

Apache Zeppelin使用入门指南:添加外部依赖

Apache Zeppelin使用入门指南:安装 Apache Zeppelin使用入门指南:编程

Apache Zeppelin使用入门指南:添加外部依赖

在前面的两篇文章中我们介绍了如何编译和部署Apache Zeppelin、如何使用Apache Zeppelin。这篇文章中将介绍如何将外部依赖库加入到Apache Zeppelin中。

在现实情况下,我们编写程序一般都是需要依赖外部的相关类库的,比如我们现在需要Spark读取Mysql里面的数据,所以我们需要添加Mysql相关的依赖。这里将介绍两种方法来添加外部依赖。

一、我们都知道,很多类库都是可以通过Maven仓库下载到的,所以我们可以指定依赖库的 group ID、artifact ID以及version来指定具体的依赖。在Zeppelin中,我们可以通过%dep Interpreter来加载依赖,如下:

%dep

z.load("mysgl:mysgl-connector-java:5.1.35")

%dep默认就可以使用,z代表的是Zeppelin context。

如果我们觉得Maven中央仓库下载的速度比较慢,我们可以自定义Maven仓库,如下:

%dep

z.addRepo("RepoName").url("RepoURL")

二、通过指定jar的本地路径来加载外部依赖,具体如下:

%dep

z.load("/path/to.jar")

在使用dep需要主要的是,我们必须在初始化SparkContext之前就得加载相关的外部类,否则将会出现异常。所以比较明智的方法是,在创建Notebook之后的第一步就是使用dep先加载相关的类库。



上面已经加载好了Mysql驱动,现在我们需要构建连接Mysql相关的参数信息,如下:

```
val props = scala.collection.mutable.Map[String,String]();
props+=("driver" -> "com.mysql.jdbc.Driver")
props+=("url" -> "jdbc:mysql://www.iteblog.com/scalada?user=root&password=orange123")
props+=("dbtable" -> "(select id, name, phone, email, gender from scalada.student) as student
s")
props+=("partitionColumn" -> "id")
props+=("lowerBound" -> "0")
props+=("upperBound" -> "100")
props+=("numPartitions" -> "2")
```

我们可以看出这是Spark连

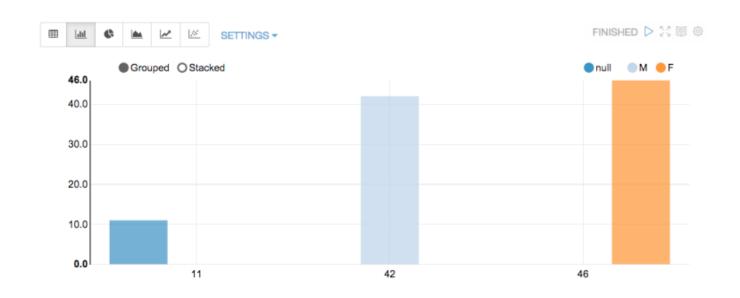
接Mysql的一种方式(更多的方式可以参见<u>《Spark读取数据库(Mysql)的四种方式讲解》</u>), 然后我们可以创建DataFrame了:

import scala.collection.JavaConverters._

val studentDf = sqlContext.load("jdbc", props.asJava)
studentDf.printSchema()
studentDf.show()
studentDf.registerTempTable("students")

```
import scala.collection.JavaConverters._
                                                                                    FINISHED ▷ ※ ■ ②
 val studentDf = sqlContext.load("jdbc", props.asJava)
  studentDf.printSchema()
  studentDf.show()
  studentDf.registerTempTable("students")
import scala.collection.JavaConverters._
warning: there were 1 deprecation warning(s); re-run with -deprecation for details
studentDf: org.apache.spark.sql.DataFrame = [id: string, name: string, phone: string, email: string, gender: s
tring]
I-- id: string (nullable = false)
I-- name: string (nullable = true)
 I-- phone: string (nullable = true)
I-- email: string (nullable = true)
 I-- gender: string (nullable = true)
lidl namel phonel
                              emaillgenderl
1101
    Mayal1-271-683-2698laccumsan.convalli...l
       Emil1-467-270-13371
                                                 FΙ
| 12| Caleb|1-683-212-0896|Suspendisse@Quisq...|
                                                 MΙ
|13|Florence|1-603-575-2444|sit.amet.dapibus@...|
```





这样将会输出相关的结果。至此,我们通过dep加载外部依赖库已经介绍完了。

本博客文章除特别声明,全部都是原创! 原创文章版权归过往记忆大数据(<u>过往记忆</u>)所有,未经许可不得转载。 本文链接:【】()