



Exercise #9

- Word count problem
 - Input: (unstructured) textual file
 - Output: number of occurrences of each word appearing in the input file
- Solve the problem by using in-mapper combiners

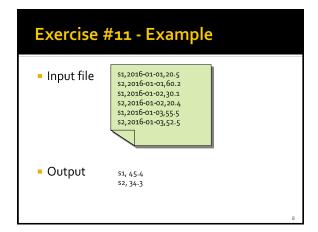
Input file Toy example file for Hadoop. Hadoop running example. Output pairs (toy, 1) (example, 2) (file, 1) (for, 1) (hadoop, 2) (running, 1)

Exercise #10

- Total count
 - Input: a collection of (structured) textual csv files containing the daily value of PM10 for a set of sensors
 - Each line of the files has the following format sensorId, date, PM10 value (μg/m³)\n
 - Output: total number of records

Exercise #11

- Average
 - Input: a collection of (structured) textual csv files containing the daily value of PM10 for a set of sensors
 - Each line of the files has the following format sensorId, date, PM10 value (μg/m³)\n
 - Output: report for each sensor the average value of PM10
 - Suppose the number of sensors is equal to 2 and their ids are s1 and s2



Exercise #12

- Select outliers
 - Input: a collection of (structured) textual files containing the daily value of PM10 for a set of sensors
 - Each line of the files has the following format sensorId, date\tPM10 value (\mu g/m³)\n
 - Output: the records with a PM10 value below a user provided threshold (the threshold is an argument of the program)

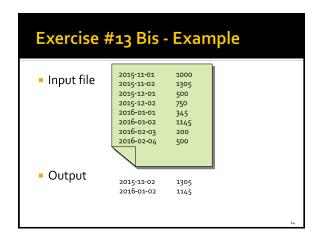
Exercise #12 - Example Input file s1,2016-01-01 52,2016-01-01 s1,2016-01-02 30.1 s2,2016-01-02 20.4 s1,2016-01-03 52,2016-01-03 52.5 Threshold: 21 Output s1,2016-01-01 20.5 52,2016-01-02

Exercise #13

- Top 1 most profitable date
 - Input: a (structured) textual csv files containing the daily income of a company
 - Each line of the files has the following format date\tdaily income\n
 - Output:
 - Select the date and income of the top 1 most profitable date
 - In case of tie, select the first date

Exercise #13 - Example 2015-11-01 1000 Input file 2015-11-02 1305 2015-12-01 500 2015-12-02 750 2016-01-01 2016-01-02 1145 2016-02-03 200 2016-02-04 500 Output 2015-11-02 1305

Top 2 most profitable dates Input: a (structured) textual csv files containing the daily income of a company Each line of the files has the following format date\tdaily income\n Output: Select the date and income of the top 2 most profitable dates In case of tie, select the first 2 dates among the ones associated with the highest income



Dictionary Input: a collection of news (textual files) Output: List of distinct words occurring in the collection

