

4.5. Hàm chuỗi

4.5.1. ASCII

Chuyển ký tự dạng chuỗi thành mã ASCII tương ứng.

Cú pháp:

BTRIM (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.

Ví dụ về ASCII:

```
SELECT ASCII('iNet solutions')
```

```
-- 105(mã ASCII của ký tự 'i')
```

```
SELECT ASCII('W')
```

```
-- 87
```

4.5.2. BTRIM

Loại bỏ khoảng trắng ở đầu và cuối ký tự.

Cú pháp:

BTRIM (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.

Ví dụ về BTRIM:

```
SELECT BTRIM(' Ititan xin chào ')
```

```
-- Ititan xin chào
```

4.5.3. CONCAT

Nối hai hoặc nhiều chuỗi lại với nhau thành một chuỗi duy nhất.

Cú pháp:

CONCAT (truong_du_lieu_1 STRING, truong_du_lieu_2 STRING) => Chuỗi

- **truong_du_lieu_1:** Chuỗi ban đầu.
- **truong_du_lieu_2:** Trường dữ liệu dạng chuỗi hoặc chuỗi.

Ví dụ về CONCAT:

```
SELECT CONCAT('Ititan xin chào','iNet solutions')
```

```
-- Ititan xin chàoiNet solutions
```

4.5.4. INITCAP

Chuyển đổi các chuỗi kí tự trong biểu thức với chữ cái đầu tiên của mỗi chữ được viết hoa.

Cú pháp:

INITCAP (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để xử lý.

Ví dụ về INITCAP:

```
SELECT INITCAP('ititan xin chào')
```

```
-- Ititan Xin Chào
```

4.5.5. INSTR

Kiểm tra sự tồn tại của một chuỗi con bên trong một chuỗi lớn hơn.

Cú pháp:

INITCAP (truong_du_lieu STRING, chuoi_con STRING) => BOOL

- **truong_du_lieu:** Chuỗi được truyền vào để xử lý.
- **chuoi_con:** Chuỗi được truyền vào để so sánh

Ví dụ về INSTR:

```
SELECT INSTR('ititan xin chào','iti')
```

```
-- 1
```

```
SELECT INSTR('ititan xin chào','abc')
```

```
-- 0
```

4.5.6. LENGTH

Trả về độ dài của chuỗi kí tự.

Cú pháp:

LENGTH (truong_du_lieu STRING) => INT

- **truong_du_lieu:** Chuỗi được truyền vào để lấy độ dài.

Ví dụ về LENGTH:

```
SELECT LENGTH('Ititan xin chào')
```

```
-- 15
```

4.5.7. LOWER

Trả về tất cả các kí tự trong một chuỗi thành chữ không in hoa.

Cú pháp:

LOWER (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để viết hoa.

Ví dụ về LOWER:

```
SELECT LOWER('ITITAN XIN CHÀO')
```

```
-- ititan xin chào
```

4.5.8. LPAD

Thêm kí tự được chỉ định hoặc khoảng trắng cho đến khi chuỗi đạt đủ số lượng.

Cú pháp:

LPAD (truong_du_lieu STRING, do_dai INT, ki_tu STRING) => Chuỗi

- **truong_du_lieu:** Chuỗi được truyền vào để thêm kí tự.
- **do_dai:** Số lượng kí tự.
- **ki_tu:** kí tự hoặc chuỗi để thêm.

Ví dụ về LPAD:

```
SELECT LPAD('Ititan xin chào', 20, ' ')
```

```
-- Ititan xin chào
```

```
SELECT LPAD('iNet Solution', 18, 'i')
```

```
-- iiiiiiNet Solution
```

Lưu ý: Nếu ki_tu không được nhập vào, giá trị mặc định là khoảng trắng sẽ được thêm. Nếu chuỗi lớn hơn do_dai sẽ trả về chuỗi gốc ban đầu.

4.5.9. LTRIM

Trả về giá trị lặp lại của các ký tự trong trường dữ liệu.

Cú pháp:

LTRIM (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.

Ví dụ về LTRIM:

```
SELECT LTRIM('iTitan xin chào')
```

```
-- iTitan xin chào
```

4.5.10. MASK_FIRST

Giấu một số ký tự đầu tiên của trường dữ liệu dạng chuỗi.

Cú pháp:

MASK_FIRST (truong_du_lieu STRING, do_dai) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.
- **do_dai:** số ký tự cần che giấu.

Ví dụ về MASK_FIRST:

```
SELECT MASK_FIRST('Ititan xin chào',4)
```

```
-- Xxxxan xin chào
```

4.5.11. MASK_LAST

Giấu một số ký tự đầu tiên của trường dữ liệu dạng chuỗi

Cú pháp:

MASK_LAST (truong_du_lieu String, do_dai) => String

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.
- **do_dai:** số ký tự cần che giấu.

Ví dụ về MASK_LAST:

```
SELECT MASK_LAST('Ititan xin chào',5)

-- Ititan xinxxxxx
```

4.5.12. REPEAT

Trả về giá trị lặp lại của các ký tự trong trường dữ liệu.

Cú pháp:

REVERSE (truong_du_lieu STRING, so_lan_lap INT) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để lặp.
- **so_lan_lap:** Số lần lặp lại các ký tự

Ví dụ về REPEAT:

```
SELECT REPEAT('Ititan xin chào')

-- Ititan xin chàoltitan xin chàoltitan xin chào
```

4.5.13. REPLACE

Tìm kiếm và thay thế giá trị của một chuỗi ký tự.

Cú pháp:

REPLACE (truong_du_lieu STRING, ki_tu_1 STRING, ki_tu_2 STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để thêm kí tự.
- **ki_tu_1:** Kí tự tìm kiếm.
- **ki_tu_2:** Kí tự thay thế.

Ví dụ về REPLACE:

```
SELECT REPLACE('Ititan xin chào', 'i', 'y')
```

```
-- ltytan xyn chào
```

```
SELECT REPLACE('iNet Solution', 'Solution', 'xin chào')
```

```
-- iNet xin chào
```

4.5.14. REVERSE

Đảo ngược vị trí các kí tự của chuỗi trong biểu thức.

Cú pháp:

REVERSE (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để đảo ngược.

Ví dụ về REVERSE:

```
SELECT REVERSE('Ititan xin chào')
```

```
-- oàhC nix natitl
```

4.5.15. RTRIM

Loại bỏ các kí tự khoảng trắng ở cuối chuỗi kí tự.

Cú pháp:

RTRIM (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để loại bỏ khoảng trắng.

Ví dụ về RTRIM:

```
SELECT RTRIM('Ititan xin chào  ')\n\n-- Ititan xin chào
```

4.5.16. STRPOS

Trả về vị trí lần đầu tiên xuất hiện của kí tự trong chuỗi.

Cú pháp:

STRPOS (truong_du_lieu STRING, ki_tu STRING) => INT

- **truong_du_lieu:** Chuỗi được truyền vào để thêm kí tự.
- **ki_tu:** Kí tự tìm kiếm.

Ví dụ về STRPOS:

```
SELECT STRPOS('Ititan xin chào', 'i')\n\n-- 3
```

```
SELECT STRPOS('iNet Solution', 'iNet')\n\n-- 1
```

```
SELECT STRPOS('iNet Solution', 'x')\n\n-- 0
```


4.5.17. STR_LEFT

Trích xuất một phần chuỗi kí tự bên trái của cột được truyền vào.

Cú pháp:

STR_LEFT (truong_du_lieu STRING, do_dai INT) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để trích xuất.
- **do_dai:** Số kí tự muốn trích xuất.

Ví dụ về STR_LEFT:

```
SELECT STR_LEFT('Ititan xin chào',3)
```

```
-- Iti
```

4.5.18. STR_RIGHT

Trích xuất một phần chuỗi kí tự bên phải của cột được truyền vào.

Cú pháp:

RTRIM (truong_du_lieu STRING, do_dai INT) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để trích xuất.
- **do_dai:** Số kí tự muốn trích xuất.

Ví dụ về RTRIM:

```
SELECT RTRIM('Ititan xin chào',3)
```

```
-- hào
```

4.5.19. SUBSTR

Trích xuất một phần chuỗi kí tự bên của trường dữ liệu được truyền vào.

Cú pháp:

STR_LEFT (truong_du_lieu STRING, vi_tri INT, do_dai INT) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để trích xuất.
- **do_dai:** Số kí tự muốn trích xuất.

Ví dụ về SUBSTR:

```
SELECT SUBSTR('Ititan xin chào',3,4)

-- itan
```

4.5.20.UPPER

Trả về tất cả các kí tự trong một chuỗi thành chữ in hoa.

Cú pháp:

UPPER (truong_du_lieu STRING) => STRING

- **truong_du_lieu:** Chuỗi được truyền vào để in hoa.

Ví dụ về UPPER:

```
SELECT UPPER('Ititan xin chào')

-- ITITAN XIN CHÀO
```

Revision #4

Created 10 July 2024 09:35:18 by Tuấn

Updated 16 July 2024 09:55:29 by Tuấn