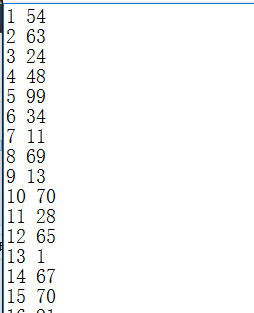
该案例中，我们将假设我们需要统计一个 1000 万人口的所有人的平均年龄，当然如果您想[**测试**](http://lib.csdn.net/base/softwaretest)Spark 对于[**大数据**](http://lib.csdn.net/base/spark)的处理能力，您可以把人口数放的更大，比如 1 亿人口，当然这个取决于测试所用集群的存储容量。假设这些年龄信息都存储在一个文件里，并且该文件的格式如下，第一列是 ID，第二列是年龄。

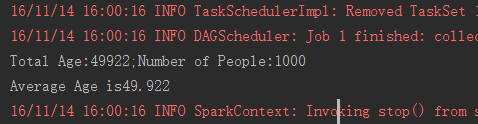
首先用scala生成数据：

**import** java.io.File  
**import** java.io.FileWriter  
**import** scala.util.Random  
  
*/\*\*  
 \* Created by 2281444815 on 2016/11/9.  
 \*/***object** DataMake {  
 **def** main(args: Array[String]) {  
 **val** write = **new** FileWriter(**new** File("D:\\peopledata.txt"),**false**)  
 **val** rand = **new** Random()  
 **for**(i <- 1 to 1000){  
 write.write(i + " " + rand.nextInt(100));  
 write.write(System.*getProperty*("line.separator"))//平台独立换行符  
 }  
 write.flush()  
 write.close()  
 }  
  
}

首先在我们本地运行该案例：

[root@hadoop-spark bin]# ./spark-submit --class AverageCount /root/data/SparkTest.jar hdfs://hadoop-spark:9000/spark/data/peopledata.txt

**import** org.apache.spark.{SparkConf, SparkContext}  
  
*/\*\*  
 \* Created by 2281444815 on 2016/11/9.  
 \*/***object** AverageCount {  
 **def** main(args: Array[String]): Unit = {  
 **val** logFile = "hdfs://hadoop-spark:9000/spark/data/peopledata.txt";  
 **if**(logFile.length < 1){  
 *println*("Usage:AvgAgeCalculator datafile")  
 System.*exit*(1)  
 }  
 **val** conf = **new** SparkConf().setAppName("SparkAverageCount").setMaster("local");  
 **val** sc = **new** SparkContext(conf);  
 **val** dataFile = sc.textFile(logFile,5);  
 **val** count = dataFile.count();  
 /\*  
flatMap与split截取字符串  
val str = "1,122,xxx,shandongyin"  
val file=sc.textFile(logFile)  
file.flatMap(line=>line.split(",")(3))  
上面代码本意是根据","分隔，取”3“位置上的字符，即”shandongyin"，但是实际取到的值是's'  
很郁闷吧。。。仔细一想与flatMap有关，”扁平化“有关  
解决方法：  
file.map(line=>line.split(",")(3))  
 \*/  
 **val** ageData = dataFile.map(line => line.split(" ")(1));//val ageData = dataFile.map(line => line.split(" ")(1))  
 **val** totalAge = ageData.map(age => Integer.*parseInt*(  
 String.*valueOf*(age))).collect().reduce((a,b) => a+b);  
 *println*("Total Age:" + totalAge + ";Number of People:"+count);  
 **val** avgAge :Double = totalAge.toDouble / count.toDouble;  
 *println*("Average Age is" + avgAge)  
 }  
  
}



第二种方式，在我们的Standalone集群中跑：

**import** org.apache.spark.{SparkConf, SparkContext}  
  
*/\*\*  
 \* Created by 2281444815 on 2016/11/9.  
 \*/***object** AverageCount {  
 **def** main(args: Array[String]): Unit = {  
 **if**(args.length < 1){  
 *println*("Usage:AvgAgeCalculator datafile")  
 System.*exit*(1)  
 }  
 **val** conf = **new** SparkConf().setAppName("SparkAverageCount");  
 **val** sc = **new** SparkContext(conf);  
 **val** dataFile = sc.textFile(args(0),5);  
 **val** count = dataFile.count();  
 /\*  
flatMap与split截取字符串  
val str = "1,122,xxx,shandongyin"  
val file=sc.textFile(logFile)  
file.flatMap(line=>line.split(",")(3))  
上面代码本意是根据","分隔，取”3“位置上的字符，即”shandongyin"，但是实际取到的值是's'  
很郁闷吧。。。仔细一想与flatMap有关，”扁平化“有关  
解决方法：  
file.map(line=>line.split(",")(3))  
 \*/  
 **val** ageData = dataFile.map(line => line.split(" ")(1));//val ageData = dataFile.map(line => line.split(" ")(1))  
 **val** totalAge = ageData.map(age => Integer.*parseInt*(  
 String.*valueOf*(age))).collect().reduce((a,b) => a+b);  
 *println*("Total Age:" + totalAge + ";Number of People:"+count);  
 **val** avgAge :Double = totalAge.toDouble / count.toDouble;  
 *println*("Average Age is" + avgAge)  
 }  
  
}

