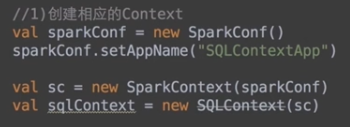
# 1 Spark

## 1.1 SQLContext

### 1.1.1 SparkContext，SQLContext，SparkSession的区别

SparkContext是Spark 1.\*中的Spark SQL的入口，必须先创建SparkContext。再用SparkContext创建SQLContext。

图1（先创建SparkContext，再用SparkContext创建SQLContext）



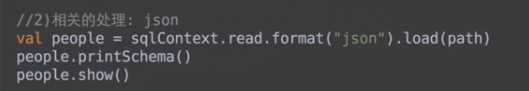
SparkSession是Spark2.\*中的入口。

图2



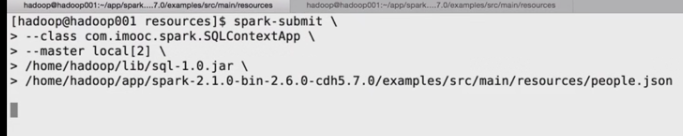
### 1.1.2 SQLContext读取JSON文件

图1



## 1.2 服务器上运行Spark SQL的jar包

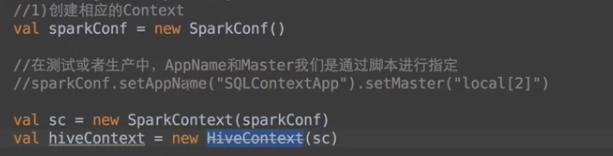
图1



## 1.3 HiveContext

SparkContext是Spark 1.\*中的Spark SQL的入口，必须先创建SparkContext。再用SparkContext创建HiveContext。

图1



### 1.3.1 HiveContext显示table的数据

图1



## 1.4 SparkSession

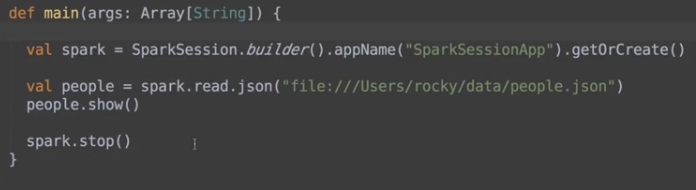
SparkSession是Spark2.\*中的入口。

图1



### 1.4.1 SparkSession读取JSON文件

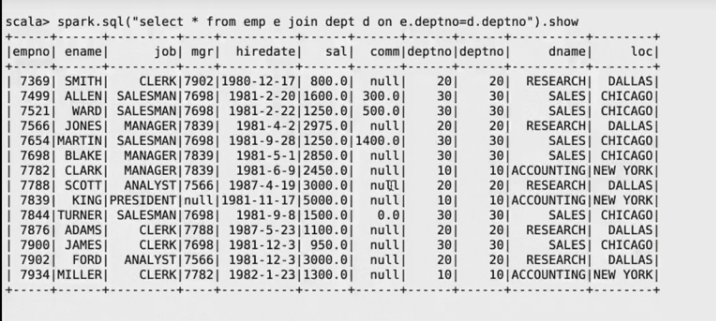
图1



## 1.5 Spark SQL相比Hive的速度

Spark SQL比Hive SQL的执行速度快很多。

图1



## 1.6 Spark thriftserver

启动thriftserver，然后通过客户端（beeline/code）去连接该thriftserver。

图1



图2



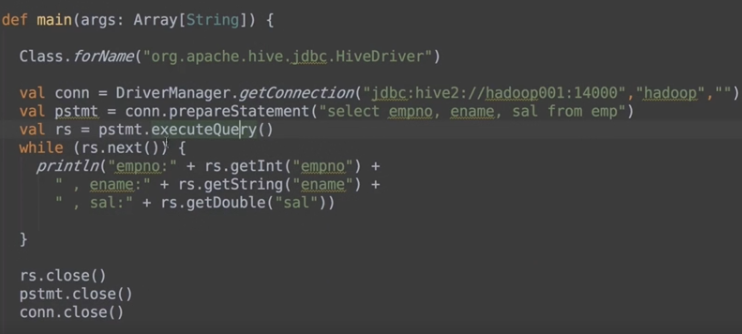
### 1.6.1 thriftserver和spark-shell/spark-sql的区别

1.每一次启动spark-shell或spark-sql都会启动一个Spark Application。

2.启动thriftserver时会启动一个Spark Application，但后面无论多少个客户端连接，永远都是一个Spark Application。

### 1.6.2 JDBC访问thriftserver

图1



## 1.7 DataFrame和DataSet

DataSet是一个分布式数据集，DataFrame是一个以列（列名，列类型，列值）的形式构成的分布式数据集。

图1



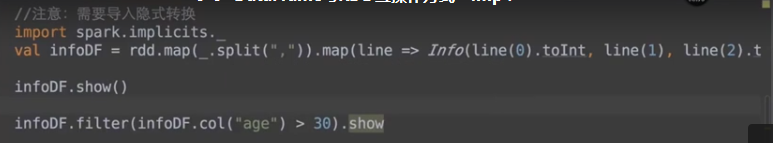
图2（DataFrame的基本API）



### 1.7.1 反射的方式根据RDD获取DataFrame

这种方式必须预先知道schema。

图1



### 1.7.2 structType的方式根据RDD获取DataFrame

这种方式不需要预先知道schema。

图1

