Abdullah Ghayumi

634072@student.inholland.nl

Parallel Distributed Processing (HADOOP)

Assignment 2

Assignment 2

# Details

Name: Abdullah Ghayumi

Student number: 634072

GitHub URL: https://github.com/Aras53/Hadoop

Table of Contents

[Details 1](#_Toc79910803)

[Problem 2](#_Toc79910804)

[Solution 3](#_Toc79910805)

[Uploading orders.csv 3](#_Toc79910806)

[Script 3](#_Toc79910807)

[Result 4](#_Toc79910808)

# Problem

• Make an alphabetic list from all locations from the orders.csv.

• Group by “location” with target “Holland”

• Count how many times Holland was the target from that location

• Code is executed from Pig View

• Upload a document on Moodle with the following information:

[• Name and student number](#_Details)

[• URL from GitHub with your source code](#_Details)

• Explain in the document what steps have to be taken to execute the code

• Explain always in your own words every step included source code.

• Make a screenshot from the result and include it in your document

# Solution

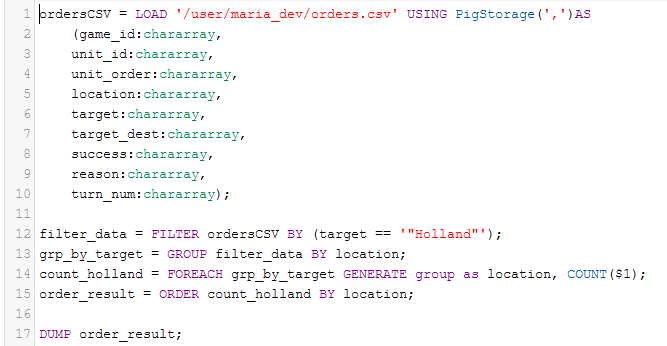
## Uploading orders.csv

The first step I took was to upload the file to the Files View, but I quickly found out that the file was too big and wouldn’t fully upload. To mitigate this, I tried uploading the file via MobaXTerm.

I uploaded the file first to the /home/maria\_dev location that I could access. Then I used the sudo root command to be able to access and execute command on the hdfs repo. I copied the orders file to the /users/maria\_dev folder with the following command:

hdfs dfs -put /home/maria\_dev/orders.csv /user/maria\_dev/

## Script



First, I filter the data, so that the only data that is left is the rows with the target “Holland”. After having filtered the data, I grouped the data by the location. This would result in a dataset with the following structure: {location: {rows}, location2: {rows}, etc.}

After having grouped the data, I counted the times Holland was the target from the location. To close it out I ordered the data by the location to get a list on alphabetical order.

## Result

