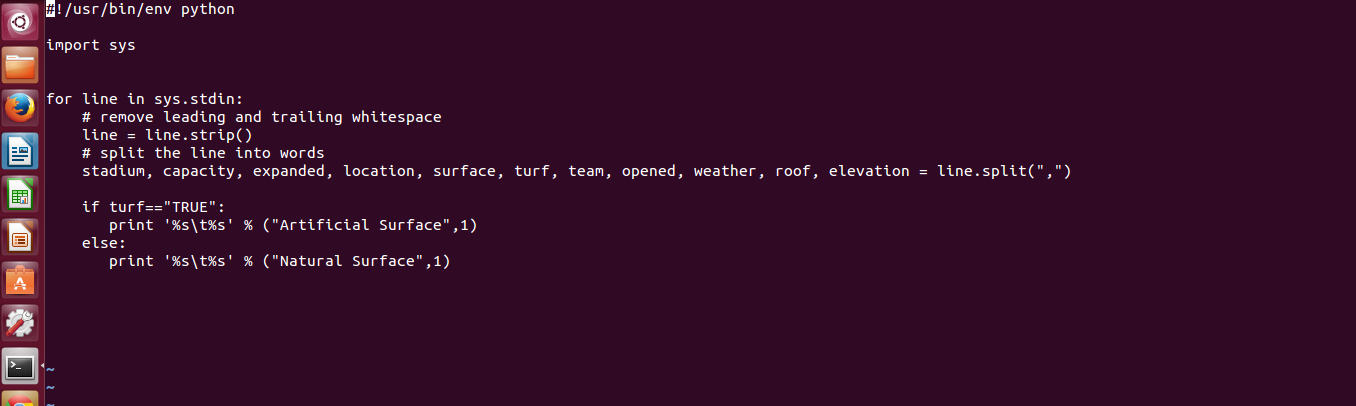
**Project 1**

Steps

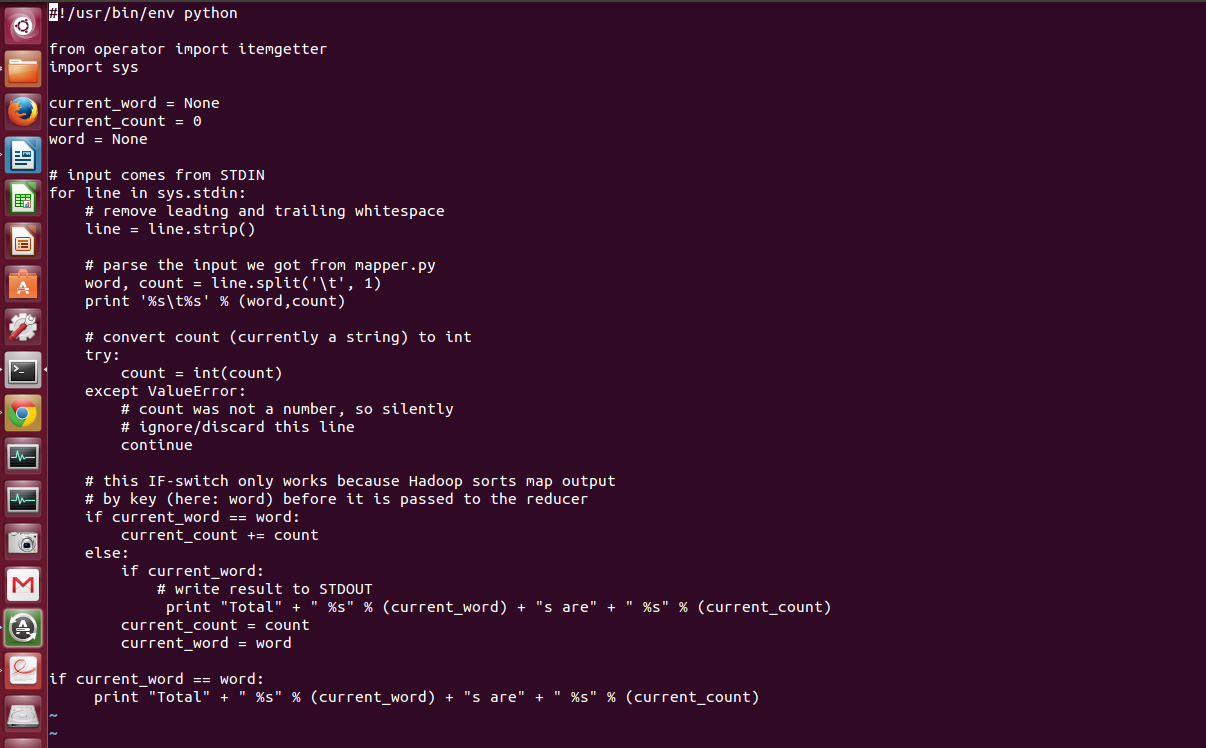
1. Mapper file is created at location /home/dyj924/Hadoop/mapperV2.py in which key-value pair is created with “Turf”



For loop will run for each line and will split the words on the basis of comma. This file will assign number to each turf, whether its artificial or natural.

2. Reducer File is created at location /home/dyj924/Hadoop/reducerV2.py. Before Execution of this file, we have

a list of keys and values which detrmine the type of turf as a key and its corresponding value.

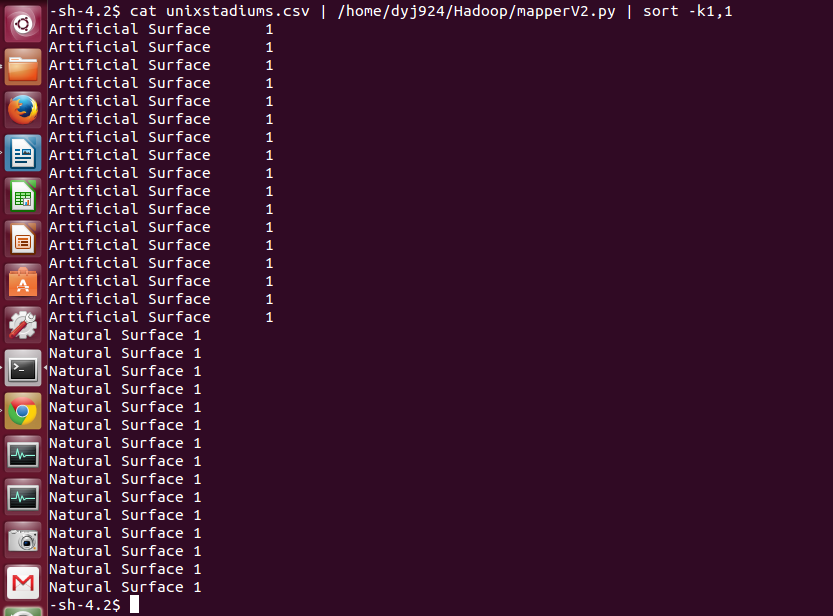


Loop will run the file line by line and this time words will be split by “single space”.Thus, word variable will have type of the turf and count variable will have corresponding value.

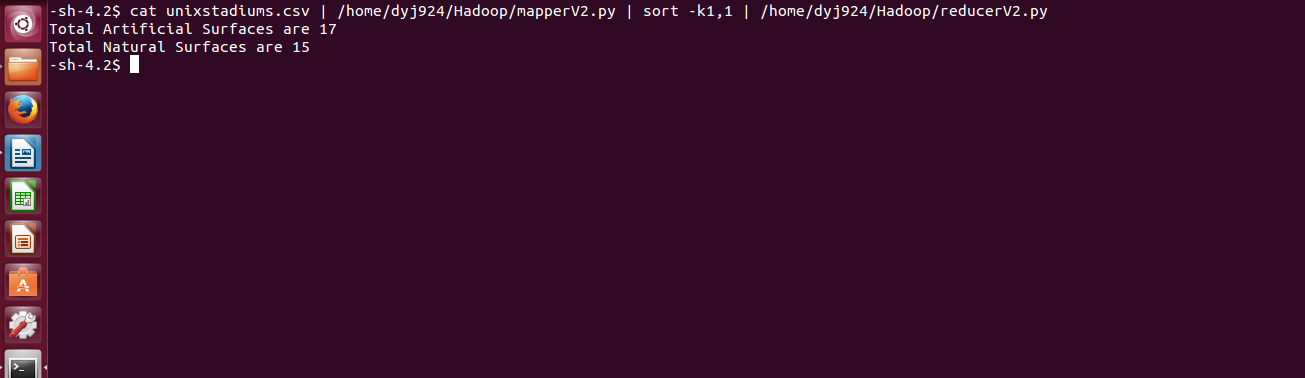
Try block is added, to convert the string value of count to int ( as we are supposed to add these counts further).

The logic is written such that, if the loop detects the same word, instead of assigning the new count value, the value is added in previous count, giving us the total number of counts for each type turf.

3. The program is initially tested on local machine with following command:



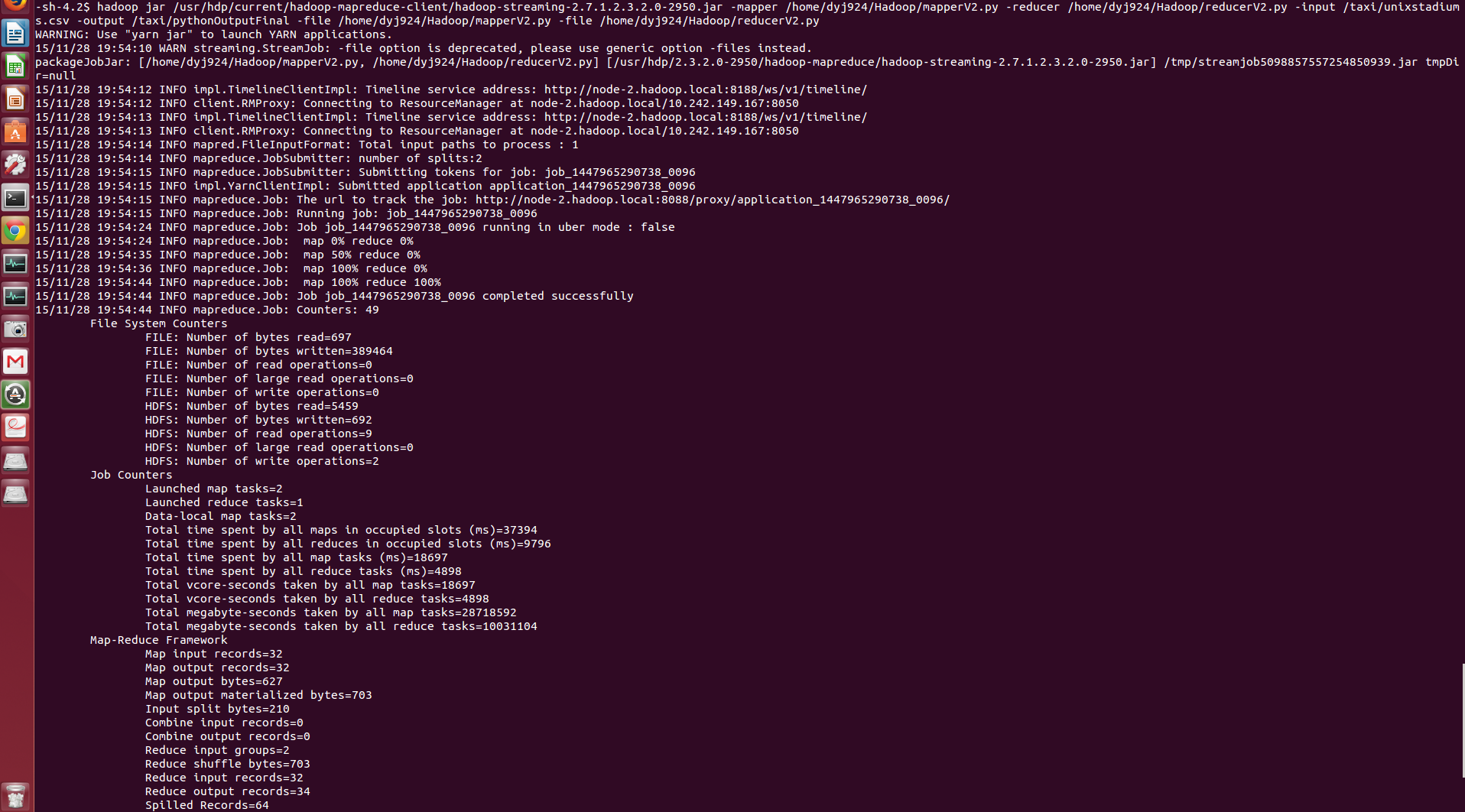
Initially command runs the mapper on unixstadiums.csv which prints the type of surface with the count and output is sorted alphabetically.



Once, the reducer is applied on the previous output, it gives the total count of natural and artificial surfaces.

As successful testing on local machine is done, now it is time to run the code on hadoop cluster.

4. Following command is executed to run the Map-Reduce program on cluster



This command creates two files inside directory dyj924Output, one of which is \_SUCCESS and other one is part-00000 hich shows the desired output.

The first parameter gives the path of the streaming jar file.

Mapper tag denotes the mapper file while reducer tag denotes the reducer file.

Input file is a file on which we are supposed to run mapper and reducer and in output file we store the output of the result.

In file tag, we enter a mapper file once and then reducer file.



