**ЛАБОРАТОРНАЯ РАБОТА 4.** Работа с пакетом SPARK

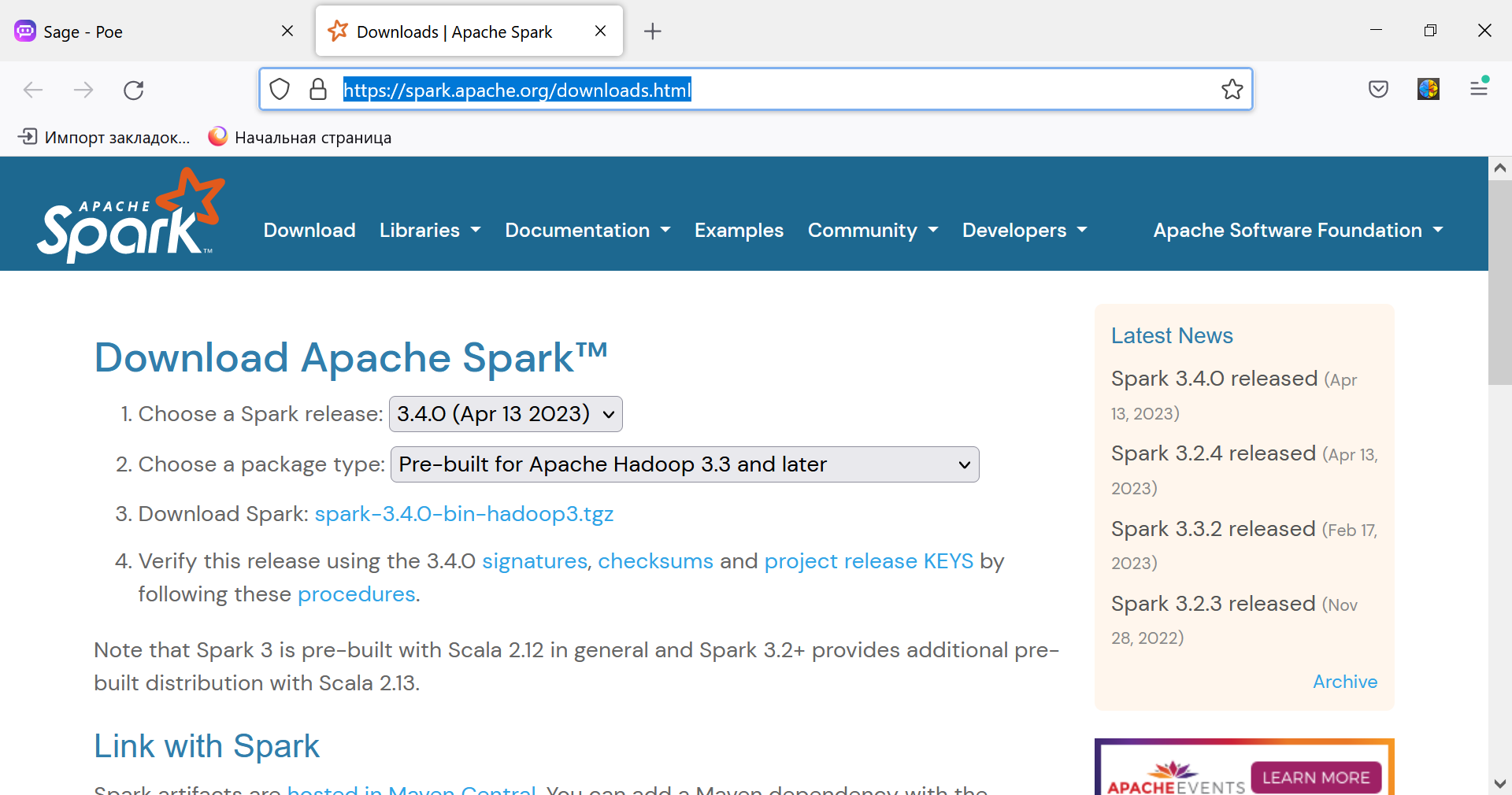
Цель. Изучить технику обработки текста в SPARK Scala.

**Краткие теоретические сведения**.

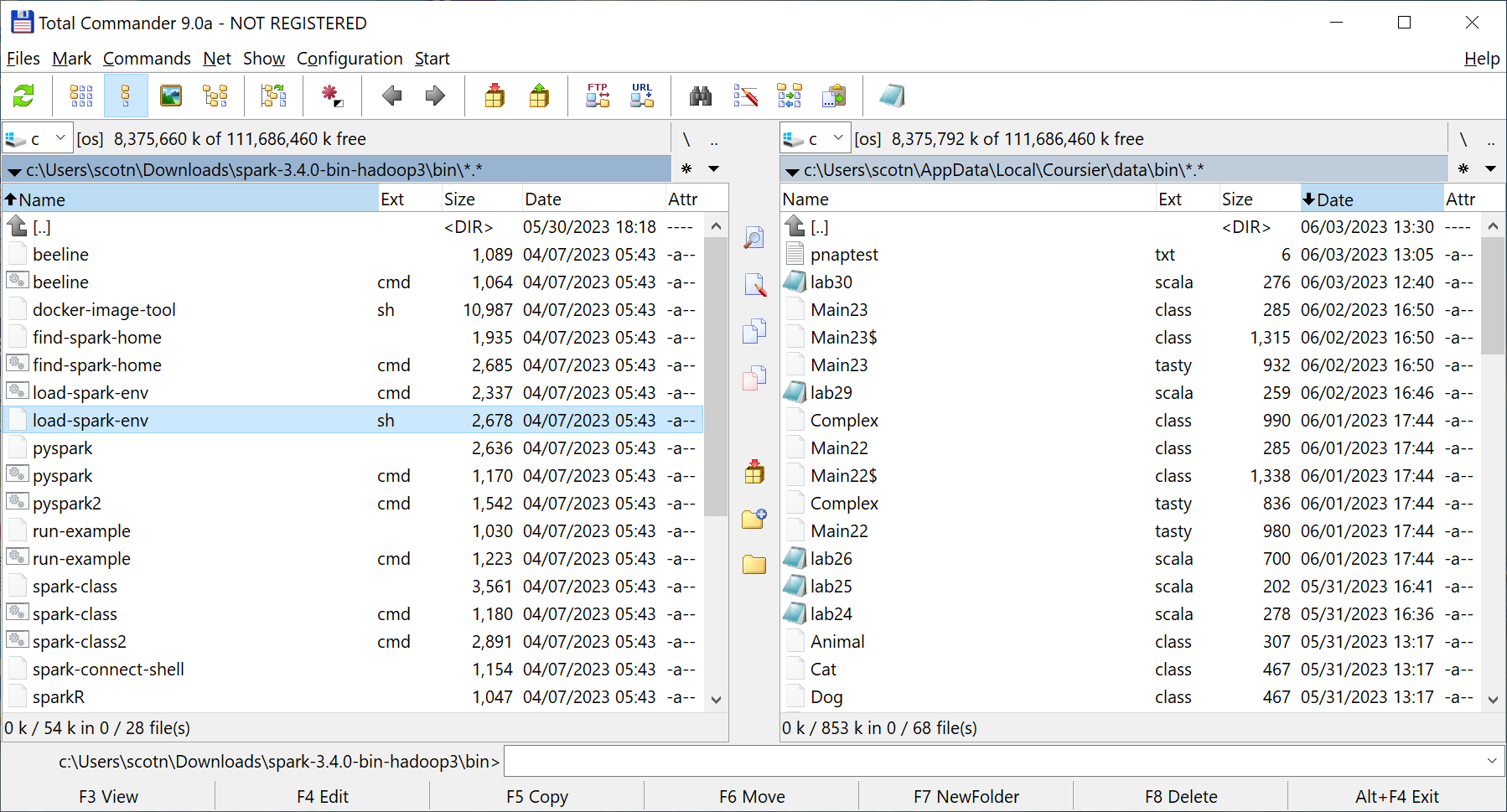
Будем использовать теоретический материал курса. Вот основные выдержки по SPARK.

Загрузим SPARK с сайта <https://spark.apache.org/downloads.html>

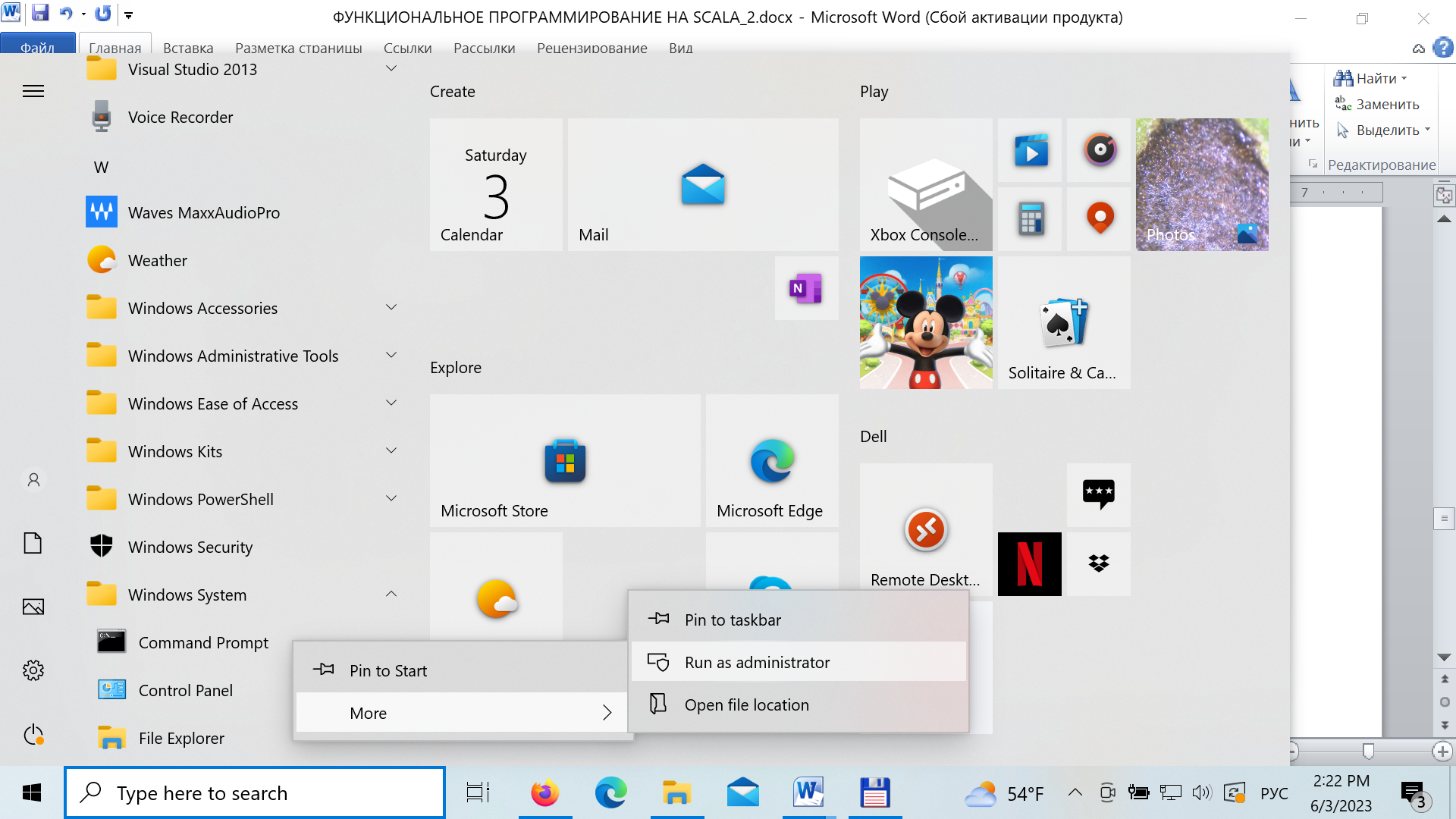
Вот его окно



Нужно распаковать загруженную библиотеку. На моей машине это



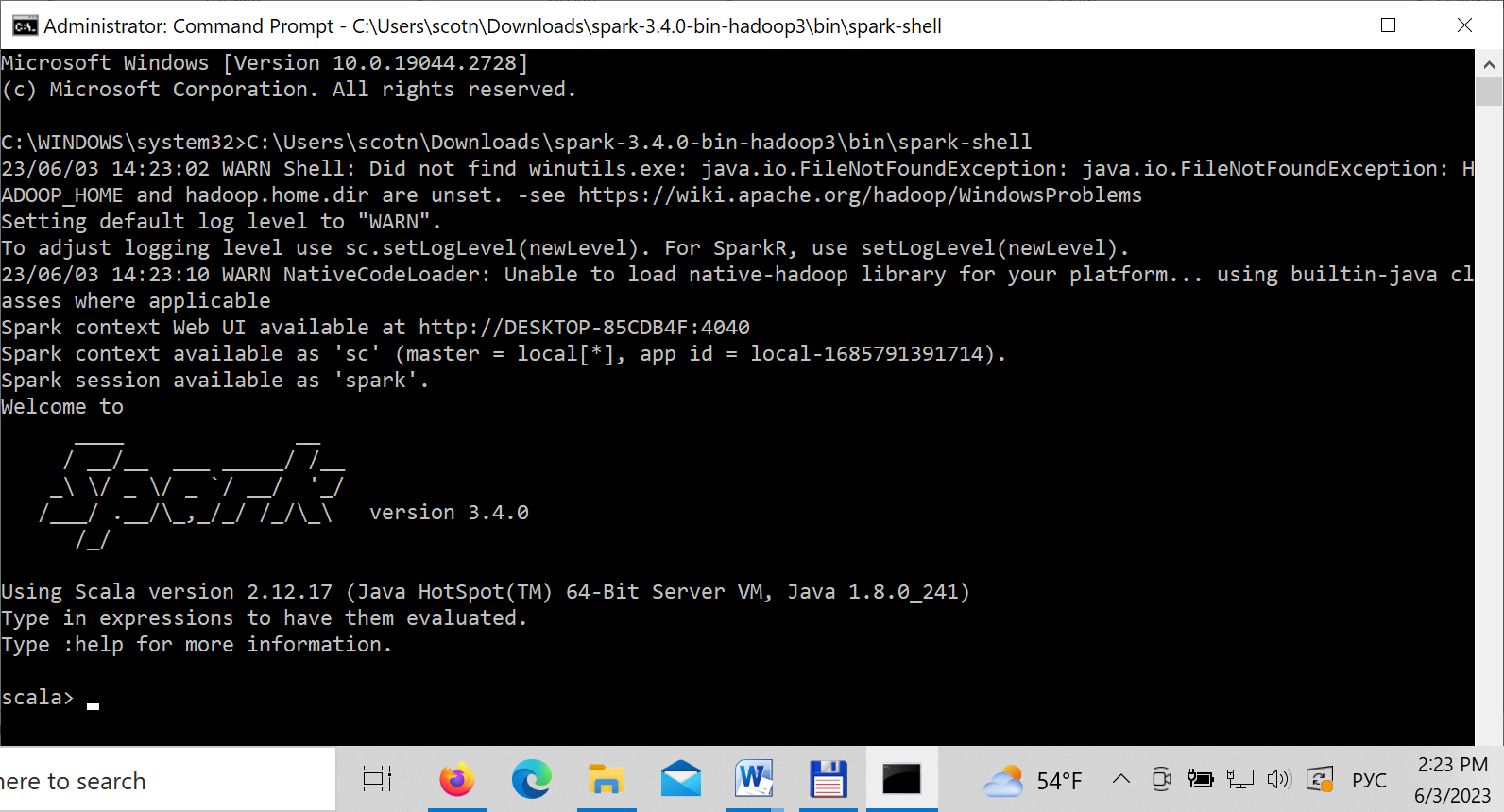
Чтобы запустить SPARK, нужно открыть командное окно в режиме администратора и ввести команду



**C:\Users\scotn\Downloads\spark-3.4.0-bin-hadoop3\bin\spark-shell**

**c:\Users\HP\AppData\Local\Coursier\data\bin\spark\bin\spark-shell**

Получите окно



Имея в распоряжении объект spark и sc, можно двигаться дальше.

Создаем журнал регистрации ошибок

scala> spark.sparkContext.setLogLevel("ERROR")

Это вообще-то нам не очень-то и нужно.

Далее создаем объект-конференции

scala> val conf = new org.apache.spark.SparkConf()

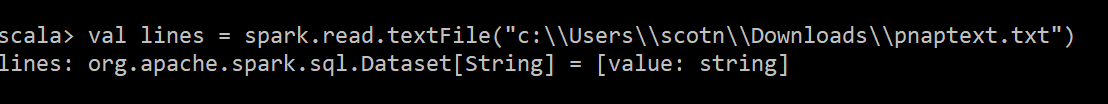
conf: org.apache.spark.SparkConf = [org.apache.spark.SparkConf@1e2a39b7](mailto:org.apache.spark.SparkConf@1e2a39b7)

Вот теперь мы читаем текстовый файл:

scala> val lines = spark.read.textFile("c:\\Users\\scotn\\Downloads\\pnaptext.txt")

Прописываем полный путь. Да еще дублируем симол-разделитель - \\

Строки прочтены

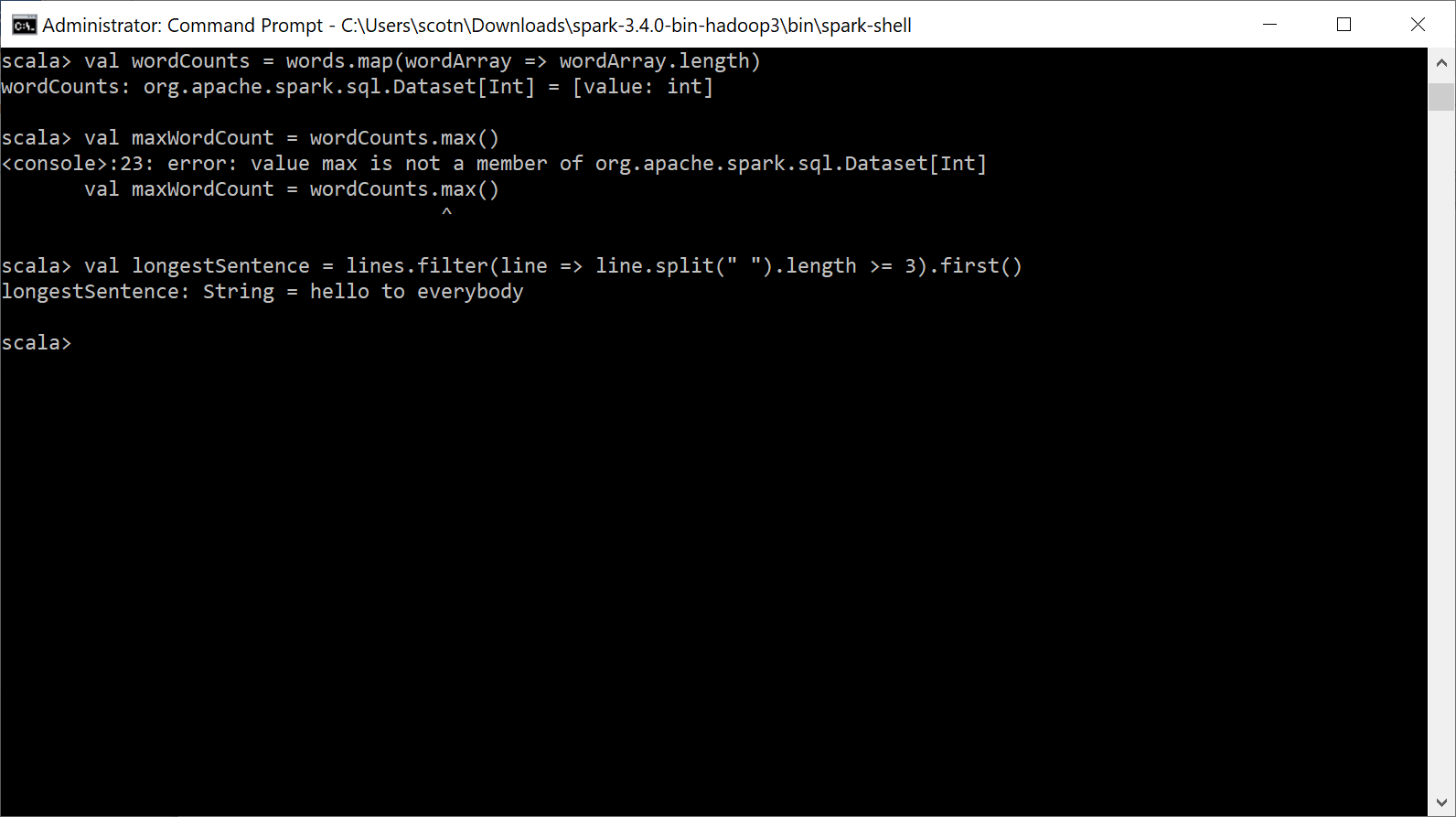


Получаем массив всех строк

scala> val words = lines.map(line => line.split(" "))

words: org.apache.spark.sql.Dataset[Array[String]] = [value: array<string>]

Выводим предложение, содержащее не менее трех слов



В нашем текстовом файле хранится такой текст

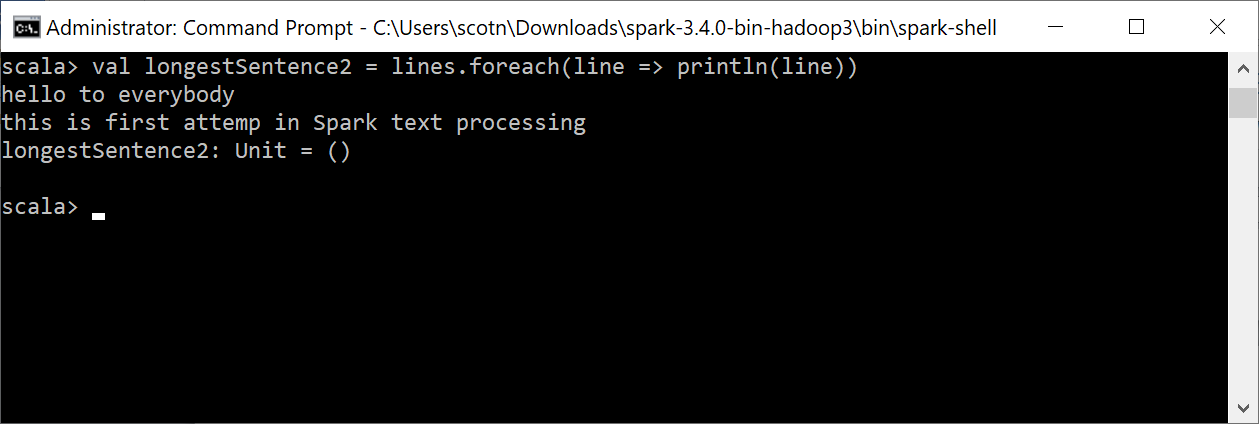
hello to everybody

this is first attemp in Spark text processing

Be careful with people No all of them sympthatize you

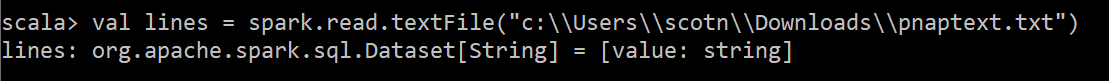
Выводим строки из текстового файла

val longestSentence2 = lines.foreach(line => println(line))

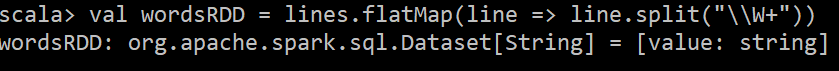


**Повторение.**

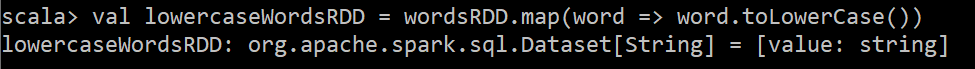
Начинаем с чтения текстового файла



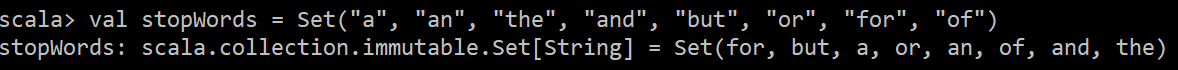
**Расщепляем текст на отдельные слова.** При этом используем регулярное выражение\\W+ (означает любую последовательность символов)

****

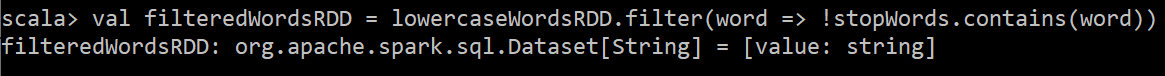
Слова приводим к нижнему регистру



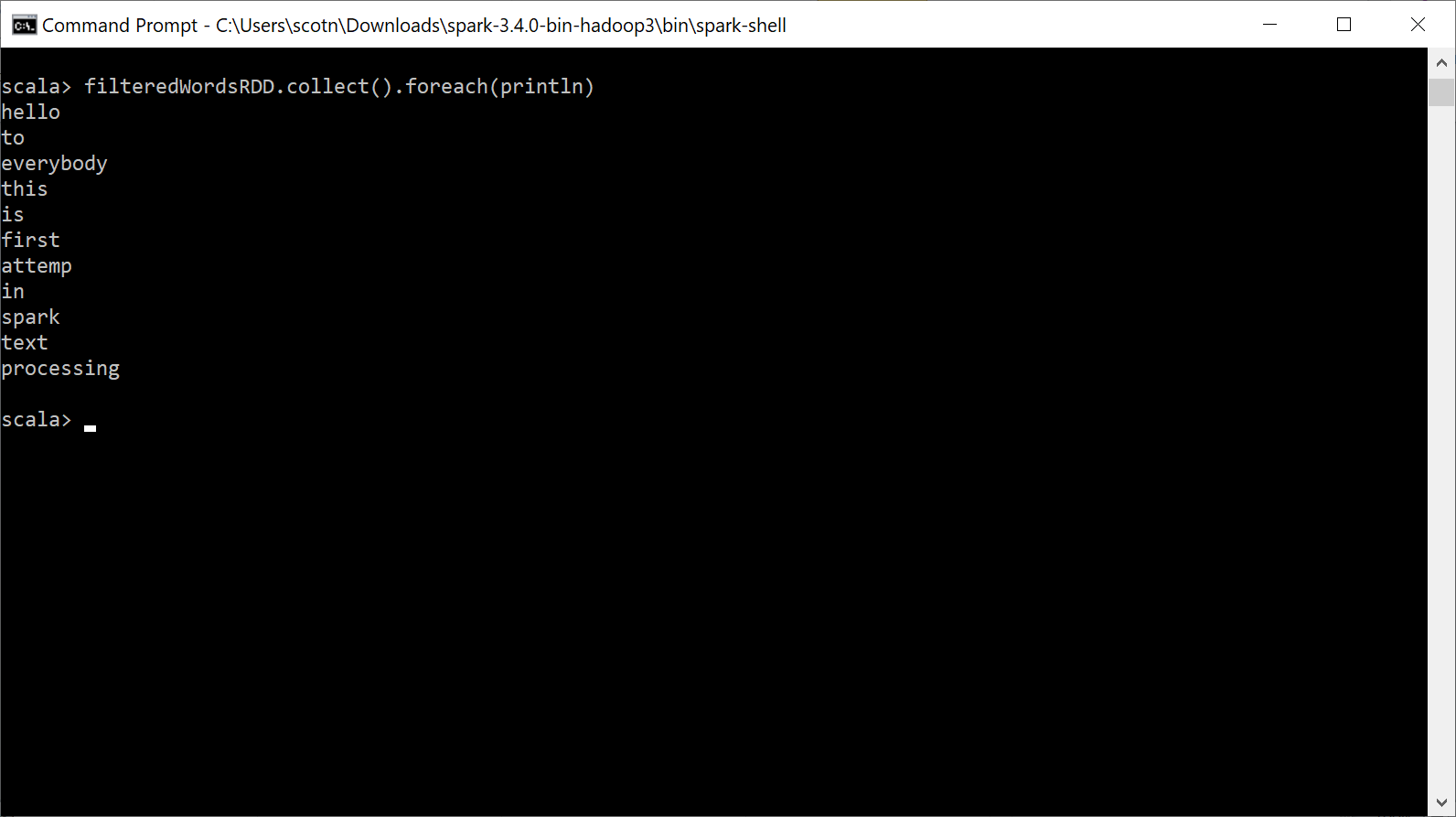
Создаем множество stop-слов

****

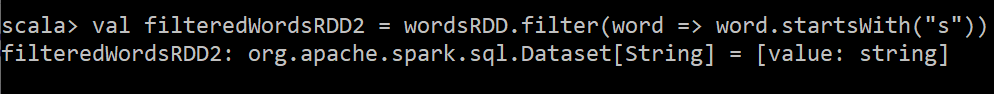
Удаляем стоп-слова из текста



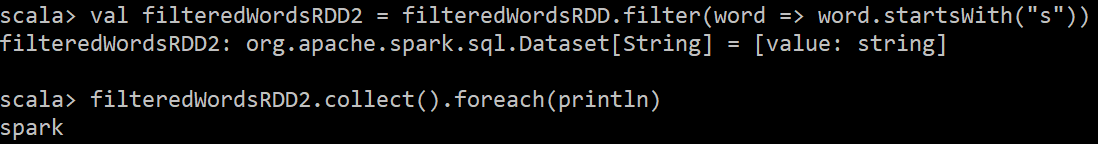
Выводим отфильтрованное множество слов



Отфильтруем слова, начинающиеся на s:



Выведем их



Итак, мы научились формировать множества слов и фильтровать их.

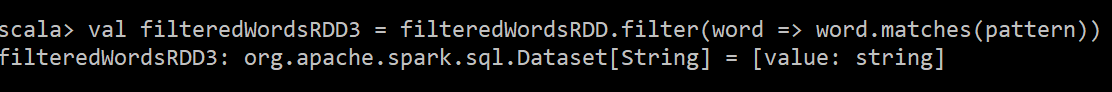
Познакомимся чуть ближе с регулярными выражениями

Создадим паттерн

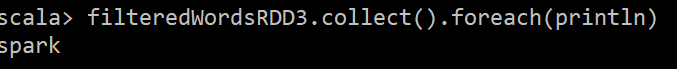


Этот паттерн определяет все слова, начинающиеся на s, содержащие где-то посередине r и заканчивающиеся на k

Сформируем отфильтрованное множество по паттерну



Выведем результат на консоль



***Порядок выполнения работы***

1. Изучить теоретическую часть.
2. Получить индивидуальное задание у преподавателя из прилагаемого списка.
3. Создать и отладить приложение в среде Python.
4. Написать отчет.
5. Защитить работу.

***Индивидуальные задания***

**Вариант 1.**

Создать собственный текстовый файл на английском или немецком языке – 4-5 предложений.

1. Вывести все слова из текстового файла, исключая stop-слова
2. Вывести все слова, содержащие букву t
3. Вывести все слова, заканчивающиеся на ing
4. Вывести все слова вторая буква которых а
5. Вывести все слова, последняя буква которых s
6. Вывести каждое второе слово

**Вариант 2.**

Создать собственный текстовый файл на английском или немецком языке – **4-5** предложений. Тема - программирование

1. Вывести все слова из текстового файла, исключая stop-слова
2. Вывести все слова, содержащие вхождение prog
3. Вывести все слова, заканчивающиеся на ion
4. Вывести все слова третья буква которых r
5. Вывести все слова, содержащие как минимум две буквы e
6. Вывести последнее слово

**Вариант 3.**

Создать собственный текстовый файл на английском или немецком языке – **4-5** предложений. Тема - спорт

1. Вывести все слова из текстового файла, исключая stop-слова
2. Вывести все слова, содержащие вхождение ball или sport
3. Вывести все слова, заканчивающиеся на ion
4. Вывести все слова третья буква которых c
5. Вывести все слова, длина которых больше 4
6. Вывести предпоследнее слово

**Вариант 4.**

Создать собственный текстовый файл на английском или немецком языке – **4-5** предложений. Тема - авто

1. Вывести все слова из текстового файла, исключая stop-слова
2. Вывести все слова, содержащие вхождение car или avto
3. Вывести все слова, заканчивающиеся на my
4. Вывести все слова третья буква которых d
5. Вывести все слова, у которых три и более гласных
6. Вывести второе слово

***Контрольные вопросы***

1. Что такое сокет?
2. Какую роль играет потоковый класс Thread в сетевом приложении?
3. Для чего используется функция accept?
4. Объясните в общих чертах порядок создания сетевого сокетного приложения.