import org.apache.spark.SparkContext

import org.apache.spark.SparkContext.\_

import org.apache.spark.SparkConf

class WordCount(argfile: String, argthreshold: String) extends java.io.Serializable{

// val sc = new SparkContext(new SparkConf().setAppName("Spark Count"))

val threshold = argthreshold.toInt

// split each document into words

val tokenized = sc.textFile(argfile).flatMap(\_.split(" "))

// count the occurrence of each word

val wordCounts = tokenized.map((\_, 1)).reduceByKey(\_ + \_)

// filter out words with less than threshold occurrences

val filtered = wordCounts.filter(\_.\_2 >= threshold)

// count characters

//val charCounts = filtered.flatMap(\_.\_1.toCharArray).map((\_, 1)).reduceByKey(\_ + \_)

//def calc(thetext: String): Unit = println(thetext + charCounts.collect().mkString(", "))

def calc(thetext: String): Unit = println(thetext + filtered.collect().mkString(", "))

}

val wcount =new WordCount("text.txt", "2")

//val wcount =new WordCount("hdfs://localhost:9900/user/zdina/test/text.txt", "2")

wcount.calc("Resultat")