ sở dữ liệu có cấu trúc trên mô hình quan hệ.
- Đặc trưng cơ bản của một mô hình quan hệ là dữ liệu được lưu trữ trong các quan hệ.



# Hệ quản trị CSDL quan hệ (RDBMS) 3-3

- Hai bảng **Capitals** và **Currency** dưới đây cho biết danh sách các quốc gia và các thành phố của quốc gia đó, các quốc gia và đơn vị tiền tệ sử dụng :

Country	Capital
Greece	Athens
Italy	Rome
USA	Washington
China	Beijing
Japan	Tokyo
Australia	Sydney
France	Paris

**Capitals Table**

Country	Currency
Greece	Drachma
Italy	Lira
USA	Dollar
China	Renminbi (Yuan)
Japan	Yen
Australia	Australian Dollar
France	Francs

**Currency Table**

- Cả các bảng có một cột chung, đó là cột **Country**.
- Bây giờ, để hiển thị thông tin về tiền tệ sử dụng ở Rome, đầu tiên tìm trong bảng **Capitals** tên của quốc gia mà Rome thuộc về.
- Tiếp theo, quốc gia đó cần được xem xét trong bảng **Currency** để tìm ra đơn vị tiền tệ.
- Có thể lấy được thông tin này vì có thể thiết lập một mối quan hệ giữa hai bảng thông qua một cột chung được gọi là **Country**.

# Các thuật ngữ có liên quan đến RDBMS 1-3

- Có một số thuật ngữ chủ yếu được sử dụng trong RDBMS. Chúng được mô tả như sau:

Dữ liệu được biểu diễn như là một các quan hệ.

Mỗi quan hệ được mô tả như là một bảng

Các cột là các thuộc tính.

Các hàng ('các bộ') biểu diễn cho các thực thể.

Mỗi bảng có một tập các thuộc tính được lấy ra làm 'khóa' (về mặt kỹ thuật, 'siêu khóa'), dùng để phân biệt các thực thể là duy nhất.

# Các thuật ngữ có liên quan đến RDBMS 2-3

- Hãy xem xét kịch bản của một công ty duy trì thông tin khách hàng và đơn đặt hàng cho các sản phẩm đang được bán và thông tin chi tiết khách hàng đặt hàng trong một tháng cụ thể là tháng Tám.
- Các bảng sau đây được sử dụng để minh họa cho kịch bản này:

Cust_No	Cust_Name	Phone No
002	David Gordon	0231-5466356
003	Prince Fernandes	0221-5762382
003	Charles Yale	0321-8734723
002	Ryan Ford	0241-2343444
005	Bruce Smith	0241-8472198

**Customer**

Item_No	Description	Price
HW1	Power Supply	4000
HW2	Keyboard	2000
HW3	Mouse	800
SW1	Office Suite	15000
SW2	Payroll Software	8000

**Items**

Ord_No	Item_No	Qty
101	HW3	50
101	SW1	150
102	HW2	10
103	HW3	50
104	HW2	25
104	HW3	100
105	SW1	100

**Order\_Details**

Ord_No	Ord_Date	Cust_No
101	02-08-12	002
102	11-08-12	003
103	21-08-12	003
104	28-08-12	002
105	30-08-12	005

**Order\_August**

# Các thuật ngữ có liên quan đến RDBMS 3-3

➤ Dưới đây là bảng liệt kê các thuật ngữ có liên quan đến bảng (table):

Term	Meaning	Example from the Scenario
Relation	A table	Order_August, Order_Details, Customer and Items
Tuple	A row or a record in a relation	A row from Customer relation is a Customer tuple
Attribute	A field or a column in a relation	Ord_Date, Item_No, Cust_Name, and so on
Cardinality of a relation	The number of tuples in a relation	Cardinality of Order_Details relation is 7
Degree of a relation	The number of attributes in a relation	Degree of Customer relation is 3
Domain of an attribute	The set of all values that can be taken by the attribute	Domain of Qty in Order_Details is the set of all values which can represent quantity of an ordered item
Primary Key of a relation	An attribute or a combination of attributes that uniquely defines each tuple in a relation	Primary Key of Customer relation is Cust_No  Ord_No and Item_No combination forms the primary key of Order_Details
Foreign Key	An attribute or a combination of attributes in one relation R1 that indicates the relationship of R1 with another relation R2.  The foreign key attributes in R1 must contain values matching with those of the values in R2	Cust_No in Order_August relation is a foreign key creating reference from Order_August to Customer. This is required to indicate the relationship between orders in Order_August and Customer

# Người dùng RDBMS 1-2

- Nhiều người tham gia vào việc thiết kế, sử dụng và duy trì một cơ sở dữ liệu lớn với hàng trăm người sử dụng.

## Người quản trị csdl (DBA)

- Thu thập các thông tin sẽ được lưu trữ trong cơ sở dữ liệu
- Chịu trách nhiệm việc cho phép truy cập vào cơ sở dữ liệu
- Phối hợp và giám sát việc sử dụng nó
- Mua phần mềm và tài nguyên phần cứng khi cần thiết
- Chịu trách nhiệm về các vấn đề như vi phạm về bảo mật hoặc thời gian đáp ứng của hệ thống kém

## Người thiết kế csdl (Database Designer)

- Chịu trách nhiệm về việc nhận diện các dữ liệu được lưu trữ trong cơ sở dữ liệu
- Lựa chọn cấu trúc phù hợp để biểu diễn và lưu trữ dữ liệu này
- Giao tiếp với tất cả người dùng cơ sở dữ liệu tương lai, để hiểu rõ yêu cầu của họ
- Để đi đến một thiết kế đáp ứng các yêu cầu

# Người dùng RDBMS 2-2

## Nhà phân tích hệ thống và nhà lập trình ứng dụng.

- Xác định các yêu cầu của người dùng cuối.
- Phát triển các đặc tả kỹ thuật cho các giao dịch được xác định từ trước mà đáp ứng các yêu cầu này
- Triển khai các chi tiết kỹ thuật như là các chương trình
- Kiểm tra, gỡ lỗi, tài liệu, và duy trì các giao dịch được xác định từ trước
- Thiết kế, phát triển và thao tác của các phần mềm DBMS và môi trường hệ thống

## DBMS Designers and Implementers

- Thiết kế và thực thi các mô-đun DBMS và giao diện như là một gói phần mềm.

## End User

- Người dùng cuối gọi một ứng dụng để tương tác với hệ thống, hoặc viết một truy vấn để dễ dàng lấy, sửa đổi, hoặc xóa dữ liệu.



# Thực thể

Một thực thể là một người, địa điểm, điều, đối tượng, sự kiện, hoặc thậm chí là một khái niệm, có thể nhận diện được rõ ràng..

Ví dụ, các thực thể trong một trường đại học là sinh viên, giảng viên, và các khóa học.

Mỗi thực thể có một số đặc điểm được biết đến như là thuộc tính.

Ví dụ, các thực thể sinh viên có thể gồm có các thuộc tính như số học sinh, tên, và lớp. Mỗi thuộc tính nên được đặt tên một cách phù hợp.

Một nhóm các thực thể có liên quan sẽ trở thành một tập thực thể. Mỗi tập thực thể được đặt một tên. Tên của tập thực thể phản ánh các nội dung.

Do đó, các thuộc tính của tất cả các sinh viên của các trường đại học sẽ được lưu trữ trong một tập thực thể gọi là *SinhVien*.



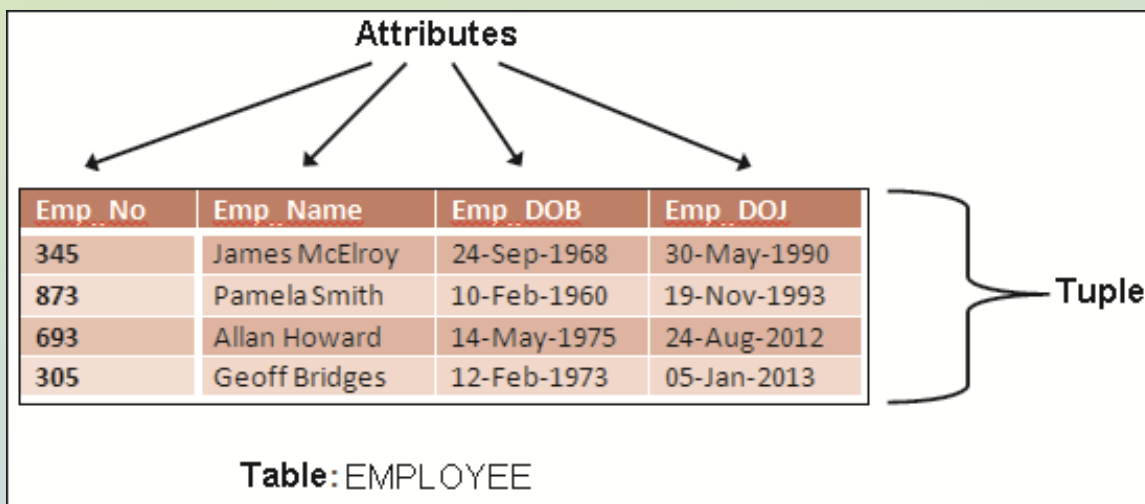
# Bảng và các đặc điểm của chúng 1-2

Truy cập và thao tác dữ liệu được hỗ trợ bởi việc tạo ra các mối quan hệ dữ liệu dựa trên một cấu trúc được gọi là một bảng.

Một bảng chứa một nhóm của các thực thể có liên quan với nhau đó là một tập thực thể. Thuật ngữ tập thực thể và bảng thường được sử dụng thay thế cho nhau.

Một bảng cũng được gọi là một quan hệ. Các hàng được gọi là bộ dữ liệu. Các cột được gọi là thuộc tính

➤ Hình dưới đây làm nổi bật các đặc tính của một bảng:



## Bảng và các đặc điểm của chúng 2-2

### ➤ Các đặc tính của bảng như sau:

Một cấu trúc hai chiều gồm các hàng và cột được coi là một bảng.

Mỗi tuple đại diện cho một thực thể duy nhất trong tập thực thể.

Mỗi cột có một tên riêng biệt.

Mỗi giao nhau của hàng với cột biểu diễn cho một giá trị dữ liệu duy nhất.

Mỗi bảng phải có một khóa, còn gọi là khóa chính xác định duy nhất mỗi hàng.

Tất cả các giá trị trong một cột phải phù hợp với các định dạng dữ liệu tương tự.

Mỗi cột có một phạm vi cụ thể của các giá trị được gọi là miền thuộc tính.

Mỗi hàng mang thông tin mô tả về một thực thể.

Thứ tự của các hàng và cột là không quan trọng trong một DBMS.

# Sự khác biệt giữa DBMS và RDBMS

DBMS	RDBMS
Nó không nhất thiết phải chứa dữ liệu trong cấu trúc như là bảng(tabular) và không cần phải có các mối quan hệ giữa các mục dữ liệu	Trong RDBMS, bắt buộc phải là cấu trúc bảng và mối quan hệ giữa các bảng được ép buộc bởi hệ thống. Các mối quan hệ này cho phép người dùng có thể áp dụng và quản lý các qui tắc xử lý bằng các đoạn mã nhỏ.
Số lượng nhỏ dữ liệu được lưu trữ và thao tác.	RDBMS có thể lưu trữ và thao tác trên một số lượng lớn dữ liệu.
DBMS kém đảm bảo hơn RDBMS.	RDBMS kém đảm bảo hơn DBMS.
Nó là 1 hệ thống đơn người dùng.	Nó là 1 hệ thống đa người dùng
Hầu hết các DBMS không hỗ trợ kiến trúc client/server.	Nó hỗ trợ kiến trúc client/server.

- Một CSDL là một tập dữ liệu có quan hệ được lưu trữ trong các bảng.
- Mô hình dữ liệu mô tả về container lưu trữ dữ liệu, và quá trình lưu trữ và truy xuất dữ liệu từ container đó.
- Một hệ quản trị CSDL có thể được định nghĩa như một tập các bản ghi quan hệ và các chương trình có thể truy cập và thao tác trên các bản ghi.
- DBMS là một tập các chương trình mà nó cho phép người dùng lưu trữ, cập nhật và trích rút thông tin từ một CSDL.
- Một hệ quản trị CSDL quan hệ (RDBMS) là một phần mềm cho phép tạo, thao tác, cập nhật trên một CSDL quan hệ.
- Một CSDL quan hệ được chia nhỏ thành các đơn vị logic gọi là bảng. Các đơn vị logic này có quan hệ với nhau trong CSDL.
- Các thành phần chính của RDBMS là các thực thể và các bảng.