University of Augsburg, Institute of Computer Science
Prof. Dr. P. Fischer
J. Kastner, V. Le Claire

SS 2020
April 29th, 2020
Blatt 1

Forschungsmodul Datenbanken und Informationssysteme

Hinweis *The goal of this sheet is to make yourself familiar with the basic abstractions in Spark: a) the idea of a framework with callbacks, b) RDDs as the original data model c) data frames as the refined data model for structured data. The last exercise is deliberately somewhat open-ended to allow for your different levels of previous knowledge. These tasks are meant to be executed on your local machine(s) and use (rather) small data. There is not an explicit deadline on finishing - we would like you to report back to us how well these exercises suit you by mid next week. If you chose to work with Java, a sample project (Maven, Eclipse) is provided.*

Aufgabe 1: Spark

a) In this exercise we want to *find and output duplicate files*. Given is a .csv-file *filehash.csv* (available in Digicampus). Each line of this file is a record that consists of <filename><md5hash>, separated by a comma. Find and report the names of duplicate files. You should report only **distinct file names**. Two files are duplicate if their md5hash values are equal.

   Solve this exercise with Spark's RDD API (documentation: Spark RDD).

b) In this exercise we want to implement *word count* using Spark's RDD. You can use the example from here as the starting point for your solution. The example *word count*-code has to be extended with the following features:

   - All words should be turned to lower case before being counted.
   - Sort your output by the word frequency in descending order (the most common words come at the beginning).
   - Before counting, remove stop words (e.g., "the", "a", ...), as they are very frequent, but have little relevance. The list of stop words should be entered directly in the code.
   - Count the number of words (without stop words). The number of words should be outputted separately.

Aufgabe 2: Spark RDDs

As a dataset for this exercise we use a part of the TPC-H benchmark. Now you should express some queries with Spark's RDDs.

a) Find the 25 suppliers with the lowest account balance.

b) How many suppliers do have a positive account balance?

c) Find all brands produced by the same manufacturer and calculate the number of items as well as the total sales price for each brand of each manufacturer.

d) How many items have 3 words in their name?

e) How many different items does each supplier have?

Can you express all requests with the RDDs? Do you think there is a better way to express these queries with Spark?

Aufgabe 3: Spark DataFrames (live)

In this Exercise you should get to know another concept of Spark - **DataFrames**. Solve the subtasks a)-e) from Exercise ! of this sheet with the DataFrames this time.

Please do consider the following aspects:

- different representations for the input data are possible in a Data Frame, e.g. Row, typed objects, ...
- generated query plans can be studied using the `explain(true)` statement.
- the number of cores used in local mode can be changed, e.g. `local`, `local[2]`, `local[*]`

Other useful resources are the Programming Guide and the API Reference.