NTNU

TDT4225 Store, distribuerte datamengder

Assignment #4

Øving 4

Medvirkende:
Brede Yabo Kristensen

March 13, 2019

NTNU
Kunnskap for en bedre verden

1 RDD API Tasks

1.1 Task 1

The files we are reading are rows with columns seperated by , .

I first split each column using , as a deliminator. This is done by using the map() function as it applies a function to all of the rows. The result is yet another RDD. When splitting the rows we get an array for each row, we then want the fourth column as it is the genre column. We could of used flatmap(), but we want to an rdd that has the same number of rows.

We then use the function distinct() to remove all duplicate from the rdd. Finally we count the number of rows using the count() function, which returns number.

1.2 Task 2

We start the same way as last time, only now we get the fourth column from the artist.csv file (year). We also make sure that the rows are converted to integers as we will compare them later. We then reduce the rdd using the reduce() function. We want to return the oldest artist and do this by comparing all the rows.

Finally we print the oldest artist birth date.

1.3 Task 3

Like before, we split the lines but now as a tupple with the integer 1, used for counting later. We then use reduceByKey() which executes the function for each rows that have the same key, in our case, we count each time and add it to the counter in the tupple. This leaves us with the number of artist for each country.

The task specifies that we sort the total number of artist for each country descending and artist alphabetically for countries with the same number of artists. This is done by first sorting alphabetically and then sorting for count.

When sorting the keys, we just flip the tupple and sortByKey (with false for descending sort) and the flip back. The result is outputted to $\text{result}_3 directory$

1.4 Task 4

We begin by mapping out artist$_i dcolumnandkeepinganumber 1inthetupple$.

1.5 Coalesce vs repartition