# 快速理解Spark Dataset

## 1. 前言

RDD、DataFrame、Dataset是Spark三个最重要的概念，RDD和DataFrame两个概念出现的比较早，Dataset相对出现的较晚（1.6版本开始出现），有些开发人员对此还不熟悉，本文重点引领快速理解Dataset。

带着几个问题去阅读：  
1、DataFrame比RDD有哪些优点？  
2、DataFrame和Dataset有什么关系？  
3、有了DataFrame为什么还有引入Dataset？  
4、Dataset在Spark源码中长什么样？

**注：本文的环境基于当前最新版本 Spark-2.1.1**

## **2、RDD/DataFrame快速回顾**

### RDD

弹性分布式数据集，是Spark对数据进行的一种抽象，可以理解为Spark对数据的一种组织方式，更简单些说，RDD就是一种数据结构，里面包含了数据和操作数据的方法

从字面上就能看出的几个特点：

#### 弹性：

数据可完全放内存或完全放磁盘，也可部分存放在内存，部分存放在磁盘，并可以自动切换

RDD出错后可自动重新计算（通过血缘自动容错）

可checkpoint（设置检查点，用于容错），可persist或cache（缓存）

里面的数据是分片的（也叫分区，partition），分片的大小可自由设置和细粒度调整

#### 分布式：

RDD中的数据可存放在多个节点上

#### 数据集：

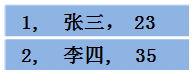
数据的集合，没啥好说的

相对于与DataFrame和Dataset，RDD是Spark最底层的抽象，目前是开发者用的最多的，但逐步会转向DataFrame和Dataset（当然，这是Spark的发展趋势）

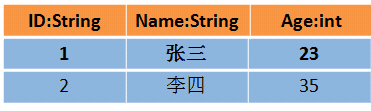
#### DataFrame：

理解了RDD，DataFrame就容易理解些，DataFrame的思想来源于Python的pandas库，RDD是一个数据集，DataFrame在RDD的基础上加了Schema（描述数据的信息，可以认为是元数据，DataFrame曾经就有个名字叫SchemaRDD）

假设RDD中的两行数据长这样



那么DataFrame中的数据长这样



从上面两个图可以看出，DataFrame比RDD多了一个表头信息（Schema），像一张表了，DataFrame还配套了新的操作数据的方法，DataFrame API（如df.select())和SQL(select id, name from xx\_table where ...)。

有了DataFrame这个高一层的抽象后，我们处理数据更加简单了，甚至可以用SQL来处理数据了，对开发者来说，易用性有了很大的提升。

不仅如此，通过DataFrame API或SQL处理数据，会自动经过Spark 优化器（Catalyst）的优化，即使你写的程序或SQL不高效，也可以运行的很快，很爽吧！

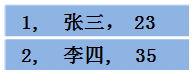
**注意：DataFrame是用来处理结构化数据的**

## 3. 步入正文，Dataset

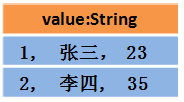
官方解释如下（英语不好的同学随意的瞄一眼即可）：  
A Dataset is a distributed collection of data. Dataset is a new interface added in Spark 1.6 that provides the benefits of RDDs (strong typing, ability to use powerful lambda functions) with the benefits of Spark SQL’s optimized execution engine. A Dataset can be constructed from JVM objects and then manipulated using functional transformations (map, flatMap, filter, etc.). The Dataset API is available in Scala and Java. Python does not have the support for the Dataset API. But due to Python’s dynamic nature, many of the benefits of the Dataset API are already available (i.e. you can access the field of a row by name naturally row.columnName). The case for R is similar.

相对于RDD，Dataset提供了强类型支持，也是在RDD的每行数据加了类型约束

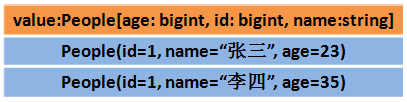
假设RDD中的两行数据长这样



那么Dataset中的数据长这样



或者长这样（每行数据是个Object）

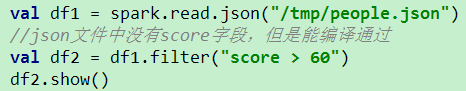


使用Dataset API的程序，会经过Spark SQL的优化器进行优化（优化器叫什么还记得吗？）

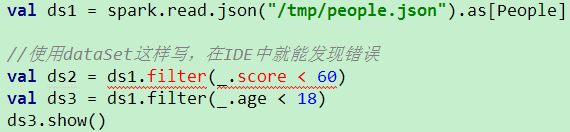
目前仅支持Scala、Java API，尚未提供Python的API（所以一定要学习Scala）

相比DataFrame，Dataset提供了编译时类型检查，对于分布式程序来讲，提交一次作业太费劲了（要编译、打包、上传、运行），到提交到集群运行时才发现错误，实在是想骂人，这也是引入Dataset的一个重要原因。

使用DataFrame的代码中json文件中并没有score字段，但是能编译通过，但是运行时会报异常！如下图代码所示

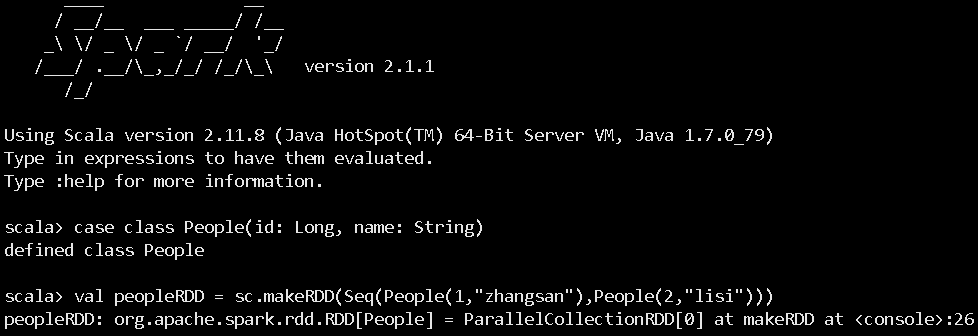


而使用Dataset实现，会在IDE中就报错，出错提前到了编译之前

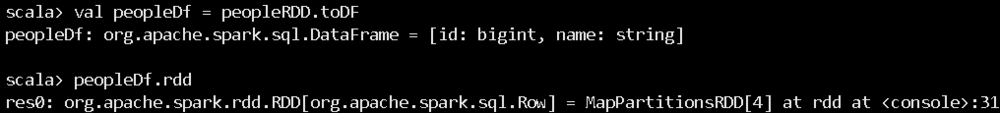


RDD转换DataFrame后不可逆，但RDD转换Dataset是可逆的（这也是Dataset产生的原因）。如下操作所示：

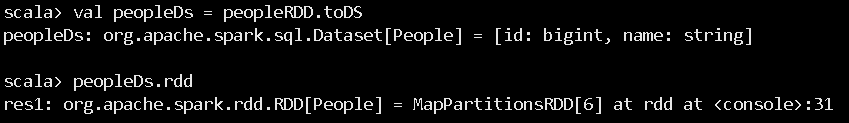
启动spark-shell，创建一个RDD



通过RDD创建DataFrame，再通过DataFrame转换成RDD，发现RDD的类型变成了Row类型

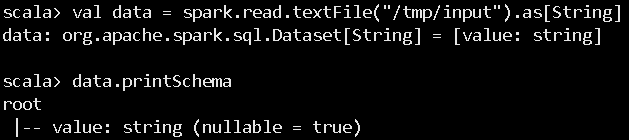


通过RDD创建Dataset，再通过Dataset转换为RDD，发现RDD还是原始类型

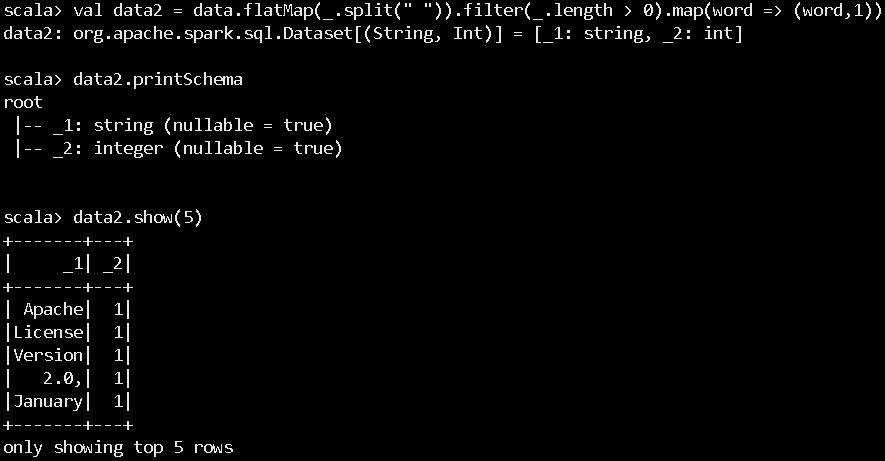


4. Dataset基本操作

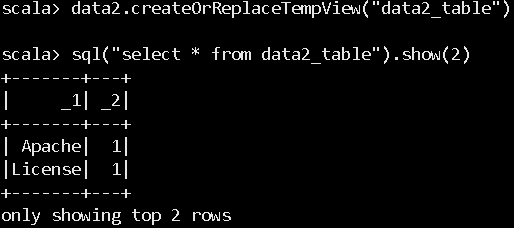
将Spark安装目录的LICENSE文件上传至HDFS上，将文件读入Spark，使用as[]转换为DataSet



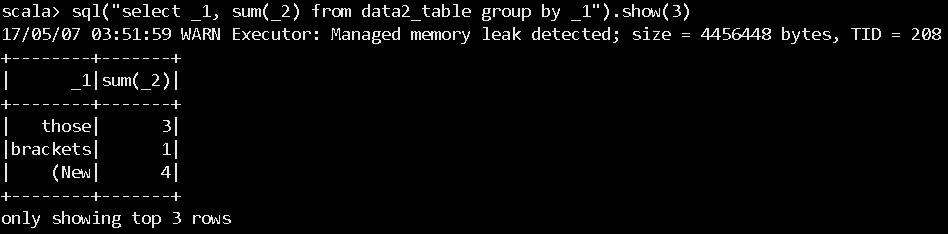
使用Dataset API做转换操作（as转换）



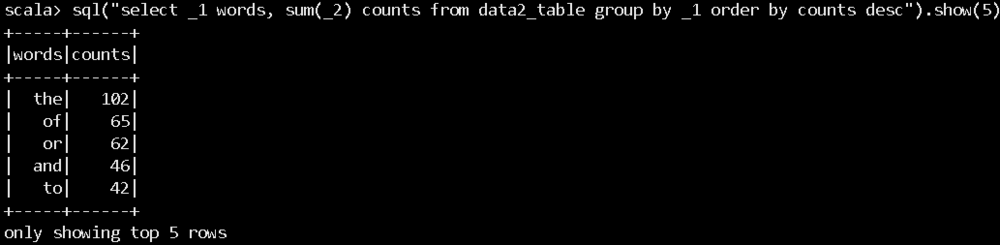
创建临时视图，进行SQL操作



使用SQL进行单词统计



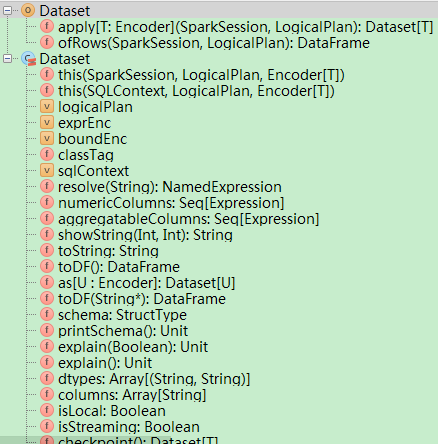
使用SQL进行排名分析



5. Dataset源码初探

如果不想本地搭建源码阅读环境，推荐一款在线阅读源码的工具[insight](https://link.jianshu.com?t=https://insight.io" \t "_blank)，不需要本地环境，可以直接引用github中的代码，非常方便.

Dataset的源码位于sql目录下：  
sql/core/src/main/scala/org/apache/spark/sql/Dataset.scala  
由此可以看出Dataset是Spark SQL组件中的东西，另外DataFrame也是SparkSQL中的东西（这也是为什么Spark SQL是Spark生态中发展最迅猛的模块）



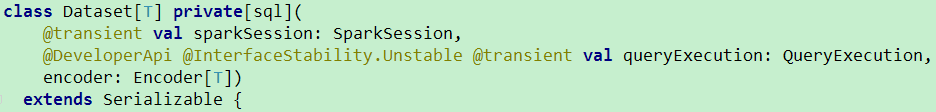
上图显示了Dataset文件的结构，可以看出：

Dataset是个类和一个伴生对象

里面包含了一些变量，比如我们常用的sqlContext

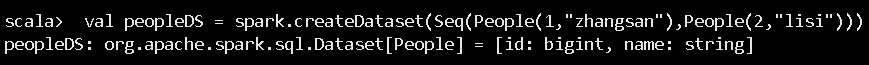
里面有很多函数和算子，比如toDF、map等操作数据的算子

前面我们说了，Dataset是个组织数据的的结构，那么数据存储在哪里呢？

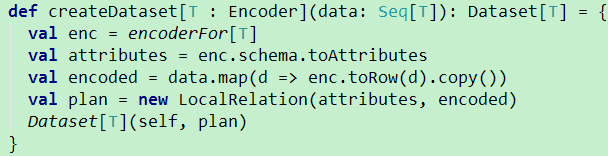


Dataset定义在sql这个包中

主构造函数中需要传递三个参数  
sparkSession：运行环境信息  
queryExecution：数据和执行逻辑信息。注意，数据在这个参数中  
encoder：编码器，用于将JVM对象转换为SparkSQL的对象（当然这里会有序列化和Schema等）



我们可以使用createDataset函数来创建一个Dataset，如上图所示。  
调用这个函数时，究竟发生了什么呢？



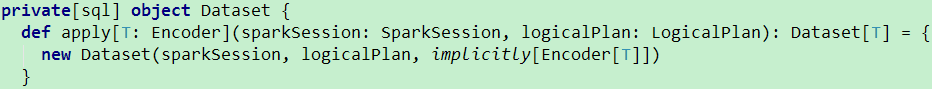
我们来看这个函数的实现：

将数据传进来后，提取数据的类型，通过类型提取Schema，并将数据转换成SparkSQL内部的Row（InternalRow）

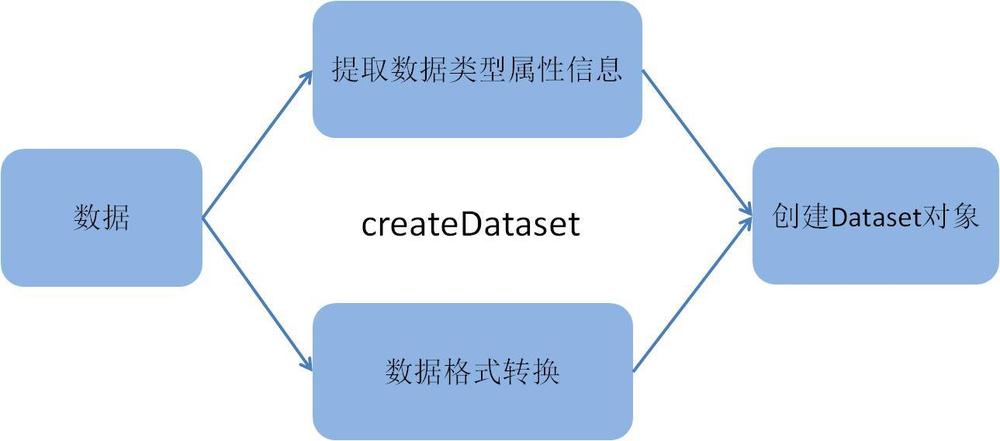
将数据和数据属性信息封装成一个relation对象

用Dataset的伴生对象创建Dataset实例，这里的self就是SparkSession对象

Dataset伴生对象中的apply方法来new Dataset生成对象，如下图所示



第三个参数没传，使用了隐式的encoder（createDataset中的encoded变量取名不规范，容易混淆）



一张简单的图，总结了创建Dataset两个最重要的步骤

到此为止我们在源码级别粗略的看了Dataset和创建流程，主要是帮助理解Dataset的概念，更多技术细节不适合本文描述。