



R a i s i n g t h e b a r

Database & CRUD

Module: Advanced Programming with PHP

Kiểm tra bài trước

Hỏi và trao đổi về các khó khăn gặp phải trong bài "SQL View, Index & Stored Procedure"

Tóm tắt lại các phần đã học từ bài "SQL View, Index & Stored Procedure"

Mục tiêu

- Trình bày được các câu lệnh CRUD để thao tác với dữ liệu trong SQL
- Sử dụng được cú pháp INSERT để thêm mới dữ liệu
- Sử dụng được cú pháp SELECT để đọc dữ liệu
- Sử dụng được cú pháp UPDATE để sửa/cập nhật dữ liệu
- Sử dụng được cú pháp DELETE để xóa dữ liệu
- Trình bày được mệnh đề WHERE trong các câu lệnh SELECT, UPDATE, DELETE
- Trình bày được các mệnh đề GROUP BY, HAVING, ORDER BY trong câu lệnh SELECT

CRUD

- Các thao tác liên quan đến quản lý dữ liệu là các thao tác rất quan trọng và phổ biến trong một ứng dụng phần mềm
- Thường dùng thuật ngữ CRUD để nhắc đến các thao tác liên quan đến quản lý dữ liệu
- CRUD (Create – Read – Update - Delete) là tên gọi ngắn gọn để nhắc đến 4 thao tác cơ bản khi quản lý dữ liệu
 - Create: Tạo/Thêm mới dữ liệu
 - Read/Retrieve: Đọc dữ liệu
 - Update: Cập nhật/Sửa dữ liệu
 - Delete/Destroy: Xóa/Hủy dữ liệu
- Trong SQL có cung cấp các câu lệnh để thực hiện các thao tác CRUD: INSERT, SELECT, UPDATE, DELETE

Các câu lệnh CRUD trong SQL

- Create: INSERT INTO
`insert into customers(cname,city,country) values ('Hai Nam','Ha Noi','Viet Nam');`
- Create: SELECT/FROM
`select cname from customers;`
- Update: UPDATE/SET
`update customers set cname='Thanh Phong' where cname='Hai Nam';`
- Delete: DELETE/FROM
`delete from customers where cname='Thanh Phong'`

Câu lệnh INSERT

- Cú pháp:

```
INSERT INTO table_name(column1, column2, column3,...)  
VALUES (value1,value2,value3,...)
```

- Hoặc (chèn dữ liệu cho tất cả các cột và đúng thứ tự cột):

```
INSERT INTO table_name  
VALUES (value1,value2,value3,...)
```

- Ví dụ:

```
insert into students(sid,sname,scity,sphone) values ('C0120G01','Hai  
Nam','Ha Noi','0900102030');
```

Hoặc:

```
insert into students values ('C0120G01','Hai Nam','Ha  
Noi','0900102030');
```

Câu lệnh SELECT

- Cú pháp:

`SELECT column1,column2,...`

`FROM table_name;`

Hoặc (đọc tất cả các cột trong bảng):

`SELECT * FROM table_name;`

- Ví dụ:

`select sid,sname,scity,sphone from students;`

Hoặc:

`select * from students;`

Câu lệnh UPDATE

- Cú pháp:

```
UPDATE    table_name  
SET column1=value1,column2=value2,...  
[WHERE condition]
```

- Ví dụ:

```
update students set sname='Thanh Phong' where sid='C0120G01';
```

Hoặc:

```
update students set scity='Ho Chi Minh';
```

- Mệnh đề WHERE là không bắt buộc

Câu lệnh DELETE

- Cú pháp:

```
DELETE FROM table_name  
[WHERE condition]
```

- Ví dụ:

```
delete from students where cname='Thanh Phong';
```

Hoặc:

```
delete from students;
```

- Mệnh đề WHERE là không bắt buộc

Mệnh đề WHERE

- WHERE là mệnh đề tùy chọn, được sử dụng trong các câu lệnh SELECT, UPDATE, DELETE
- WHERE chỉ ra điều kiện thực thi truy vấn
- Cú pháp:
`SELECT column1,column2,...`
`FROM table_name`
`WHERE condition`
- Ví dụ:
`select sid,sname from students where scity='Ha Noi';`
- Các phép toán sử dụng trong điều kiện (condition):
 - So sánh: >;>=;<;<=;<>
 - Logic: and, or, not
 - LIKE, NOT LIKE (kèm các ký tự: _, %, [], [^])
 - IN, NOT IN

Mệnh đề GROUP BY

- GROUP BY là mệnh đề tùy chọn được sử dụng trong câu lệnh SELECT, để tập hợp các dữ liệu từ nhiều bản ghi đồng nhất và nhóm kết quả vào một hoặc nhiều cột
- GROUP BY thường được sử dụng kèm với các hàm tập hợp (COUNT, MAX, MIN AUM, AVG)
- Cú pháp:

```
SELECT column_name(s), <aggregate function>  
FROM table_name  
[WHERE condition ]  
GROUP BY column_name(s);
```
- Ví dụ:

```
select StudentCity, count(StudentID) from Students group by StudentCity
```

Mệnh đề HAVING

- HAVING là mệnh đề tùy chọn đi kèm với mệnh đề GROUP BY trong câu lệnh SELECT (muốn sử dụng HAVING bắt buộc phải có GROUP BY)
- HAVING chỉ ra điều kiện trên hàm tập hợp có trong mệnh đề SELECT

- Cú pháp:

```
SELECT column_name(s), < aggregate function >  
FROM table_name  
[WHERE condition ]  
GROUP BY column_name(s)  
HAVING condition;
```

- Ví dụ:

```
select StudentCity, count(StudentID) from Students group by StudentCity having  
count(StudentID)>3;
```

Mệnh đề ORDER BY

- ORDER BY là mệnh đề tùy chọn trong câu lệnh SELECT, được sử dụng để sắp xếp tập kết quả theo thứ tự tăng hoặc giảm dần
- Cú pháp:

```
SELECT column1, column2, ...  
FROM table_name  
[WHERE condition ]  
[GROUP BY column_name(s)  
[HAVING condition ]]  
ORDER BY column1, column2, ... [ASC|DESC];
```
- Ví dụ:

```
select * from Students order by StudentName desc;
```

Tổng kết

- Các thao tác liên quan đến quản lý dữ liệu CRUD là các thao tác rất quan trọng và phổ biến trong một ứng dụng phần mềm
- Bao gồm: CRUD (Create – Read – Update - Delete)
 - Create: Thêm mới dữ liệu, sử dụng câu lệnh INSERT
 - Read: Đọc và hiển thị dữ liệu, sử dụng câu lệnh SELECT
 - Update: Cập nhật dữ liệu, sử dụng câu lệnh UPDATE
 - Delete: Xóa dữ liệu, sử dụng câu lệnh DELETE
- Câu lệnh SELECT dùng để hiển thị dữ liệu từ bảng, có một số mệnh đề tùy chọn như WHERE, GROUP BY, HAVING, ORDER BY,...



R a i s i n g t h e b a r