1. **Kiểu dữ kiệu trong Hive**

Trong phần này chúng ta sẽ tìm hiểu qua những loại dữ liệu có liên quan đến tạo bảng ở trong Hive. Toàn bộ kiểu dữ liệu trong Hive được chia thành 4 loại như sau:

* Column types
* Literals
* Null Values
* Complex Types

## 1.1 Column Types

Được sử dụng để làm kiểu dữ liệu của những cột trong Hive. Chúng bao gồm:

### **Kiểu nguyên**

Các dữ liệu kiểu nguyên có thể được xác định bằng cách sử dụng kiểu dữ liệu nguyên, INT. Khi phạm vi kiểu dữ liệu vượt quá INT (4-byte số nguyên có dấu, từ -2,147,483,648 đến 2,147,483,647), chúng ta sẽ sử dụng BIGINT (8-byte số nguyên có dấu từ -9,223,372,036,854,775,808 đến 9,223,372,036,854,775,807) và nếu giới hạn nhỏ hơn INT, chúng ta có thể sử dụng SMALLINT (2- byte số nguyên có dấu) hoặc TINYINT (1-byte số nguyên có dấu). Bảng sau đây mô tả các loại dữ liệu khác nhau:

| **Type** | **Postfix** | **Example** |
| --- | --- | --- |
| TINYINT | Y | 10Y |
| SMALLINT | S | 10S |
| INT |  | 10 |
| BIGINT | L | 10L |

### **Cái kiểu String**

Kiểu dữ liệu String có thể được chỉ định bằng dấu ngoặc đơn ('') hoặc dấu ngoặc kép (""). Nó chứa hai loại dữ liệu: VARCHAR và CHAR.

| **Data type** | **Length** |
| --- | --- |
| Varchar | 1 -> 65355 |
| Char | 255 |

### **Timestamp**

Hive hỗ trợ Unix time với độ chính xác đến nano giây. Hỗ trời những dạng java.sql.Timestamp: “YYYY-MM-DD HH:MM:SS.fffffffff” và định dạng “yyyy-mm-dd hh:mm:ss.ffffffffff”.

### **Dates**

Giá trị DATE được mô tả theo định dạng năm/tháng/ngày ở dạng {{YYYY-MM-DD}}.

### **Decimals**

Kiểu định dạng Decimal trong Hive giống như định dang Big Decimal của Java. Nó được sử dụng để biểu diễn các số thập phân có độ chính xác bất biến. Cú pháp và ví dụ:

DECIMAL(precision, scale)

decimal(10,0)

## 1.2 Literals

### **Các loại dấu phẩy động**

Là những số có dấu thập phân, được giả định là Double.

### **Kiểu Decimal**

Decimal cũng là những số có dấu phẩy động với phạm vi cao hơn kiểu Double. Phạm vị của Decimal khoảng từ (-10^308 to 10^308)

## 1.3 Null Value

Các giá trị thiếu được biểu thị bằng giá trị đặc biệt Null.

## 1.4 Complex types

Dưới đâu là những kiểu dữ liệu complex:

### **Array**

* Arrays: ARRAY<data\_type>
* Maps: MAP<primitive\_type, data\_type>
* Structs: STRUCT<col\_name : data\_type [COMMENT col\_comment], ...>
* Union: UNIONTYPE<data\_type, data\_type, ...>

1. **Tìm hiểu HiveSQL**

Hive định nghĩa ra một ngôn ngữ truy vấn đơn giản có cú pháp gần giống với SQL (SQL-like query language) được gọi là HiveQL, nó cho phép người sử dụng đã quen thuộc với các truy vấn SQL thực hiện việc truy vấn dữ liệu. Ngoài ra ngôn ngữ này còn cho phép các lập trình viên người đã quen thuộc với MapReduce framework có thể nhúng các mappers và reducers cho chính họ viết ra để thực thi nhiều hơn nữa các phân tích phức tập mà không được hỗ trợ bởi các hàm đã có sẵn trong ngôn ngữ HiveQL. HiveQL cũng có thể được mở rộng với các custom scalar functions (UDF’s), aggregations (UDAF’s) và các table funtions (UDTF’s)

• Câu lệnh SELECT được sử dụng để lấy dữ liệu từ một bảng.

• Mệnh đề WHERE hoạt động tương tự như một điều kiện. Nó lọc dữ liệu bằng cách sử dụng điều kiện và cung cấp cho bạn một kết quả hữu hạn. Các toán tử và hàm tích hợp tạo ra một biểu thức, đáp ứng điều kiện.

• Mệnh đề ORDER BY được sử dụng để truy xuất các chi tiết dựa trên một cột và sắp xếp kết quả được đặt theo thứ tự tăng dần hoặc giảm dần.

• Mệnh đề GROUP BY được sử dụng để nhóm tất cả các bản ghi trong tập kết quả bằng cột thu thập cụ thể. Nó được sử dụng để truy vấn một nhóm các hồ sơ.

Cú pháp truy vấn:

SELECT [ALL | DISTINCT] select\_expr, select\_expr, ...

FROM table\_reference

[WHERE where\_condition]

[GROUP BY col\_list]

[HAVING having\_condition]

[ORDER BY col\_list]]

[LIMIT number];