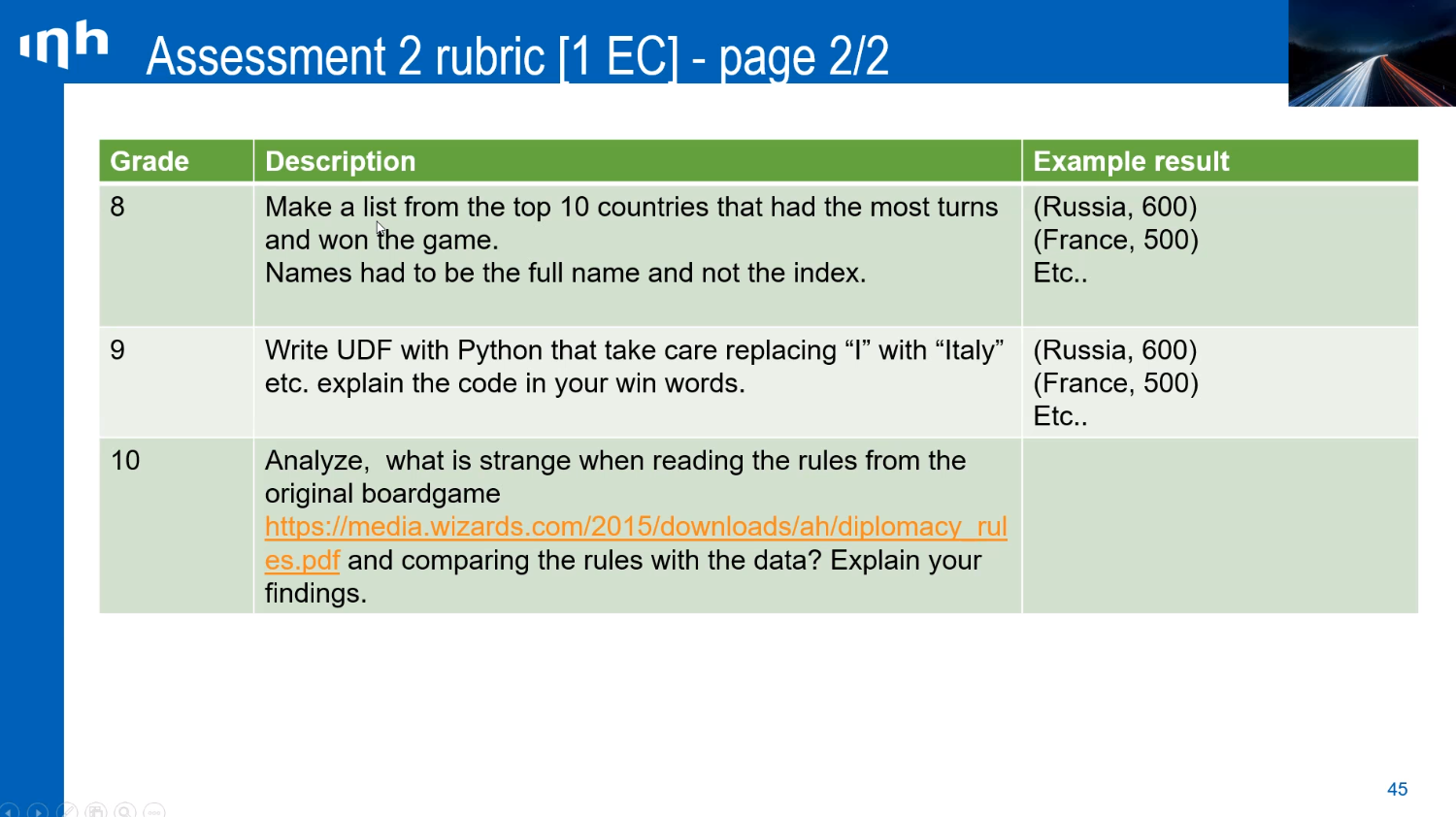
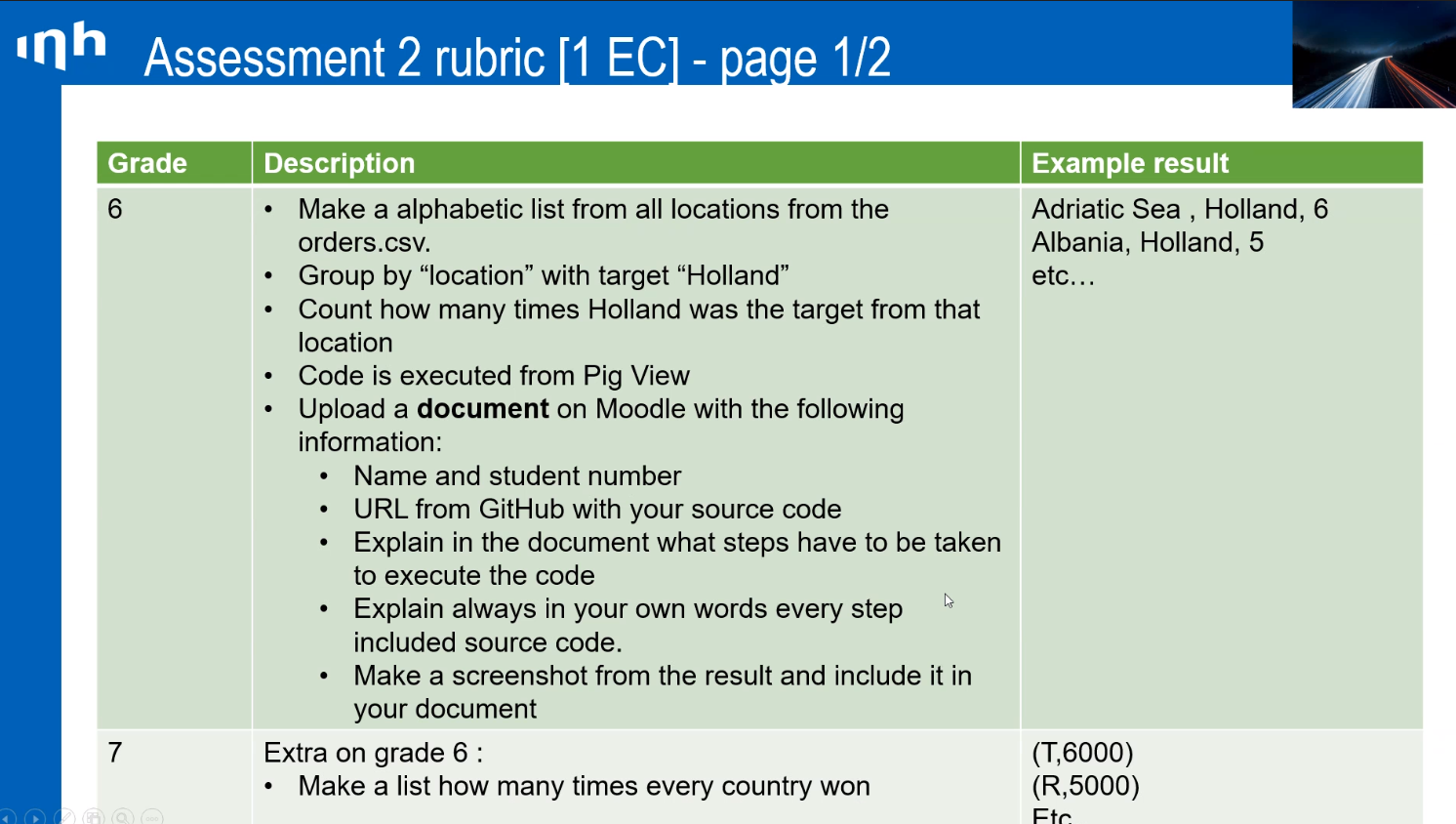
# Assignment 2



Prerequisities:

1. Installeer VirtualBox
2. Installeer HortonWorks sandbox versie 2.5.0
3. Importeer sandbox in VirtualBox
4. Download Diplomacy dataset
5. Start sandbox in VirtualBox

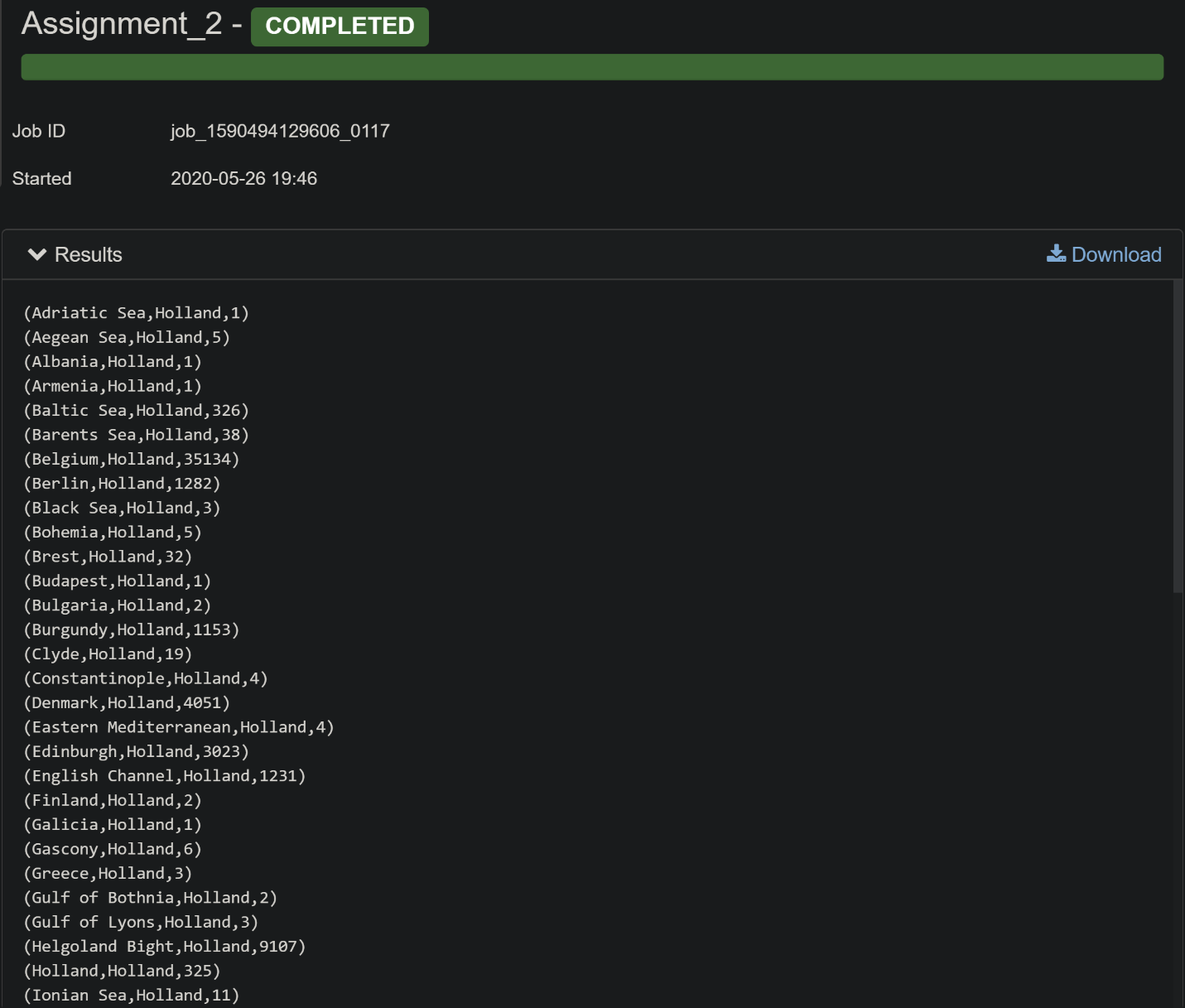
Ondernomen stappen:

1. Surf in de browser naar 127.0.0.1:8080
2. Log in met gebruikersnaam maria\_dev en wachtwoord maria\_dev
3. Klik op het rastermenu rechtsboven in de hoofdbalk
4. Klik op ‘Files view’
5. Maak een map aan genaamd ‘Diplomacy’ in /user/maria\_dev
6. Upload de datasets in de aangemaakte map
7. Klik op het rastermenu rechtsboven in de hoofdbalk
8. Klik op ‘Pig view’
9. Maak een nieuw script aan met de naam ‘Assignment\_2’
10. *Script schrijven (zie* [*https://www.github.com/LennartdeWaart/Hadoop/Assignment\_2*](https://www.github.com/LennartdeWaart/Hadoop/Assignment_2)*)*
11. Run het script met TEZ

*Uitleg code in script. Zie onderstaande URL:* <https://github.com/LennartdeWaart/Hadoop/blob/master/Assignment_2/script.txt>

**Resultaat**

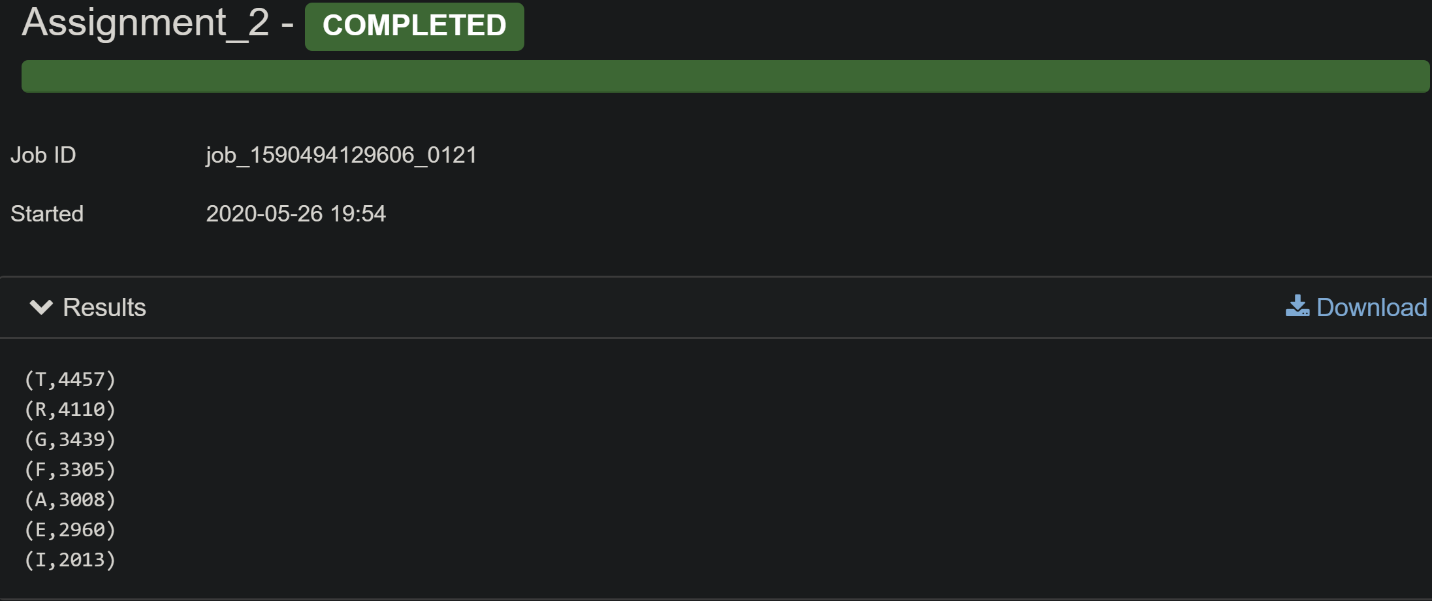
*6-opdracht*

De orders dataset wordt ingeladen met behulp van CSVLoader. De data wordt vervolgens gefilterd waar het target ‘Holland’ is. De gefilterde dataset wordt gegroepeerd op ‘location’ en voor iedere groep wordt een tuple met de locatie, het target en hoe vaak de locatie het target heeft aangevallen toegevoegd aan een nieuwe ‘bag’-variabele. Deze variabele wordt vervolgens geordend op alfabetische volgorde en uitgeprint. Het resultaat is als volgt:

**Figuur 1: Resultaat 6-opdracht**

*7-opdracht*

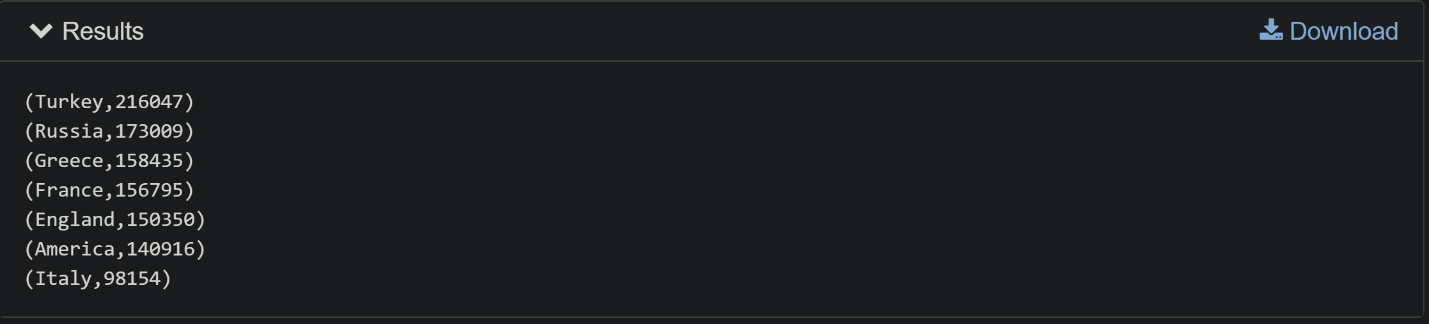
De players dataset wordt ingeladen met behulp van CSVLoader. De data wordt vervolgens gefilterd op spelers die gewonnen hebben. De gefilterde dataset wordt gegroepeerd op land en voor iedere groep wordt een tuple met het land en hoe vaak deze gewonnen heeft aan een nieuwe ‘bag’-variabele. Deze variabele wordt vervolgens aflopend geordend op basis van het winstaantal en uitgeprint. Het resultaat is als volgt:



**Figuur 2: Resultaat 7-opdracht**

*8-opdracht*

De games en players datasets worden ingeladen met behulp van CSVLoader. De data wordt vervolgens

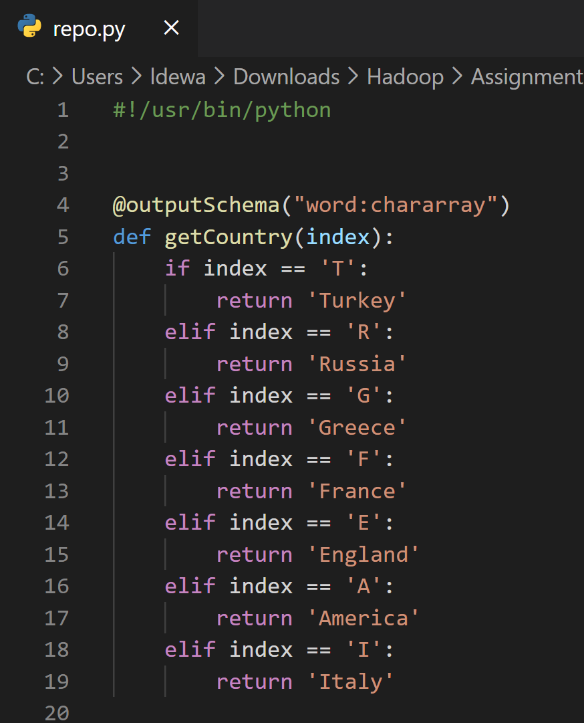
Samengevoegd op basis van de id kolom in de games dataset en de gmae\_id kolom in de players dataset. Deze relatie wordt gefilterd op winst. De gefilterde relatie wordt vervolgens gegroepeerd op land en voor iedere groep wordt een tuple met de naam van het land (gebaseerd op een geneste conditionele voorwaarde) en de som van het aantal gemaakte zetten van dit land toegevoegd aan een nieuwe ‘bag’-variabele. Deze variabele wordt vervolgens aflopend geordend op basis van het aantal zetten. De geordende data wordt gelimiteerd tot de eerste 10 rijen en daarna uitgeprint. Het resultaat is als volgt:

**Figuur 3: Resultaat 8-opdracht**

*9-opdracht*

Er wordt een Python UDF geschreven die input koppelt aan land en de naam van dit land teruggeeft.

Zie ook <https://github.com/LennartdeWaart/Hadoop/blob/master/Assignment_2/repo.py>.

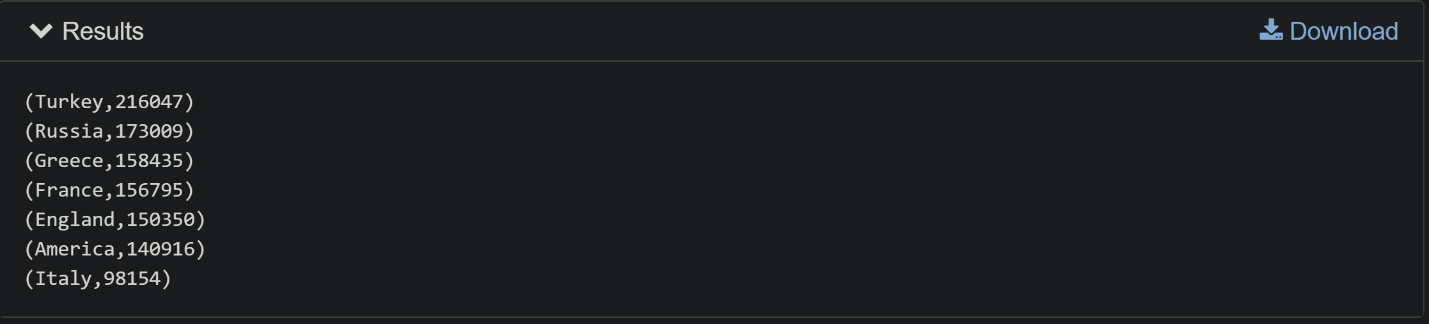


**Figuur 4: Python UDF**

Het Python-script wordt vervolgens geüpload in de eerder aangemaakte map ‘Diplomacy’. De CLASSPATH wordt op het begin van het script geregistreerd zodat deze kan worden aangeroepen in de code.

De games en players datasets worden ingeladen met behulp van CSVLoader. De data wordt vervolgens

Samengevoegd op basis van de id kolom in de games dataset en de gmae\_id kolom in de players dataset. Deze relatie wordt gefilterd op winst. De gefilterde relatie wordt vervolgens gegroepeerd op land en voor iedere groep wordt een tuple met de naam van het land (door de UDF in het Python-script aan te roepen) en de som van het aantal gemaakte zetten van dit land toegevoegd aan een nieuwe ‘bag’-variabele. Deze variabele wordt vervolgens aflopend geordend op basis van het aantal zetten. De geordende data wordt gelimiteerd tot de eerste 10 rijen en daarna uitgeprint. Het resultaat is als volgt:



**Figuur 5: Resultaat 9-opdracht**