Chế độ **temporaryview** của Apache Spark là một tính năng cho phép bạn tạo các view tạm thời trên các tập dữ liệu. Các view này chỉ có sẵn cho phiên Spark hiện tại và sẽ bị xóa khi phiên kết thúc.

Chế độ temporary view có thể được sử dụng cho một số mục đích, bao gồm:

* Chia sẻ dữ liệu giữa các truy vấn Spark.
* Tạo các view tạm thời từ các tập dữ liệu lớn.
* Thử nghiệm các truy vấn Spark mà không cần tạo các bảng tạm thời.

Có hai loại chế độ temporary view trong Spark:

* Local temporary views: Các view này chỉ có sẵn cho phiên Spark hiện tại.
* Global temporary views: Các view này có sẵn cho tất cả các phiên Spark trong cùng một SparkContext.

Local temporary views

Các local temporary views được tạo bằng cách sử dụng phương thức createTempView() của SparkSession. Phương thức này nhận một chuỗi đại diện cho tên view và một DataFrame.

Ví dụ:

// Create a local temporary view

val df = spark.createDataFrame(Seq((1, "a"), (2, "b")))

df.createTempView("my\_view")

// Query the view

spark.sql("select \* from my\_view").show()

Global temporary views

Các global temporary views được tạo bằng cách sử dụng phương thức createGlobalTempView() của SparkSession. Phương thức này nhận một chuỗi đại diện cho tên view và một DataFrame.

Ví dụ:

// Create a global temporary view

val df = spark.createDataFrame(Seq((1, "a"), (2, "b")))

df.createGlobalTempView("my\_view")

// Query the view from another session

val otherSession = SparkSession.builder.appName("Other Session").getOrCreate()

otherSession.sql("select \* from my\_view").show()

So sánh local temporary views và global temporary views

Dưới đây là bảng so sánh các local temporary views và global temporary views:

|  |  |  |
| --- | --- | --- |
| Tính năng | Local temporary views | Global temporary views |
| Có sẵn cho | Phiên Spark hiện tại | Tất cả các phiên Spark trong cùng một SparkContext |
| Tên view | Local\_temp\_view\_[id] | global\_temp\_view\_[id] |
| Truy cập | Chỉ có thể truy cập từ phiên Spark hiện tại | Có thể truy cập từ tất cả các phiên Spark trong cùng một SparkContext |

Tiêu đề view

Khi bạn tạo một view, bạn có thể cung cấp một tiêu đề cho view đó. Tiêu đề view sẽ được sử dụng để hiển thị tên của view trong các truy vấn Spark.

Ví dụ:

// Create a view with a title

val df = spark.createDataFrame(Seq((1, "a"), (2, "b")))

df.createTempView("my\_view", "My View")

// Query the view

spark.sql("select \* from my\_view").show()

// Output:

// 0 | My View

// 1 | a

// 2 | b

Xóa các view

Các local temporary views sẽ tự động bị xóa khi phiên Spark kết thúc. Các global temporary views có thể được xóa bằng cách sử dụng phương thức dropGlobalTempView() của SparkSession.

Ví dụ:

// Drop a global temporary view

spark.sql("dropGlobalTempView('my\_view')")

Tóm tắt

Chế độ temporary view là một tính năng hữu ích của Apache Spark cho phép bạn tạo các view tạm thời trên các tập dữ liệu. Các view này có thể được sử dụng cho một số mục đích, bao gồm:

* Chia sẻ dữ liệu giữa các truy vấn Spark.
* Tạo các view tạm thời từ các tập dữ liệu lớn.
* Thử nghiệm các truy vấn Spark mà không cần tạo các bảng tạm thời.

**GlobalTemporaryView** là một loại view tạm thời trong Apache Spark. Các view này có sẵn cho tất cả các phiên Spark trong cùng một SparkContext.

GlobalTemporaryView được tạo bằng cách sử dụng phương thức createGlobalTempView() của SparkSession. Phương thức này nhận một chuỗi đại diện cho tên view và một DataFrame.

Ví dụ:

// Create a global temporary view

val df = spark.createDataFrame(Seq((1, "a"), (2, "b")))

df.createGlobalTempView("my\_view")

// Query the view from another session

val otherSession = SparkSession.builder.appName("Other Session").getOrCreate()

otherSession.sql("select \* from my\_view").show()

Tên của các global temporary view được lưu trữ trong hệ thống lưu trữ metastore của Spark. Do đó, các view này có thể được truy cập từ tất cả các phiên Spark trong cùng một SparkContext, ngay cả khi các phiên Spark đang chạy trên các máy chủ khác nhau.

GlobalTemporaryView được sử dụng cho một số mục đích, bao gồm:

* Chia sẻ dữ liệu giữa các phiên Spark.
* Tạo các view tạm thời từ các tập dữ liệu lớn.
* Thử nghiệm các truy vấn Spark mà không cần tạo các bảng tạm thời.

Một số điểm khác biệt chính giữa GlobalTemporaryView và LocalTemporaryView bao gồm:

* Sự sẵn có: GlobalTemporaryView có sẵn cho tất cả các phiên Spark trong cùng một SparkContext, trong khi LocalTemporaryView chỉ có sẵn cho phiên Spark hiện tại.
* Tên: Tên của các GlobalTemporaryView được lưu trữ trong hệ thống lưu trữ metastore của Spark, trong khi tên của các LocalTemporaryView không được lưu trữ trong hệ thống lưu trữ metastore.
* Tuổi thọ: Các GlobalTemporaryView sẽ tồn tại cho đến khi chúng bị xóa thủ công, trong khi các LocalTemporaryView sẽ bị xóa khi phiên Spark kết thúc.

Tóm lại, GlobalTemporaryView là một tính năng hữu ích của Apache Spark cho phép bạn chia sẻ dữ liệu giữa các phiên Spark.