# Spark Shell Programs (Scala)

## 1. Word Count

val lines = sc.parallelize(Seq("hello world", "hello spark"))  
val words = lines.flatMap(line => line.split(" "))  
val wordCounts = words.map(word => (word, 1)).reduceByKey(\_ + \_)  
wordCounts.collect().foreach(println)

## 2. Character Count

val text = sc.parallelize(Seq("spark shell"))  
val chars = text.flatMap(line => line.replaceAll(" ", "").toCharArray)  
val charCount = chars.map(char => (char, 1)).reduceByKey(\_ + \_)  
charCount.collect().foreach(println)

## 3. Vowel and Consonant Count

val text = sc.parallelize(Seq("spark shell example"))  
val vowels = Set('a', 'e', 'i', 'o', 'u')  
val characters = text.flatMap(\_.toLowerCase.replaceAll(" ", ""))  
  
val counts = characters.map(char => {  
 if (vowels.contains(char)) ("vowel", 1)  
 else if (char.isLetter) ("consonant", 1)  
 else ("other", 1)  
}).reduceByKey(\_ + \_)  
  
counts.collect().foreach(println)

## 4. Number Count

val text = sc.parallelize(Seq("abc123 456def 789"))  
val digits = text.flatMap(\_.toCharArray).filter(\_.isDigit)  
val numberCount = digits.map(d => ("digit", 1)).reduceByKey(\_ + \_)  
numberCount.collect().foreach(println)

## 5. Sorting Words Alphabetically

val lines = sc.parallelize(Seq("banana apple cherry"))  
val sortedWords = lines.flatMap(\_.split(" ")).sortBy(word => word)  
sortedWords.collect().foreach(println)