



Lập trình Database

Các hàm xử lý của MySQL

Ths. Vũ Duy Khương

1

Hàm xử lý chuỗi ký tự

2

Hàm điều kiện if

3

Hàm LAST_INSERT_ID

4

Hàm xử lý thời gian

Hàm xử lý chuỗi SUBSTRING

- Hàm Substring cho phép trích xuất một chuỗi con từ một chuỗi khác, bắt đầu tại vị trí cụ thể và với một độ dài nhất định
- Cú pháp:

SUBSTRING(*str,pos*);
SUBSTRING(*str FROM pos*);

Kết quả trả về một chuỗi con từ 1 chuỗi **str** bắt đầu từ vị trí **pos**

Hàm xử lý chuỗi SUBSTRING

➤ Cú pháp:

SUBSTRING(*str, pos, len*);

SUBSTRING(*str FROM pos FOR len*);

Kết quả trả về một chuỗi con từ 1 chuỗi **str** bắt đầu từ vị trí **pos** và chuỗi con trả về chỉ có len ký tự

Hàm xử lý chuỗi SUBSTRING

➤ Ví dụ:

```
SELECT substring('MySQL Substring',7);
```

➔ Trả về: Substring

```
SELECT substring('MySQL Substring' FROM 7);
```

➔ Trả về: Substring

```
SELECT substring('MySQL Substring',7,3);
```

➔ Trả về: Sub

```
SELECT substring('MySQL Substring' FROM 7 FOR 3);
```

➔ Trả về: Sub

Hàm xử lý chuỗi SUBSTRING

- Cũng có thể sử dụng giá trị âm cho tham số pos. Nếu sử dụng giá trị âm cho tham số **pos**, sự bắt đầu của chuỗi con được tính từ cuối của chuỗi
- Ví dụ:

```
SELECT substring('MySQL Substring',-9);
```

➔ Trả về: Substring
- **Chú ý:** Hàm substr() tương đương hàm substring() trong MySQL

Hàm CONCAT

- Hàm Concat được sử dụng để nối hai hoặc nhiều chuỗi. Nếu các đối số là số, chúng sẽ được chuyển đổi thành chuỗi trước khi nối. Nếu bất kỳ đối số trong danh sách đối số là NULL, hàm concat sẽ trả về NULL
- Cú pháp:

CONCAT(*str1, str2, ...*)

- Ví dụ:

*Select CONCAT(title, '___', release_year)
title_release_year from sakila.film;*

	title_release_year
▶	ACADEMY DINOSAUR__2006
	ACE GOLDFINGER__2006
	ADAPTATION HOLES__2006
	AFFAIR PREJUDICE__2006
	AFRICAN EGG__2006
	AGENT TRUMAN__2006
	AIRPLANE SIERRA__2006
	AIRPORT POLLOCK__2006
	ALABAMA DEVIL__2006
	ALADDIN CALENDAR__2006
	ALAMO VIDEOTAPE 2006

Hàm CONCAT_WS

- MySQL cũng hỗ trợ hàm concat_ws cho phép chúng ta nối hai hay nhiều hơn hai chuỗi với một dấu phân cách được xác định trước
- Cú pháp:

CONCAT_WS(seperator, str1, str2, ...)

- Ví dụ:

*Select CONCAT_WS(';', title, release_year)
title_release_year from film;*

	title_release_year
▶	ACADEMY DINOSAUR;2006
	ACE GOLDFINGER;2006
	ADAPTATION HOLES;2006
	AFFAIR PREJUDICE;2006
	AFRICAN EGG;2006
	AGENT TRUMAN;2006
	AIRPLANE SIERRA;2006
	AIRPORT POLLOCK;2006

Hàm REPLACE

- MySQL cung cấp cho một hàm xử lý chuỗi hữu ích là Replace, cho phép thay thế một chuỗi trong một cột của một bảng bằng một chuỗi mới.
- **Cú pháp:**
***UPDATE** <tên bảng>*
***SET** tên cột = **REPLACE**(tên cột, xâu cần tìm, xâu thay thế)*
***WHERE** <các điều kiện>*
- **Chú ý:** khi tìm kiếm các văn bản để thay thế, MySQL có phân biệt chữ hoa và chữ thường.

Hàm REPLACE

- **Ví dụ:** Nếu muốn sửa lỗi chính tả trong bảng film trong cơ sở dữ liệu mẫu, sử dụng hàm Replace như sau:

```
UPDATE film  
SET special_features = REPLACE( special_features, 'Deleted','Empty' )  
WHERE film_id = 1
```

Hàm IF

- **IF** là một hàm điều khiển, trả về kết quả là một chuỗi hoặc số dựa trên một điều kiện cho trước. Cú pháp của hàm IF như sau:

IF(*expr, if_true_expr, if_false_expr*)

Trong đó:

- Tham số đầu tiên là *expr* sẽ được kiểm tra là đúng hay sai. Giá trị thực có nghĩa là *expr* không bằng 0 và *expr* không bằng NULL. Lưu ý rằng NULL là một giá trị đặc biệt, không bằng bất cứ điều gì khác, ngay cả bản thân nó.
- Nếu *expr* được đánh giá là đúng, hàm IF sẽ trả lại *if_true_expr*, nếu không nó sẽ trả lại *if_false_expr*.

➤ Ví dụ:

```
SELECT IF(1 = 2, 'true', 'false');
```

➔ Trả về: false

```
SELECT IF(1 = 1, 'true', 'false');
```

➔ Trả về: true

➤ Ví dụ:

```
SELECT title, release_year, length,  
IF(rating like 'G', 'N/A', rating) rating  
FROM film;
```

	title	release_year	length	rating
▶	ACADEMY DINOSAUR	2006	86	PG
	ACE GOLDFINGER	2006	48	N/A
	ADAPTATION HOLES	2006	50	NC-17
	AFFAIR PREJUDICE	2006	117	N/A
	AFRICAN EGG	2006	130	N/A

➤ Ví dụ:

```
SELECT SUM(IF(status = 'Shipped',1,0)) AS Shipped,  
SUM(IF(status = 'Cancelled',1,0)) AS Cancelled  
FROM orders;
```

	Shipped	Cancelled
▶	303	6

Hàm **LAST_INSERT_ID**

- Hàm **LAST_INSERT_ID** trả về ID của bản ghi cuối cùng được chèn vào bảng, với điều kiện đó là ID của cột có thuộc tính **AUTO_INCREMENT**. Trong thiết kế cơ sở dữ liệu, thường sử dụng một cột tự động tăng **AUTO_INCREMENT**. Khi chèn một bản ghi mới vào bảng có cột **AUTO_INCREMENT**, MySQL tạo ra ID cho tự động dựa trên các thiết lập của cột đó. có thể có được ID này bằng cách sử dụng hàm **LAST_INSERT_ID**.

Hàm LAST_INSERT_ID

- **Ví dụ:** Tạo ra một bảng mới để thử nghiệm được gọi là TBL. Trong bảng TBL, chúng ta sử dụng ID là cột AUTO_INCREMENT

```
CREATE TABLE tbl(  
    id INT AUTO_INCREMENT NOT NULL PRIMARY KEY,  
    description varchar(250) NOT NULL  
);
```

Sau đó, chúng ta sử dụng hàm LAST_INSERT_ID () để có được ID mới chèn.

```
INSERT INTO tbl(description)  
VALUES('MySQL last_insert_id');
```

Hàm LAST_INSERT_ID

Thực hiện truy vấn:

```
SELECT LAST_INSERT_ID();
```

	LAST_INSERT_ID()
▶	1

Hàm DATEDIFF

- Trong một số trường hợp, cần phải tính toán số ngày giữa hai mốc thời gian, ví dụ số ngày từ ngày vận chuyển và ngày yêu cầu trong một đơn đặt hàng. Trong những trường hợp này, cần phải sử dụng hàm

DATEDIFF

- **Cú pháp:**

DATEDIFF(expr1,expr2)

expr1 và *expr2* là hai mốc thời gian.

- **Ví dụ:**

SELECT DATEDIFF('2011-08-17','2011-08-17');

➔ Trả về: 0 day

SELECT DATEDIFF('2011-08-17','2011-08-08');

➔ Trả về: 9 days

Hàm DATEDIFF

➤ Ví dụ:

Để tính toán số ngày còn lại giữa ngày trả và ngày thuê trong đơn thuê film, chúng ta sử dụng **DATEDIFF** như sau:

```
SELECT inventory_id,  
DATEDIFF(return_date, rental_date) AS daysLeft  
FROM rental  
ORDER BY daysLeft DESC;
```

	inventory_id	daysLeft
▶	3644	10
	2452	10
	2058	10
	1849	10
	3773	10

Hàm ADDDATE, EXTRACT

- MySQL cũng hỗ trợ một số hàm xử lý ngày tháng khác như:
ADDDATE, EXTRACT
 - Hàm **ADDDATE**: trả về một giá trị thời gian là kết quả của thao tác trên một giá trị thời gian khác.
 - Hàm **EXTRACT**: tách ra các giá trị như ngày, tháng, năm từ một giá trị có kiểu thời gian
- **Ví dụ**: đưa ra ngày tháng sau ngày giờ hiện tại 30 ngày
SELECT ADDDATE(NOW(), INTERVAL 30 DAY);

	ADDDATE(NOW(), INTERVAL 30 DAY)
▶	2021-07-27 21:40:01

Hàm ADDDATE, EXTRACT

- Ví dụ: đưa ra các đơn đặt hàng trong khoảng 30 ngày tính từ ngày 1/5/2005

*SELECT **

FROM orders

WHERE orderDate >= '2005-5-1' AND orderDate <

ADDDATE('2005-5-1', INTERVAL 30 DAY);

orderNumber	orderDate	requiredDate	shippedDate	status	comments
10411	2005-05-01 00:00:00	2005-05-08 00:00:00	2005-05-06 00:00:00	Shipped	<small>HULL</small>
10412	2005-05-03 00:00:00	2005-05-13 00:00:00	2005-05-05 00:00:00	Shipped	<small>HULL</small>
10413	2005-05-05 00:00:00	2005-05-14 00:00:00	2005-05-09 00:00:00	Shipped	Customer requested that DHL is used for this shipping
10414	2005-05-06 00:00:00	2005-05-13 00:00:00	<small>HULL</small>	On Hold	Customer credit limit exceeded. Will ship when a payment is received.
10415	2005-05-09 00:00:00	2005-05-20 00:00:00	2005-05-12 00:00:00	Disputed	Customer claims the scales of the models don't match what was discuss
10416	2005-05-10 00:00:00	2005-05-16 00:00:00	2005-05-14 00:00:00	Shipped	<small>HULL</small>
10417	2005-05-13 00:00:00	2005-05-19 00:00:00	2005-05-19 00:00:00	Disputed	Customer doesn't like the colors and precision of the models.
10418	2005-05-16 00:00:00	2005-05-24 00:00:00	2005-05-20 00:00:00	Shipped	<small>HULL</small>
10419	2005-05-17 00:00:00	2005-05-28 00:00:00	2005-05-19 00:00:00	Shipped	<small>HULL</small>
10420	2005-05-29 00:00:00	2005-06-07 00:00:00	<small>HULL</small>	In Process	<small>HULL</small>
10421	2005-05-29 00:00:00	2005-06-06 00:00:00	<small>HULL</small>	In Process	Custom shipping instructions were sent to warehouse

Hàm ADDDATE, EXTRACT

- Ví dụ: đưa ra tháng của một giá trị thời gian:

SELECT EXTRACT(MONTH FROM '2021-06-28 23:59:59');

	EXTRACT(MONTH FROM '2004-06-28 23:59:59')
▶	6

- đưa ra năm của một giá trị thời gian:

SELECT EXTRACT(YEAR FROM '2021-06-28 23:59:59');

	EXTRACT(YEAR FROM '2021-06-28 23:59:59')
▶	2021

Bài tập thực hành

Bài 1: Import CSDL **classicmodels** từ link dưới đây:



mysqlsampledatabase.sql

1. Lấy ra 50 ký tự đầu tiên của phần mô tả sản phẩm (“productDescription”) từ Table “products”, đặt tên là ‘Title of products’
2. Đưa ra mô tả về các nhân viên (“employees”) theo định dạng ‘Fullname, jobTitle.’
3. Thay thế toàn bộ tên nhóm hàng ‘Cars’ từ “productLine” của Table “products” thành ‘Automobiles’.
4. Tìm 5 đơn hàng được vận chuyển sớm nhất so với ngày yêu cầu từ Table “orders”.
5. Đưa ra các đơn đặt hàng trong tháng 5 năm 2005 và có ngày chuyển hàng đến chưa xác định.





THANK YOU