

Ngôn ngữ truy vấn SQL

Nội dung

1. Giới thiệu SQL
2. Câu lệnh định nghĩa dữ liệu
3. Câu lệnh truy vấn dữ liệu
4. Thực hành

Giới thiệu SQL



Ngôn ngữ SQL là gì ?

- SQL - Structured Query Language
- Được sử dụng phổ biến trong các CSDL quan hệ
- Cách thức chuẩn để giao tiếp với CSDL
- Cung cấp đầy đủ tính năng cơ bản (định nghĩa, thêm, sửa, xóa, thay đổi,...)



SQL

Ngôn ngữ SQL là gì ?

- Ví dụ về câu lệnh SQL

```
SELECT * FROM Users
```

Search: <input type="text" value="user_id"/> <input type="text" value="="/> <input type="text" value=""/> <input type="button" value="Q Search"/> <input type="button" value="Filter"/>							
user_id	user_name	department	note	is_deleted	created_at	created_by	updated_at
123	123	admin		1	2018-05-14 22:00:00	NULL	2018-05-14 22:00:00
ID123	huong	user	Comment	0	2018-05-03 22:00:00	NULL	2018-05-24 22:00:00
shimoinaba	下稲葉耕大	admin		0	2018-05-27 22:00:00	NULL	2018-05-27 22:00:00
suzu	suzu	user		0	2018-05-06 22:00:00	NULL	2018-05-24 22:00:00
USE00001	admin	admin		0	2017-12-11 22:00:00	NULL	2018-05-18 10:23:34
USE00002	kondo1	cons		0	2018-04-26 22:00:00	NULL	2018-05-17 22:00:00
USE00003	guest	user		0	2018-04-26 22:00:00	NULL	2018-05-22 22:00:00
USE006	admin2	admin	1	0	2018-06-05 00:00:00	NULL	2018-06-05 00:00:00

Đặc điểm SQL

- Bản thân SQL không phải là một hệ quản trị CSDL, nó không thể tồn tại độc lập
- SQL là một phần của hệ quản trị CSDL, nó đóng vai trò ngôn ngữ và là công cụ giao tiếp giữa người sử dụng và hệ quản trị CSDL

Đặc điểm SQL

- Ngôn ngữ định nghĩa dữ liệu (DDL)
 - Dùng để định nghĩa, thay đổi cấu trúc hoặc xóa cấu trúc các đối tượng CSDL (TABLE, VIEW, ...)
- Ngôn ngữ thao tác dữ liệu (DML)
 - Dùng để thao tác với dữ liệu trong bảng
- Ngôn ngữ điều khiển dữ liệu (DCL)
 - Dùng để thiết lập quyền truy cập trên các đối tượng CSDL

Câu lệnh định nghĩa dữ liệu (DDL)



Câu lệnh Tạo Database

- Tạo Database

```
CREATE DATABASE <Tên_Database>
```

- Sử dụng Database

```
USE <Tên_Database>
```

Câu lệnh Tạo Table

- Tạo Table

```
CREATE TABLE <Tên_TABLE> (  
    column_name1 data_type [option],  
    column_name2 data_type [option],  
    column_name3 data_type [option],  
    ...  
)
```

Câu lệnh Tạo Table

- **Các kiểu dữ liệu cơ bản**

bigint	binary	bit	char
cursor	decimal	float	image
int	varchar	numeric	datetime
text	timestamp	tinyint	xml

Câu lệnh Tạo Table

- Ví dụ

```
CREATE TABLE example (  
    id smallint unsigned not null auto_increment,  
    name varchar(20) not null,  
    constraint pk_example primary key (id)  
);
```

Câu lệnh Sửa Table

- Thêm cột

```
ALTER TABLE <Tên_TABLE>  
    ADD COLUMN column_name data_type;
```

- Xóa cột

```
ALTER TABLE <Tên_TABLE>  
    DROP COLUMN column_name;
```

Câu lệnh Xóa Table

- Thêm cột

```
DROP TABLE <Tên_TABLE>
```

- Ví dụ

```
DROP TABLE Users
```

Câu lệnh thao tác dữ liệu (DML)



Các câu lệnh cơ bản

- SELECT
- INSERT
- UPDATE
- DELETE

Câu lệnh SELECT

- Cú pháp

```
SELECT [DISTINCT] Column  
FROM Table_name  
[WHERE Condition]  
[ORDER BY Columns [asc|desc]]  
[GROUP BY Columns]
```

Mệnh đề WHERE

- Cú pháp

```
SELECT [DISTINCT] Column  
FROM Table_name  
[WHERE Condition1 AND|OR Condition2]
```

Mệnh đề WHERE

- Toán tử quan hệ

Toán tử	Miêu tả	Ví dụ
=	Kiểm tra xem giá trị của hai toán hạng có cân bằng không, nếu có thì điều kiện trở thành true	$(A = B)$ là không true
!=	Kiểm tra xem giá trị của hai toán hạng là cân bằng hay là không, nếu không cân bằng thì điều kiện trở thành true	$(A != B)$ là true
>	Kiểm tra xem giá trị của toán hạng trái có lớn hơn giá trị của toán hạng phải không, nếu có thì điều kiện trở thành true	$(A > B)$ là không true
<	Kiểm tra xem giá trị của toán hạng trái có nhỏ hơn giá trị của toán hạng phải không, nếu có thì điều kiện trở thành true	$(A < B)$ là true
>=	Kiểm tra xem giá trị của toán hạng trái có lớn hơn hoặc bằng giá trị của toán hạng phải không, nếu có thì điều kiện trở thành true	$(A >= B)$ là không true
<=	Kiểm tra xem giá trị của toán hạng trái có nhỏ hơn hoặc bằng giá trị của toán hạng phải không, nếu có thì điều kiện trở thành true	$(A <= B)$ là true

Mệnh đề ORDER BY

- Cú pháp

```
SELECT [DISTINCT] Column  
FROM Table_name  
[ORDER BY column1, [column2] asc|desc]
```

- Ví dụ

```
SELECT * FROM sinhvien ORDER BY name ASC
```

Mệnh đề LIKE

- Cú pháp

```
SELECT [DISTINCT] Column  
FROM Table_name  
[WHERE column LIKE '%value1%']
```

- Ví dụ

```
SELECT *  
FROM sinhvien  
WHERE ten LIKE 'Nguyen%'
```

Câu lệnh INSERT

- Cú pháp

```
INSERT INTO Table_name(column1, column2,...)  
VALUES (value1, value2,...)
```

Câu lệnh UPDATE

- Cú pháp

```
UPDATE Table_name  
SET column1 = value1, column2 = value2  
[WHERE conditions]
```

Câu lệnh UPDATE

- Cú pháp

```
DELETE FROM Table_name  
[WHERE conditions]
```