**A. KIẾN THỨC CƠ BẢN**

**1. Cơ sở dữ liệu**

**Cơ sở dữ liệu** (*database*) (viết tắt là CSDL) là tập hợp dữ liệu có liên quan với nhau và được lưu trữ có tổ chức.

**Tập hợp dữ liệu có liên quan với nhau:** Các dữ liệu được dùng chung với nhau để giải quyết các yêu cầu (re- quirement) của người dùng (*user)*.

**Ví dụ 1:**

- Dữ liệu của học sinh gồm: Mã số, họ tên, mã lớp. Dữ liệu của lớp gồm: mã lớp, tên lớp.

– Sự liên quan với nhau: Một học sinh chỉ học một lớp. Một lớp có nhiều học sinh.

– Yêu cầu của người dùng: "In danh sách học sinh gồm các cột: mã số, họ tên, mã lớp, tên lớp".

**Tập hợp dữ liệu được lưu trữ có tổ chức:** Dữ liệu được lưu trữ trong các cấu trúc dữ liệu sao cho dễ quản lí, truy cập và cập nhật.

**Ví dụ 2:**

– Dữ liệu được lưu trữ trong các **bảng dữ liệu** (bao gồm nhiều hàng, nhiều cột).

– Bảng HọcSinh gồm các cột: MãSố, HọTên, Ngày Sinh, Giới Tính, MãLớp (cho biết học sinh học lớp nào); mỗi hàng chứa dữ liệu của một học sinh.

– Bảng Lớp gồm các cột: MãLớp, TênLớp; mỗi hàng chứa dữ liệu của một lớp.

– Bảng HọcSinh và bảng Lớp liên quan với nhau qua cột MãLớp.

**2. Cơ sở dữ liệu quan hệ**

**Quan hệ** (*relation*) được xem là một **bảng** (*table*) bao gồm nhiều **hàng** (*row*), nhiều **cột** (*column*) và thoả mãn các điều kiện: Hai hàng bất kì phải khác nhau; thứ tự của các cột là không quan trọng; thứ tự của các hàng là không quan trọng.

Mỗi hàng là một **bộ** (*tuple*) hoặc **bản ghi** (*record*). Mỗi cột là một **thuộc tính** (*attribute)* hoặc **trường** (*field*). Hai bộ bất kì của quan hệ không thể giống hệt hoàn toàn ở tất cả các thuộc tính.

**Ví dụ 3:** Quan hệ HọcSinh (được xem là bảng HọcSinh) gồm các thuộc tính: Mã Số, HọTên, Ngày Sinh, Giới Tính, Mã Lớp. Không thể có hai học sinh có cùng mã số, họ tên, ngày sinh, giới tính và mã lớp; nghĩa là bảng HọcSinh không được có hai hàng giống nhau. Quan hệ HọcSinh được biểu diễn là HọcSinh(Mã Số, Họ Tên, Ngày Sinh, Giới Tính, Mã Lớp).

**Cơ sở dữ liệu quan hệ** (*relational database*) là CSDL bao gồm các quan hệ (*bảng*) có liên quan với nhau.

**Ví dụ 4:** CSDL quan hệ CSDLHọcBạ gồm có quan hệ HọcSinh(MãSố, HọTên, Ngày Sinh, Giới Tính, MãLớp) và quan hệ Lớp(MãLớp, TênLớp); hai quan hệ này liên quan với nhau qua thuộc tính MãLớp.

**Khoá của quan hệ** (*relation key)* là tập K các thuộc tính (một hoặc nhiều thuộc tính) mà một bộ giá trị của K xác định duy nhất một bộ (bản ghi) của quan hệ và không tồn tại tập con K’ (là tập K bỏ bớt một thuộc tỉnh) mà một bộ giá trị của K′ xác định duy nhất một bộ (bản ghi) của quan hệ.

**Ví dụ 5:** Xét quan hệ Học Sinh(Mã Số, HọTên, Ngày Sinh, Giới Tính, MãLớp). Một học sinh chỉ có một mã số và một mã số chỉ gán cho một học sinh. Tập K = {MãSố, HọTên} không là khoá vì tồn tại tập con K′ = {MãSố} (là tập K bỏ bớt thuộc tỉnh HọTên) xác định duy nhất một học sinh. Do đó, khoá của quan hệ HọcSinh là {MãSố}.

***Lưu ý:***

- Khoá của quan hệ được xác định dựa vào ngữ nghĩa của các thuộc tính và sự liên quan về ngữ nghĩa giữa các thuộc tỉnh.

– Một quan hệ có thể có nhiều khoá. Các khoá của quan hệ được gọi là khoá dự tuyển

*(candidate key).*

**Ví dụ 6:** Quan hệ Học Sinh có hai khoá dự tuyển là {MãSố} và {Họ Tên, NgàySinh}.

– Kí hiệu khoá: Khoá được gạch dưới nét liền.

**Ví dụ 7:** Quan hệ HọcSinh(MãSố, HoTên, NgàySinh, Giới Tính, MãLớp) có hai khoá dự tuyển là {MãSố} và {HọTên, NgàySinh} được gạch dưới nét liền.

*-* Thuộc tỉnh ở trong một khoá được gọi là **thuộc tính khoá** (*key attribute*). Thuộc tính không ở trong khoá được gọi là **thuộc tính không khoá** (*non-key attribute*) hoặc **thuộc tính mô** **tả** (*descriptor*).

**Ví dụ 8:** MãSố, HọTên, NgàySinh là các thuộc tính khoá; GiớiTính là thuộc tính không khoá.

– Các thuộc tính khoá không được có giá trị *NULL* (giá trị rỗng).

**Ràng buộc khoá** (*key constrainst*) là "*Các giá trị của khoá không được trùng nhau và phải khác giá trị NULL*". Ràng buộc khoá phải được kiểm tra khi cập nhật dữ liệu của khoá, như sau: – Khi thêm một bộ (bản ghi) có giá trị khoá k thì giá trị k phải chưa có trong trường khoán

–Sửa giá trị khoá *k* của một bộ (bản ghi) thành giá trị *p* thì giá trị *p* phải chưa có trong trường khoá.

**Khoá chính** (*primary key*) của quan hệ là **khoá tiêu biểu** (*representative key*), thường được dùng hơn so với các khoá dự tuyển khác.

**Lưu ý:**

– Một quan hệ chỉ có một khoá chính, các khoá còn lại được gọi là **khoá luân phiên** (*alternate key*) hoặc **khoá phụ** (*secondary key*).

– Nếu một quan hệ chỉ có một khoá thì khoá này phải là khoá chính.

- Khoá chính thường có ít thuộc tính, tốt nhất là khoá đơn (có một thuộc tính).

– Khoá chính dùng để liên kết dữ liệu với các quan hệ khác thông qua mối liên kết "khoá chính - khoá ngoài".

**Khoá ngoài** (*foreign key*) của quan hệ là tập K các thuộc tỉnh (một hoặc nhiều thuộc tỉnh) của quan hệ này mà K là khoá chính của quan hệ khác.

**Ví dụ 9:** Quan hệ Học Sinh (Mã Số, HọTên, Ngày Sinh, Giới Tính, MãLớp ) có khoá ngoài là {MãLớp} vì {MãLớp) là khoá chính của quan hệ Lớp (MãLớp. Tên Lớp).

**Lưu ý**

- Kí hiệu khóa ngoài: Khoá ngoài được gạch dưới nét rời.

**Ví dụ 10:** Quan hệ Học Sinh (Mã Số, Họ Tên, Ngày Sinh, Giới Tính, MãLớp) có khoá ngoài là {MãLớp} được gạch dưới nét rời.

- Khoá ngoài không là duy nhất (*unique*): Nhiều bộ (*bản ghi*) khác nhau có thể có cùng giá trị khoá ngoài,

- Khoá ngoài có thể có giá trị NULL (giá trị rỗng).

- Tên của khoá ngoài và tên của khoá chính tương ứng có thể khác nhau nhưng chúng phải có *cùng ngữ nghĩa và cùng miền trị.*

**Ràng buộc tham chiếu** (*referential constraint*) hoặc **ràng buộc khoa ngoài** (*foreign key constraint*): Xét **bảng cha** (*parent table*) *Q* có trường *PK* là khoá chính và **bảng con** (*child table*) *R* có trường *FK* là khoá ngoài tham chiếu vào khoá chính PK của bảng cha Q. Trường khoá ngoài FK có ràng buộc tham chiếu là "*Giá trị khoá ngoài của trường FK của bằng con k phải có trong trường khoá chính PK của bảng cha Q ‘’*

Ràng buộc tham chiếu phải được kiểm tra khi cập nhật dữ liệu của bảng cha (sửa, xoá) và khi cập nhật dữ liệu (thêm, sửa) của bảng con.

–**Đối với bằng cha Q:** Trường khoá chính *PK* có ràng buộc khoá và ràng buộc tham chiều của khoả ngoài *FK* của bảng con *R*.

+ Khi sửa giá trị khoá *k* của một bản ghi trong bảng *Q* thành giá trị *p* thì giá trị *p* phải chưa có trong trường khoá *PK* (ràng buộc khoá) và giá trị *k* phải không có trong trường khoả ngoài *FK* (ràng buộc tham chiều).

+ Khi xoả một bản ghi có giá trị khoá *k* trong bảng *Q* thì giá trị *k* phải không có trong trường khoả ngoài *FK* (ràng buộc tham chiếu).

– **Đối với bảng con R:** Trường khoa ngoài *FK* có ràng buộc tham chiếu vào trường khoá chính *PK* của bảng cha *Q*.

+ Khi thêm bản ghì mới có giá khoá ngoài *k* vào bảng con *R* thì phải kiểm tra ràng buộc khoá chính của bảng con *R* và giá trị *k* phải có trong trường khoá *PK* (ràng buộc tham chiếu).

+ Khi sửa giá trị khoá ngoài *k* của một bản ghi trong bảng *R* thành giá trị *p* thì giá trị *p* phải có trong trường khoá chính *PK* (ràng buộc tham chiếu).

**3. Ngôn ngữ truy vấn sql**

**Truy vấn** *(query)* là một yêu cầu về thông tin hoặc một hành động được gửi đến hệ

quản trị CSDL để hệ quản trị CSDL thực hiện.

**Ví dụ 11:**

- Truy cập dữ liệu: Cho biết họ tên của học sinh có mã số là 123.

- Tạo bảng: Tạo bằng "Lớp" gồm các cột: Mãlớp, Tên Lớp.

**Ngôn ngữ truy vấn SQL** (*Structured Query Language)* là ngôn ngữ lập trình dùng để

quản lí và thao tác CSDL quan hệ. SQL gồm có:

*- DQL (Data Query Language*): Truy cập dữ liệu từ một hoặc nhiều bảng bằng cách

dùng lệnh SELECT.

*- DML (Data Manipulation Language):* Thêm, sửa, xoá dữ liệu bằng cách dùng các lệnh

*INSERT, UPDATE, DELETE.*

- DDL (Data Definition Language): Tạo lập, thay đổi, huỷ bỏ bảng bằng cách dùng các lệnh *CREATE, ALTER, DROP.*

*– DCL (Data Control Language):* Gán và huỷ bỏ quyền của người dùng bằng cách dùng các lệnh *GRANT, REVOKE.*

Cập nhật dữ liệu là quá trình thay đổi dữ liệu của CSDL thông qua các tác vụ: Thêm dữ liệu dùng lệnh INSERT, sửa dữ liệu dùng lệnh UPDATE, xoá dữ liệu dùng lệnh DELETE.

**Ví du 12:**

– Thêm dữ liệu: Thêm học sinh có mã số 123, họ tên là "Nguyễn Văn An", ngày sinh là 20/10/2007, giới tính "Nam", học lớp có mã lớp "12A".

**INSERT INTO** HọcSinh(MãSố, HọTên, NgàySinh, GiớiTính, Mã Lớp)

**VALUES** (123, Nguyễn Văn An','2987–19–20','Nam','12A');

– Sửa dữ liệu: Sửa họ tên của học sinh có mã số 123 thành "Nguyễn Văn A".

**UPDATE** HọcSinh

**SET** Họ Tên = 'Nguyễn Văn A’

**WHERE** *Mãsố = 123;*

– Xoá dữ liệu: Xoá học sinh có mã số là 123.

**DELETE FROM** Học Sinh

**WHERE** *Mãsố = 123;*

**Lệnh SELECT** dùng để truy cập dữ liệu của một hoặc nhiều bảng dữ liệu, gồm có các mệnh đề cơ bản:

|  |  |
| --- | --- |
| **Mệnh đề** | **Giải thích** |
| **SELECT** danh sách các cột (hoặc biểu thức). | Kết quả chứa giá trị của các cột (biểu thức) |
| **FROM** danh sách các bảng. | Các bảng có liên kết với nhau. |
| **WHERE** điều kiện chọn. | Điều kiện để chọn các hàng đưa vào kết quả. |
| **ORDER BY** cột1 **[ASC | DESC],**  cột2 **[ASC | DESC]...** | Sắp xếp kết quả của truy vấn (ASC: tăng dần; DESC: giảm dần). |

**Ví dụ 13:** Cho biết mã số, họ tên, ngày sinh của học sinh có mã số là 123.

**SELECT** *Mãsố, HọTên, NgàySinh*

**FROM** *HọcSinh*

**WHERE** *Mãsố = 123;*

Ví dụ 14: Cho biết mã số, họ tên, mã lớp, tên lớp của các học sinh học lớp có mã lớp "12A".

|  |  |
| --- | --- |
| Cách 1: Điều kiện kết nối ở mệnh đề WHERE.  **SELECT** MãSố, HọTên, Lớp.MãLớp, TênLớp  **FROM** HọcSinh, Lớp  **WHERE** HọcSinh.MãLớp = Lớp .Mãlớp  **AND** Lớp.MãLớp = '12A' | Cách 2: Điều kiện kết nối ở mệnh đề FROM.  **SELECT** MãSố, HọTên, Lớp.MãLớp, Tên Lớp  **FROM** HọcSinh **INNER JOIN** Lớp **ON** HọcSinh.MãLớp = Lớp. MãLớp  **WHERE** Lớp.Mã Lớp '12A'; |

***Lưu ý.***

– *SELECT \* FROM* bảng cho kết quả có tất cả các trường (cột) của bảng.

*Lệnh SELECT \* FROM bảng1, bảng2 (*không có mệnh đề *WHERE)* cho kết quả là tích Đề-Các (Cartesian product) *bảng1 × bảng2.*

– Cú pháp *tênbảng.têntrường* dùng để chỉ định một trường (cột) của một bảng.

– Cú pháp *têntrường* **IN** *(v1,v2, ..., vn*) dùng để kiểm tra giá trị của *têntrường* có trong danh sách giá trị (v1, v2,..,vn).

- Cú pháp têntrường **BETWEEN** *minVal* **AND** *maxVal* dùng để kiểm tra điều kiện *minVal* *≤ têntrường < maxVal.*

**4. Hệ CSDL**

**Chương trình ứng dụng** (*application program)* là các chương trình tương tác với CSDL thông qua hệ quản trị CSDL để giải quyết các yêu cầu của người dùng.

**Ví dụ 15:** Chương trình QuảnLíHọcSinh thực hiện việc cập nhật, truy cập dữ liệu bảng HọcSinh và bảng Lớp, in danh sách học sinh,... là chương trình ứng dụng sử dụng CSDL CSDLHọcBą.

**Hệ quản trị** **CSDL** (*DBMS - DataBase Management System*) là hệ thống các chương trình (phần mềm) cho phép người dùng định nghĩa, tạo lập CSDL; tạo lập, truy cập và cập nhật các bảng dữ liệu của CSDL; bảo đảm tính toàn vẹn dữ liệu (*data integrity*) của CSDL, sao lưu và phục hồi CSDL, bảo mật CSDL.

Hệ quản trị CSDL tương tác với các chương trình ứng dụng của người dùng và CSDL. Các phần mềm hệ quản trị CSDL thông dụng là: Microsoft Access, MySQL, PostgreSQL, SQL Server, Oracle, MongoDB,...

**Hệ CSDL** *(database system)* bao gồm: CSDL, hệ quản trị CSDL và các chương trình ứng dụng. Các chương trình ứng dụng truy cập và cập nhật dữ liệu của CSDL thông qua hệ quản trị CSDL.

**CSDL tập trung** (centralized database) là CSDL được lưu trữ trên một máy tính.

**Hệ CSDL tập trung** *(centralized database system)* là hệ CSDL có CSDL được lưu trữ trên

một máy tính (CSDL tập trung).

**CSDL phân tán** (*distributed database*) là tập hợp dữ liệu được phân tán trên các trạm (site) khác nhau của một mạng máy tính. Dữ liệu được lưu trữ tại mỗi trạm hình thành một CSDL cục bộ (*local database*) của trạm này. Mỗi trạm phải thực hiện các ứng dụng cục bộ . Mỗi trạm phải tham gia thực hiện ứng dụng toàn cục, còn được gọi là ứng dụng phân tán *(distributed application).*

*–***Ứng dụng cục bộ** (*local application*) là ứng dụng chạy tại một trạm và chỉ sử dụng dữ liệu cục bộ của trạm này để cho ra kết quả cuối cùng.

– **Ứng dụng toàn cục** (*global application*) là ứng dụng chạy tại một trạm và sử dụng dữ liệu của ít nhất hai trạm để cho ra kết quả cuối cùng.

*–***Hệ CSDL phân tán** (*distributed database system*) bao gồm các CSDL cục bộ, các hệ quản trị CSDL và các ứng dụng tại các trạm.

**5. Bảo mật và an toàn CSDL**

**Bảo mật CSDL** là bảo vệ dữ liệu của CSDL khỏi các mối đe doạ vô tình hoặc cố ý đối vớ tính toàn vẹn dữ liệu và quyền truy cập dữ liệu.

**Một số biện pháp để bảo mật CSDL:**

– *Kiểm soát truy cập:* Quản lí người dùng có thể truy cập CSDL bao gồm: xác thực người dùng (xác minh danh tính của người dùng), cấp quyền truy cập và kiểm tra hoạt động của người dùng.

– *Mã hoá:* Mã hoá dữ liệu để bảo mật dữ liệu được lưu trữ và được truyền giữa CSDL và các hệ thống khác, tránh bị truy cập trái phép khi lưu trữ và truyền dữ liệu.

*– Đảm bảo tính toàn vẹn dữ liệu:* Đảm bảo tính chính xác và tính nhất quán của dữ liệu trong CSDL thông qua chữ kí số và cơ chế xác thực dữ liệu giúp phát hiện và ngăn chặn cập nhật dữ liệu trái phép.

– *Kiểm tra và ghi nhật kí:* Ghi nhật kí các sự kiện như lần đăng nhập của người dùng, truy cập và cập nhật dữ liệu để kiểm tra khi có các yêu cầu cần thiết sau này.

- *Cập nhật các bản vá bảo mật:* Thường xuyên cập nhật các bản vá bảo mật cho hệ quản trị CSDL và các phần mềm ứng dụng để giảm thiểu các lỗ hổng và giải quyết kịp thời các vấn đề bảo mật.

- **An toàn CSDL** là bảo vệ dữ liệu của CSDL khỏi bị mất, bị phá huỷ hoặc sử dụng sai mục đỉnh do vô tình hoặc cố ý (ví dụ: sự cố về điện, cháy nổ, thiết bị lưu trữ bị hỏng, hệ thống máy tính bị lỗi, người dùng cập nhật nhầm dữ liệu,...).

- **Sao lưu và phục hồi CSDL:** Triển khai các quy trình sao lưu và phục hồi CSDL để khôi phục dữ liệu trong trường hợp vi phạm bảo mật, hỏng dữ liệu hoặc lỗi hệ thống. Sao lưu thường xuyên giúp giảm thiểu mất mát dữ liệu và giảm thời gian ngừng hoạt động.