

Chương 07: Ngôn ngữ SQL

MySQL

- MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở.
- MySQL thường được kết hợp cùng ngôn ngữ lập trình PHP trong việc triển khai và xây dựng các website
- Cài đặt MySQL ?

Thiết kế CSDL Quản lý công trình

THỰC TRẠNG

- Một công trình với một kinh phí nhất định được phụ trách bởi một chủ đầu tư và được xây dựng cho một chủ sở hữu nào đó.
- Công trình được chủ đầu tư thuê một hoặc nhiều kiến trúc sư để thiết kế và thuê một hoặc nhiều công nhân để tiến hành xây dựng.
- Một kiến trúc sư có thể thiết kế một hoặc nhiều công trình cùng một lúc. Khi kiến trúc sư thiết kế một công trình sẽ có một thù lao tương ứng
- Một công nhân có thể tham gia xây dựng một hoặc nhiều công trình cùng một lúc. Lúc này sẽ được ghi nhận lại ngày tham gia và số ngày đã tham gia

Thiết kế CSDL Quản lý công trình

XÂY DỰNG CƠ SỞ DỮ LIỆU

- **Architect** (id, name, birthday, sex, place, address)
- **Contractor** (id, name, address)
- **Host** (id, name, address)
- **Worker** (id, name, birthday, year, skill)
- **Building** (id, name, address, city, cost, start, *contractor_id*, *host_id*)
- **Work** (building_id, worker_id, date, total)
- **Design** (building_id, architect_id, benefit)

Ngôn ngữ SQL

ĐỊNH NGHĨA

- SQL (Structured Query Language) là một loại ngôn ngữ máy tính phổ biến để tạo, sửa, và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ.

Ngôn ngữ SQL

SQL GỒM 2 THÀNH PHẦN CHÍNH

- DDL (Data Definition Language) định nghĩa cấu trúc của CSDL.
 - Cấu trúc của CSDL: CREATE TABLE, CREATE VIEW, ALTER TABLE,...
 - Điều khiển quyền truy cập trên dữ liệu: GRANT, REVOKE,...
- DML (Data Manipulation Language) truy xuất và cập nhật dữ liệu: INSERT, UPDATE, DELETE, SELECT

Ngôn ngữ SQL – Cú pháp

- Chữ hoa đại diện cho từ khóa.
- Chữ thường đại diện cho các từ của người dùng định nghĩa
- Dấu | chỉ sự lựa chọn.
- Dấu { chỉ phần tử bắt buộc phải có.
- Dấu [chỉ phần tử tùy chọn (không bắt buộc).
- Dấu ... chỉ thành phần có thể lặp lại từ 0 đến nhiều lần.

Ngôn ngữ SQL – Lưu ý

- Hầu hết các phần trong câu lệnh SQL là không phân biệt chữ hoa chữ thường, trừ các ký tự trong chuỗi dữ liệu.
- Nên viết các từ khóa của SQL bởi ký tự chữ hoa và các từ do người dùng định nghĩa (table, column, ...) bởi ký tự chữ thường

Ngôn ngữ SQL

CƠ SỞ DỮ LIỆU QUẢN LÝ PHÒNG

PHONG (ma_phong, ten_phong, truong_phong)

Phòng có một mã số duy nhất, một tên phòng, trưởng phòng là mã nhân viên của trưởng phòng.

NHANVIEN (ma_nv, hten_nv, phai, cviec, luong, phu_cap, ma_phong)

Nhân viên có một mã duy nhất, họ tên, phái, công việc, lương, phụ cấp và thuộc một phòng nào đó.

TĐO_NN (ma_nv, ngoaingu, bang_cap)

Một nhân viên có thể biết nhiều ngoại ngữ, mỗi ngoại ngữ có thể có các bằng cấp khác nhau.

Truy vấn đơn giản

```
SELECT [ DISTINCT | ALL]  
      { * | <tên cột 1> [, <tên cột 2> [,....] ] }  
FROM <tên bảng>
```

- DISTINCT Hiển thị những dòng phân biệt.
- ALL: hiển thị tất cả các dòng (giá trị mặc định)
- * thay thế tất cả các cột trong bảng.

→ example 01-03

SẮP XẾP KẾT QUẢ

```
ORDER BY <Tên Cột | STT Cột> [ASC | DESC]  
[,<Tên Cột | STT Cột> [ASC | DESC],...]
```

- ASC Sắp xếp theo kết quả tăng dần (giá trị mặc định)
- DESC Sắp xếp theo kết quả giảm dần

→ example 04

Aggregate Functions

- AVG Xác định giá trị trung bình
- SUM Xác định giá trị tổng
- MIN Xác định giá trị nhỏ nhất
- MAX Xác định giá trị lớn nhất
- COUNT Xác định tổng số phần tử

Truy vấn con – Định nghĩa

- Nếu điều kiện chọn ở mệnh đề WHERE cần truy cập thông tin ở các bảng khác với bảng đang truy vấn để kiểm tra điều kiện ta sẽ sử dụng một câu select khác lồng trong điều kiện ở mệnh đề WHERE

Truy vấn con – Các dạng thường gặp

- Dạng 01: <Tên cột> <so sánh> (<select con>) *điều kiện đúng khi giá trị của cột so sánh đúng với giá trị trả về từ select con*
- Dạng 02: <Tên cột> <so sánh> ALL (<select con>) *điều kiện đúng khi giá trị của cột so sánh đúng với tất cả các giá trị trả về từ select con*
- Dạng 3: <Tên cột> <so sánh> ANY|SOME (<select con>) *điều kiện đúng khi giá trị của cột so sánh đúng với bất kỳ một giá trị nào trả về từ select con*

Truy vấn con – Các dạng thường gặp

- Dạng 04: <Tên cột> [NOT] IN (<select con>) *điều kiện đúng khi giá trị của cột nằm trong tập hợp các giá trị trả về của select con*
- Dạng 05: [NOT] EXISTS (<select con>) *điều kiện đúng khi kết quả trả về của select con khác rỗng.*

Truy vấn con – Các quy luật

- Mệnh đề ORDER BY không được sử dụng trong truy vấn con (mặc dù chúng có thể được dùng trong truy vấn cha).
- Mệnh đề SELECT trong truy vấn con chỉ được bao gồm một cột duy hoặc 1 biểu thức duy nhất, ngoại trừ truy vấn con sử dụng EXISTS.
- Truy vấn con có thể truy cập các cột của các bảng ở truy vấn cha bằng cách sử dụng bí danh

Truy vấn con – Kết quả trả về

- Dạng 01: Một cột và một dòng
- Dạng 02: Một cột và nhiều dòng
- Dạng 03: Một cột và nhiều dòng
- Dạng 04: Một cột và nhiều dòng
- Dạng 05: Nhiều cột và nhiều dòng

GROUP BY – HAVING

GROUP BY

- Đặc tính gom nhóm cho phép chúng ta thực hiện các chức năng trên một nhóm các dòng như là một dòng riêng biệt.
- Sử dụng mệnh đề GROUP BY sau mệnh đề FROM hoặc WHERE

```
SELECT <các cột để phân nhóm>, <hàm-kết-tập(<biểu thức>)>  
FROM <tên bảng> [ WHERE <Điều kiện> ]  
GROUP BY <Cột để phân nhóm> [, <Cột để phân nhóm> [...]]
```


GROUP BY – HAVING

HAVING

- Sử dụng mệnh đề HAVING theo sau mệnh đề GROUP BY để lọc ra các nhóm theo điều kiện, sau khi đã phân nhóm:

HAVING <điềukiệnlựa chọn trên nhóm>

THỨ TỰ THỰC HIỆN CÁC MỆNH ĐỀ

SELECT ... FROM ...

WHERE ...

GROUP BY ... HAVING ...

ORDER BY ...

FROM → WHERE → GROUP BY → HAVING → SELECT → ORDER BY

NHẬN BIẾT THÔNG BÁO LỖI

1. Tên cột không tìm thấy: *Invalid column name ...*
2. Tên bảng không tìm thấy: *Invalid object name ...*
3. Lỗi cú pháp: *Incorrect Syntac near Incorrect Syntac near*
4. Tên cột có mặt ở nhiều bảng: *Ambiguous column name*
5. Phép so sánh không tương thích kiểu: *Error converting data type*

KẾT NỐI NGOẠI – OUTER JOIN

- Nếu một dòng của 1 bảng được kết nối không khớp với dòng nào bên bảng kia, thì dòng đó sẽ không xuất hiện ở bảng kết quả
- Kết nối ngoại cho phép giữ lại những dòng không thỏa điều kiện kết nối

KẾT NỐI NGOẠI – OUTER JOIN

- Có 3 loại kết nối

1. FROM <Table1> LEFT JOIN <Table2> ON <condition>
2. FROM <Table1> RIGHT JOIN <Table2> ON <condition>
3. FROM <Table1> JOIN <Table2> ON <condition>

Xử lý chuỗi trong MySQL

| STT | Tên hàm | Chức năng |
|-----|--------------------------------|--|
| 1 | LENGTH (str) | Xác định chiều dài chuỗi str |
| 2 | CONCAT (str1, str2, ..., strn) | Nối các chuỗi str1, str2, ..., strn thành một chuỗi duy nhất |
| 3 | FORMAT (str, n) | Chuyển đổi định dạng chuỗi str theo dạng #,###,###.##, trong đó làm tròn đến n số thập phân |
| 4 | INSERT (str, pos, len, newstr) | Thay thế các ký tự trong chuỗi str từ vị trí thứ pos đến vị trí thứ pos+len-1 bởi chuỗi ký tự newstr |
| 5 | INSTR (str, substr) | Trả về vị trí xuất hiện đầu tiên của chuỗi substr trong chuỗi str |
| 6 | LEFT (str, len) | Trả về chuỗi ký tự tính từ vị trí bên trái của chuỗi str từ vị trí thứ nhất đến vị trí thứ len |

Xử lý chuỗi trong MySQL

| STT | Tên hàm | Chức năng |
|-----|---------------------------|---|
| 7 | RIGHT (str, len) | Trả về chuỗi ký tự tính từ vị trí bên phải của chuỗi str từ vị trí thứ nhất đến vị trí thứ len. |
| 8 | LOCATE (substr, str, pos) | Trả về vị trí xuất hiện đầu tiên của chuỗi substr trong chuỗi str tính từ vị trí thứ pos |
| 9 | LOWER (str) | Chuyển đổi chuỗi str thành chữ thường |
| 10 | UPPER (str) | Chuyển đổi chuỗi str thành chữ hoa |
| 11 | LTRIM (str) | Loại bỏ tất cả khoảng trắng bên trái chuỗi str |
| 12 | RTRIM (str) | Loại bỏ tất cả khoảng trắng bên phải chuỗi str |

Xử lý chuỗi trong MySQL

| STT | Tên hàm | Chức năng |
|-----|--|---|
| 13 | REPLACE (str, from_str, to_str) | Thay thế các ký tự from_str trong chuỗi str bởi các ký tự to_str |
| 14 | REVERSE (str) | Đảo ngược chuỗi str |
| 15 | SUBSTRING (str FROM pos FOR len) | Trích lọc từ chuỗi str một chuỗi mới, chuỗi mới này có các ký tự bắt đầu từ vị trí pos (tính từ vị trí bên trái) cho đến vị trí pos+len-1 |
| 16 | TRIM ([{BOTH LEADING TRAILING} [remstr] FROM] str), TRIM([remstr FROM] str) | Trả về chuỗi str với các ký tự remstr đã bị loại bỏ, theo các tiêu chí LEADING (bên trái), TRAILING (bên phải) và BOTH (cả hai vị trí bên trái và bên phải) |

Regular expression

- Sử dụng biểu thức chính quy trong MySQL được chia thành 3 nhóm từ khóa sau:
 - NOT_REGEX
 - REGEXP
 - RLIKE
- Trong nội dung này trình bày việc sử dụng biểu thức chính quy với từ khóa REGEXP. Kết quả tìm kiếm sẽ trả về 1 nếu tìm thấy hoặc trả về 0 nếu không tìm thấy.

| TT | Ký hiệu | Diễn giải |
|----|---------|--|
| 1 | ^ | Tìm từ đầu chuỗi nguồn |
| 2 | \$ | Tìm từ cuối chuỗi nguồn |
| 3 | . | Đại diện một ký tự bất kỳ |
| 4 | * | Ký tự xuất hiện 0 hoặc nhiều lần |
| 5 | + | Ký tự xuất hiện 1 hoặc nhiều lần |
| 6 | ? | Ký tự xuất hiện 0 hoặc 1 lần |
| 7 | | Sự lựa chọn (hoặc) |
| 8 | {n.m} | Số lần xuất hiện của ký tự (a* tương đương a{0}; a+ tương đương a{1}; a? tương đương a{0,1}) |

CẬP NHẬT CƠ SỞ DỮ LIỆU

- Các lệnh cập nhật CSDL của SQL
 - INSERT
 - DELETE
 - UPDATE