Running your own Word Count Program

Prepared By : Siddhartha Shakya

Before completing the below tasks make sure that you have :

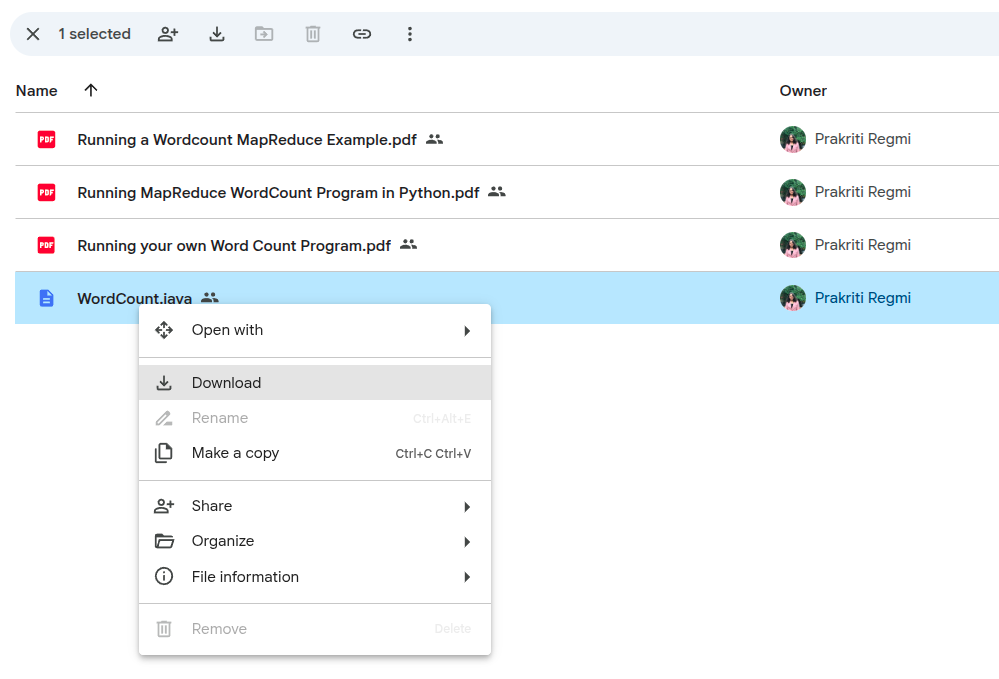
* Switched to the **hadoop user**.
* Navigated to the hadoop **Sbin** directory.
* Started the **Hadoop Distributed File System (HDFS).**
* Started the **YARN Resource Manager**.

**If not** refer to the below image to complete the above tasks :

|  |
| --- |
|  |

If **jps** **(Java Processing Status)** lists all the above **running java processes**, then you are **good to go.**

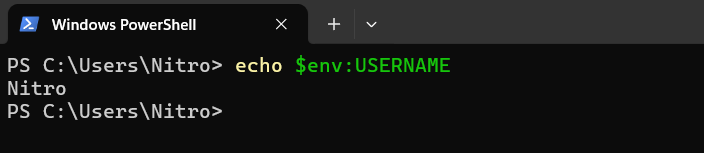
1. Download **WordCount.java** file from the drive.



*Note :*Make sure that the **java file** you downloaded is in **Downloads.**

1. In **windows powershell** type in the following command.

|  |
| --- |
| > echo $env:USERNAME |



Take note of the output, **mine is Nitro** yours **could be different**.

A black background with white text

Description automatically generated

1. Move your **WordCount.java** file to your **WSL home directory**.

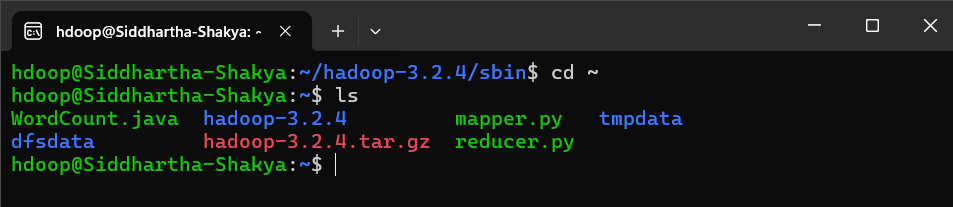
|  |
| --- |
| ~$ cp /mnt/c/Users/**YourUser**/Downloads/WordCount.java ~/ |

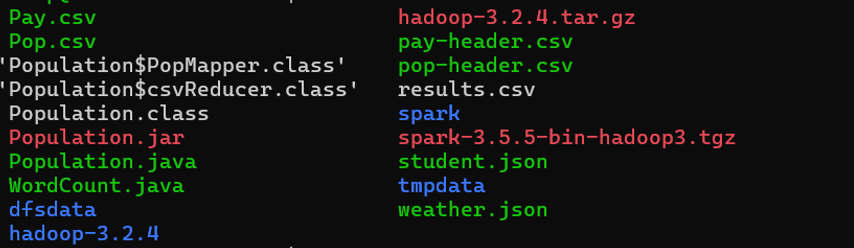
*Important:* Don’t forget to replace **“YourUser”** with the **output from the above step 2**.

**

1. Check if the **WordCount.java file** has been **successfully copied** to your **WSL home directory.**

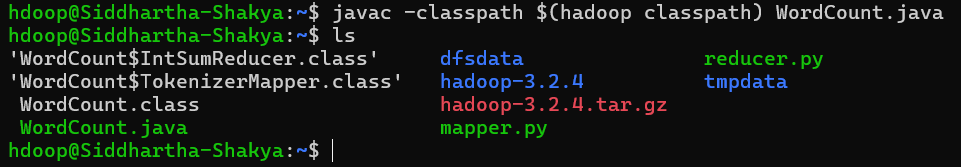
|  |
| --- |
| ~$ cd ~  ~$ ls |

****

****

1. Compile the file :

|  |
| --- |
| ~$ javac -classpath $(hadoop classpath) WordCount.java |

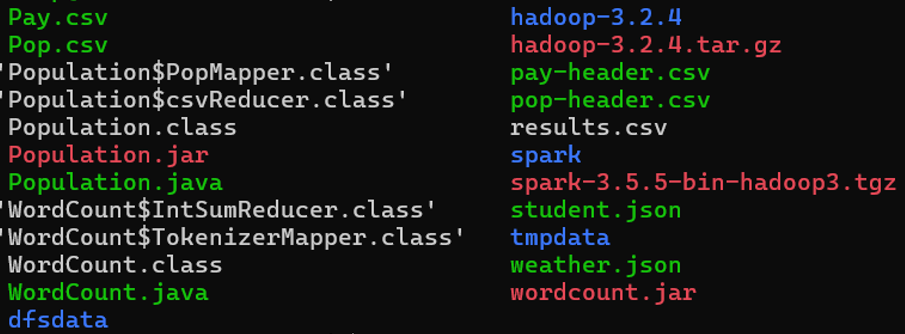


A screen shot of a computer screen

Description automatically generated

1. Produce the Jar file :

|  |
| --- |
| ~$ jar cf wordcount.jar Word\*.class |



1. Create the input directory on the hdfs :

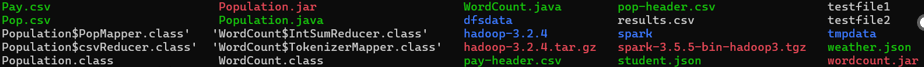
|  |
| --- |
| ~$ hdfs dfs -mkdir /input\_word  ~$ hdfs dfs -ls / |

A black screen with white text

Description automatically generated

1. Create the input files :

|  |
| --- |
| ~$ echo A long time ago in a galaxy far far away > testfile1  ~$ echo Another episode of Star Wars > testfile2 |

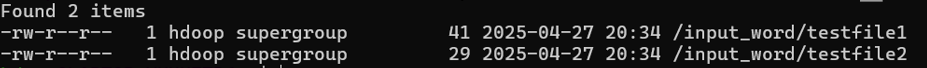


1. Save the files to the input directory :

|  |
| --- |
| ~$ hdfs dfs -put testfile? /input\_word |

1. Verify the uploads by checking if the files are inside the input\_word directory.

|  |
| --- |
| ~$ hdfs dfs -ls /input\_word |

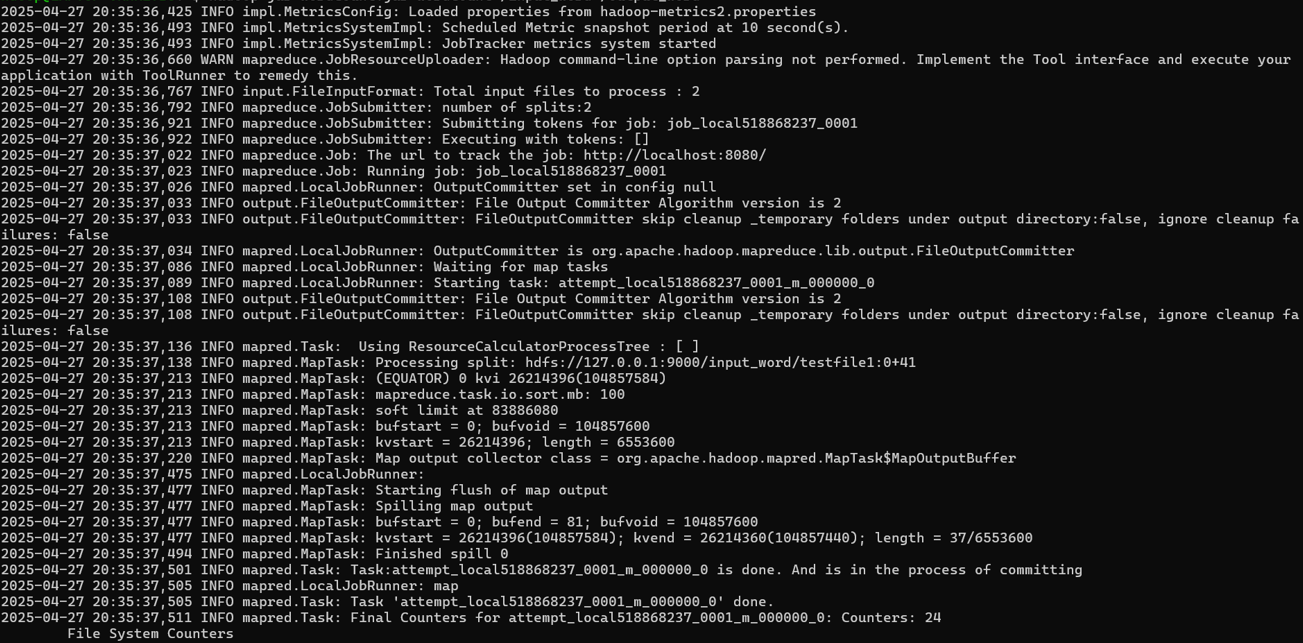


1. It is important that the output directory does not already exist. If you have run the program before you need to delete the previous output directory :

|  |
| --- |
| ~$ hdfs dfs -rm -R /output\_word |

1. Run the Map Reduce program :

|  |
| --- |
| ~$ hadoop jar wordcount.jar WordCount /input\_word /output\_word |

­

A screenshot of a computer program

Description automatically generated A screenshot of a computer program

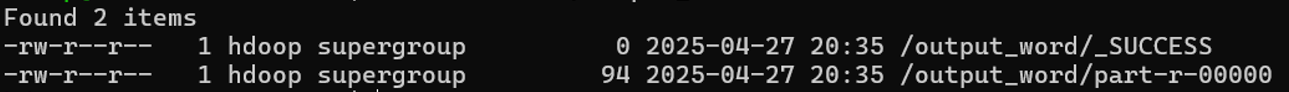
Description automatically generated A screen shot of a computer screen

Description automatically generated A screenshot of a computer

Description automatically generated

1. Check what files are in the output directory :

|  |
| --- |
| ~$ hdfs dfs -ls /output\_word |



1. See what is in the output file:

|  |
| --- |
| ~$ hdfs dfs -cat /output\_word/part-r-00000 |

A computer screen shot of a code

Description automatically generated

A screenshot of a computer screen

Description automatically generated