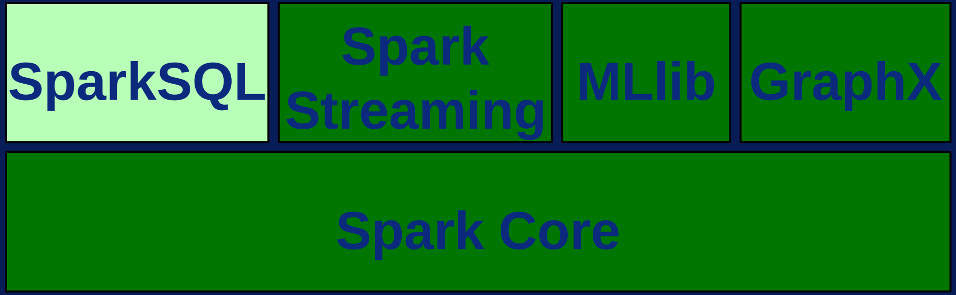
Spark SQL

After this video you will be able to..

- Process structured data using Spark’s SQL module

- Explain the numerous benefits of Spark SQL

Spark SQL



SparkSQL

Spark Streaming

MLlib

GraphX

Spark Core

- Enables querying structured and unstructured data through Spark
- Provides a common query language
- Has APIs for Scala, Java and Python to convert results into RDDs

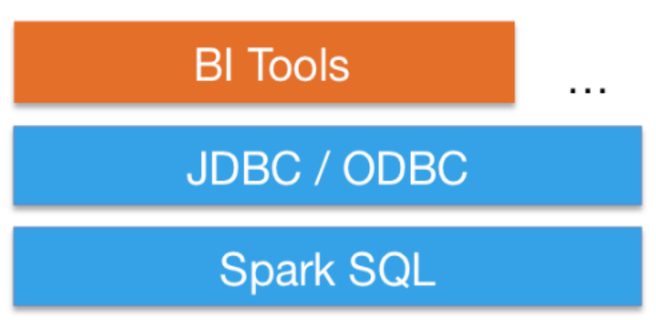
Relational Operations

Perform Relational Processing such as Declarative Queries

Embed SQL queries inside Spark Programs

Business Intelligence Tools

Spark SQL connects to all BI tools that support JDBC or ODBC standard



http://spark.apache.org/

DataFrames

Distributed Data organized as named columns

Look just like a table in relational databases

How to go Relational in Spark ?

Step 1: Create a SQLContext

```
from pyspark.sql import SQLContext
sqlContext = SQLContext(sc)
```

http://spark.apache.org/

How to go Relational in Spark ?

Create a DataFrame from

- an existing RDD
- a Hive table
- data sources

JSON → DataFrame

```
# Read
df = sqlContext.read.json("/filename.json")

# Display
df.show()
```

RDD of Row objects → DataFrame

```
# Read
from pyspark.sql import SQLContext, Row
sqlContext = SQLContext(sc)

# Load a text file and convert each line to a Row.
lines = sc.textFile("filename.txt")
cols = lines.map(lambda l: l.split(","))
data = cols.map(lambda p: Row(name=p[0], zip=int(p[1])))

# Create DataFrame
df = sqlContext.createDataFrame(data)

# Register the DataFrame as a table
df.registerTempTable(”table")

# Run SQL
Output = sqlContext.sql("SELECT * FROM table WHERE …")
```

http://spark.apache.org/

DataFrames are just like tables

http://spark.apache.org/

```
# Show the content of the DataFrame
df.show()

# Print the schema
df.printSchema()

# Select only the "X" column
df.select("X").show()

# Select everybody, but increment the discount by 5%
df.select(df["name"], df["discount"] + 5).show()

# Select people height greater than 4.0 ft
df.filter(df["height"] > 4.0).show()

# Count people by zip
df.groupBy("zip").count().show()
```

Spark SQL

Relational on Spark

Connect to variety of databases

Deploy business intelligence tools over Spark