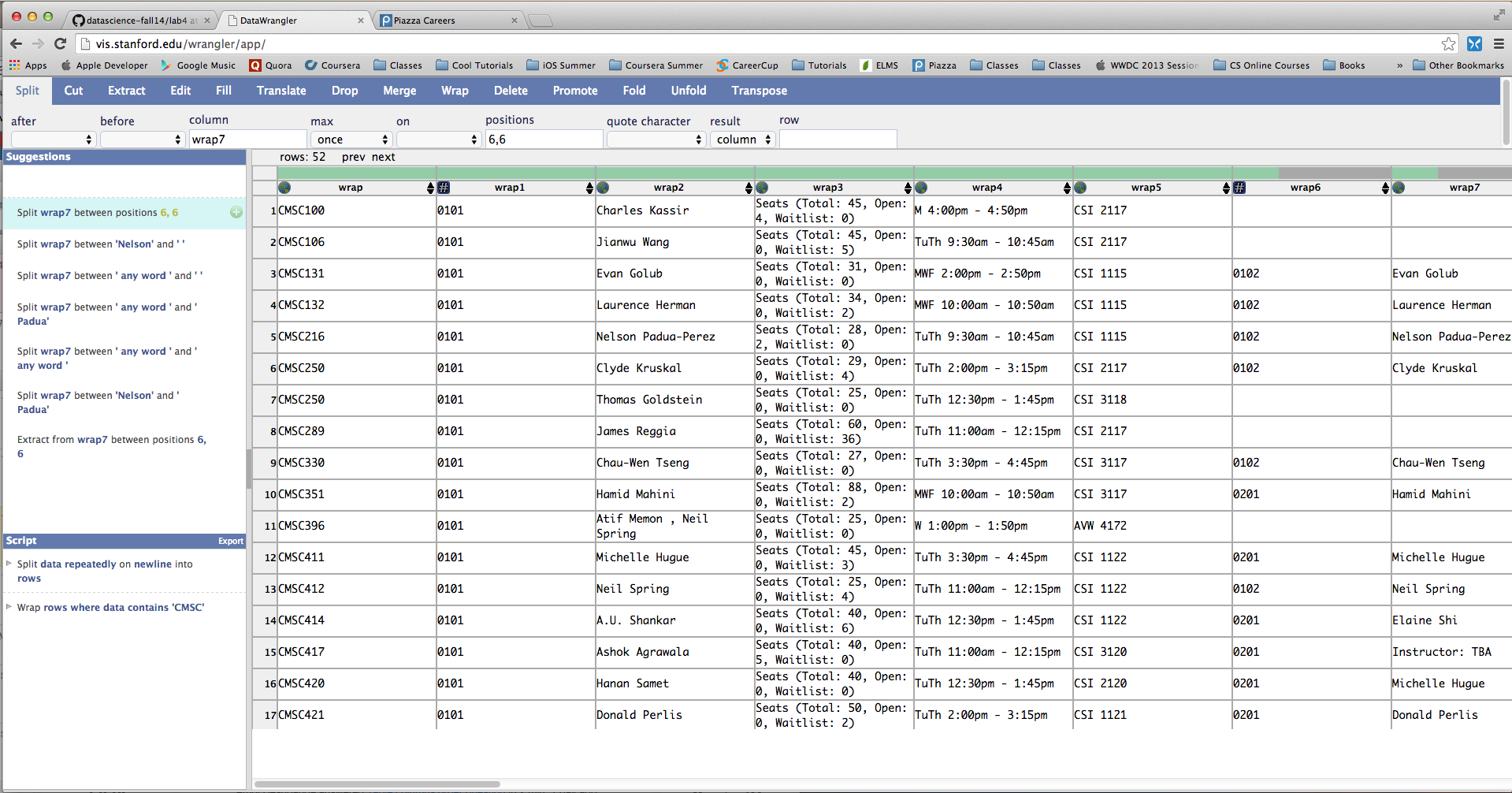
**Lab 4 Submission – Nadeem Malik**

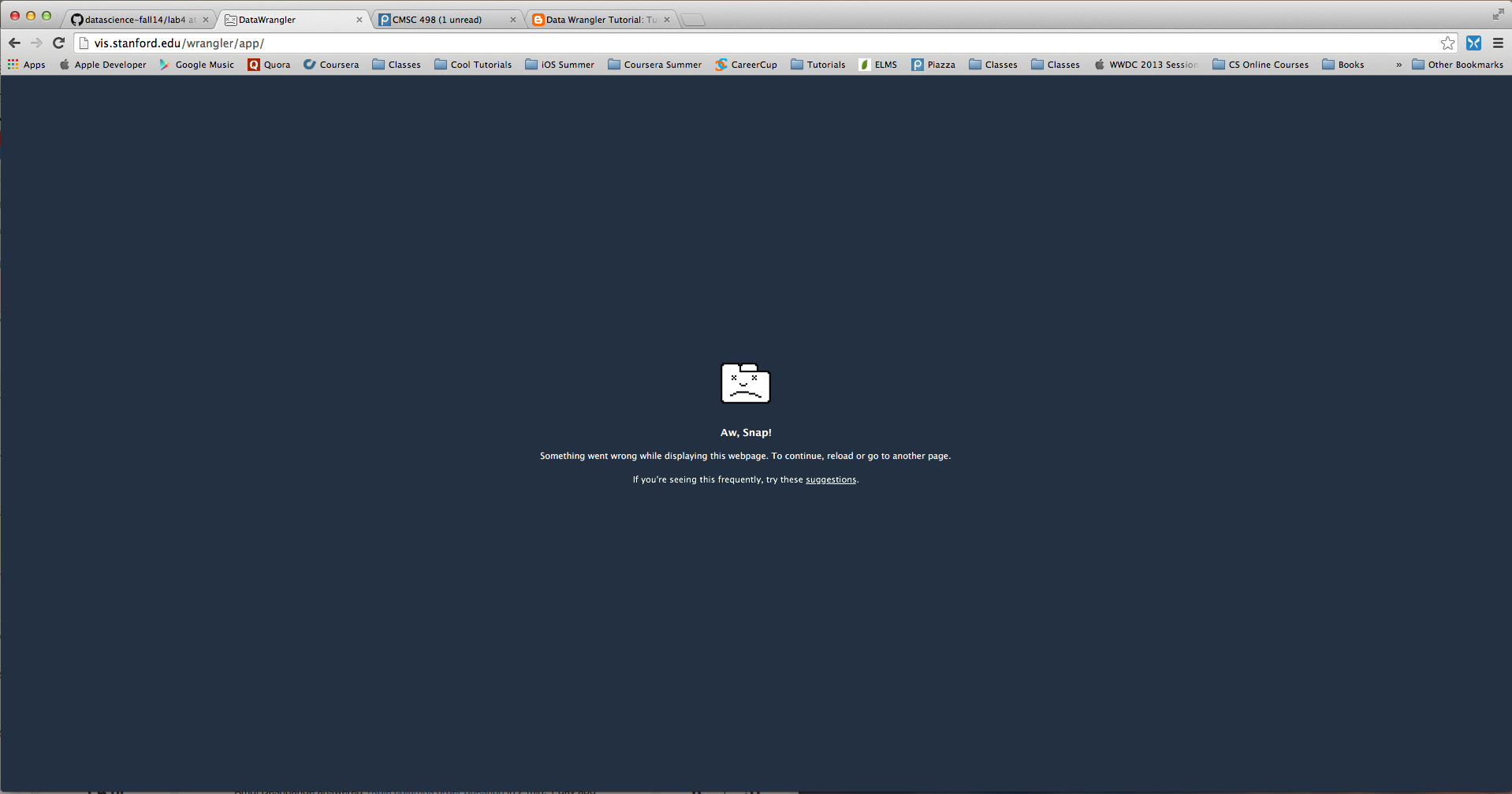
Note:

Data Wrangler was very hard to work with and I spent hours trying to get things to work without crashing, in fact this part took longer than part 3 of the lab. For the first script, CMSC, after performing the WRAP, I tried to split up the sections into different rows but I had no luck, as Data Wrangler would crash while trying the SPLIT command and other such commands. Nothing was possible for me after the WRAP command.

For parts 2 and 3, I was able to get the teams separated by name and the years in order of placements, though not in the exact CSV format for part 2. I was unable to reach part 3 due to Data Wrangler crashing.

Having worked with Data Wrangler for hours, I took a few screenshots of the crash page, but I was not able to move beyond the first couple of steps for the data even after trying multiple times on my laptop and other computers, so I hope you will understand the incompleteness of this part. It would be nice to work with a similar tool in the future provided it is in production.

**Data Wrangler Script and Screenshot: CMSC**



**from wrangler import dw**

**import sys**

**if(len(sys.argv) < 3):**

**sys.exit('Error: Please include an input and output file. Example python script.py input.csv output.csv')**

**w = dw.DataWrangler()**

**# Split data repeatedly on newline into rows**

**w.add(dw.Split(column=["data"],**

**table=0,**

**status="active",**

**drop=True,**

**result="row",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\n",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=0,**

**positions=None,**

**quote\_character=None))**

**# Wrap rows where data starts with 'CMSC'**

**w.add(dw.Wrap(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**row=dw.Row(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**conditions=[dw.StartsWith(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**lcol="data",**

**value="CMSC",**

**op\_str="starts with")])))**

**# Split wrap6 into rows**

**w.add(dw.Split(column=["wrap6"],**

**table=0,**

**status="active",**

**drop=True,**

**result="row",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on=None,**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

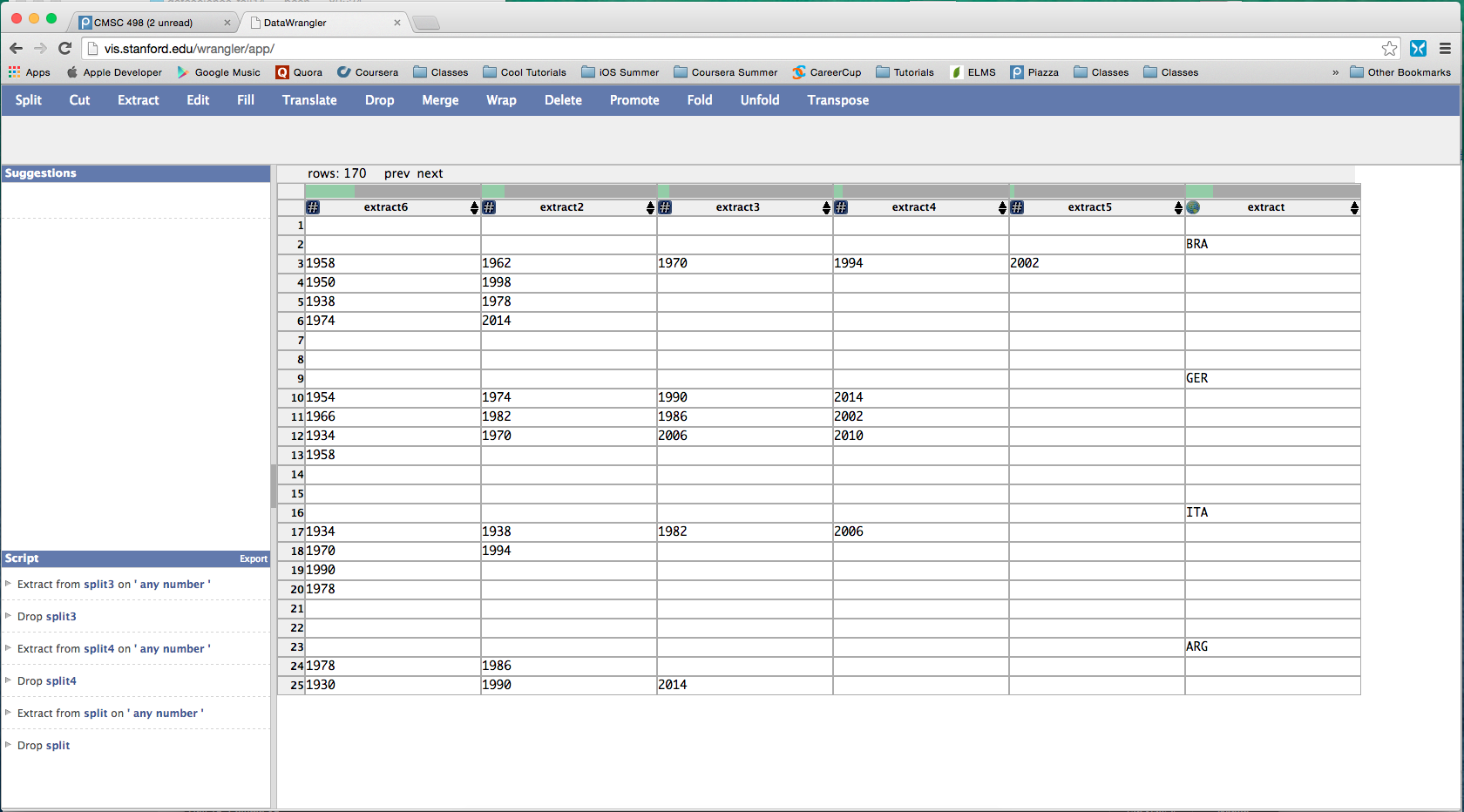
**max=1,**

**positions=None,**

**quote\_character=None))**

**w.apply\_to\_file(sys.argv[1]).print\_csv(sys.argv[2])**

**Data Wrangler Script and Screenshot: World Cup 1**

****

**from wrangler import dw**

**import sys**

**if(len(sys.argv) < 3):**

**sys.exit('Error: Please include an input and output file. Example python script.py input.csv output.csv')**

**w = dw.DataWrangler()**

**# Split data repeatedly on newline into rows**

**w.add(dw.Split(column=["data"],**

**table=0,**

**status="active",**

**drop=True,**

**result="row",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\n",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=0,**

**positions=None,**

**quote\_character=None))**

**# Cut on '"'**

**w.add(dw.Cut(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=True,**

**insert\_position="right",**

**row=None,**

**on="\"",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=0,**

**positions=None))**

**# Delete row 1**

**w.add(dw.Filter(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**row=dw.Row(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**conditions=[dw.RowIndex(column=[],**

**table=0,**

**status="active",**

**drop=False,**

**indices=[0])])))**

**# Extract from data between 'fb|' and '}'**

**w.add(dw.Extract(column=["data"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on=".\*",**

**before="}",**

**after="fb\\|",**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Extract from data between '(' and ')'**

**w.add(dw.Extract(column=["data"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on=".\*",**

**before="\\)",**

**after="\\(",**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Drop data**

**w.add(dw.Drop(column=["data"],**

**table=0,**

**status="active",**

**drop=True))**

**# Split extract1 repeatedly on ','**

**w.add(dw.Split(column=["extract1"],**

**table=0,**

**status="active",**

**drop=True,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on=",",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max="0",**

**positions=None,**

**quote\_character=None))**

**# Extract from split1 on ' any number '**

**w.add(dw.Extract(column=["split1"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\\d+",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Drop split1**

**w.add(dw.Drop(column=["split1"],**

**table=0,**

**status="active",**

**drop=True))**

**# Extract from split2 between positions 3, 7**

**w.add(dw.Extract(column=["split2"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on=None,**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=[3,7]))**

**# Drop split2**

**w.add(dw.Drop(column=["split2"],**

**table=0,**

**status="active",**

**drop=True))**

**# Extract from split3 on ' any number '**

**w.add(dw.Extract(column=["split3"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\\d+",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Drop split3**

**w.add(dw.Drop(column=["split3"],**

**table=0,**

**status="active",**

**drop=True))**

**# Extract from split4 on ' any number '**

**w.add(dw.Extract(column=["split4"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\\d+",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Drop split4**

**w.add(dw.Drop(column=["split4"],**

**table=0,**

**status="active",**

**drop=True))**

**# Extract from split on ' any number '**

**w.add(dw.Extract(column=["split"],**

**table=0,**

**status="active",**

**drop=False,**

**result="column",**

**update=False,**

**insert\_position="right",**

**row=None,**

**on="\\d+",**

**before=None,**

**after=None,**

**ignore\_between=None,**

**which=1,**

**max=1,**

**positions=None))**

**# Drop split**

**w.add(dw.Drop(column=["split"],**

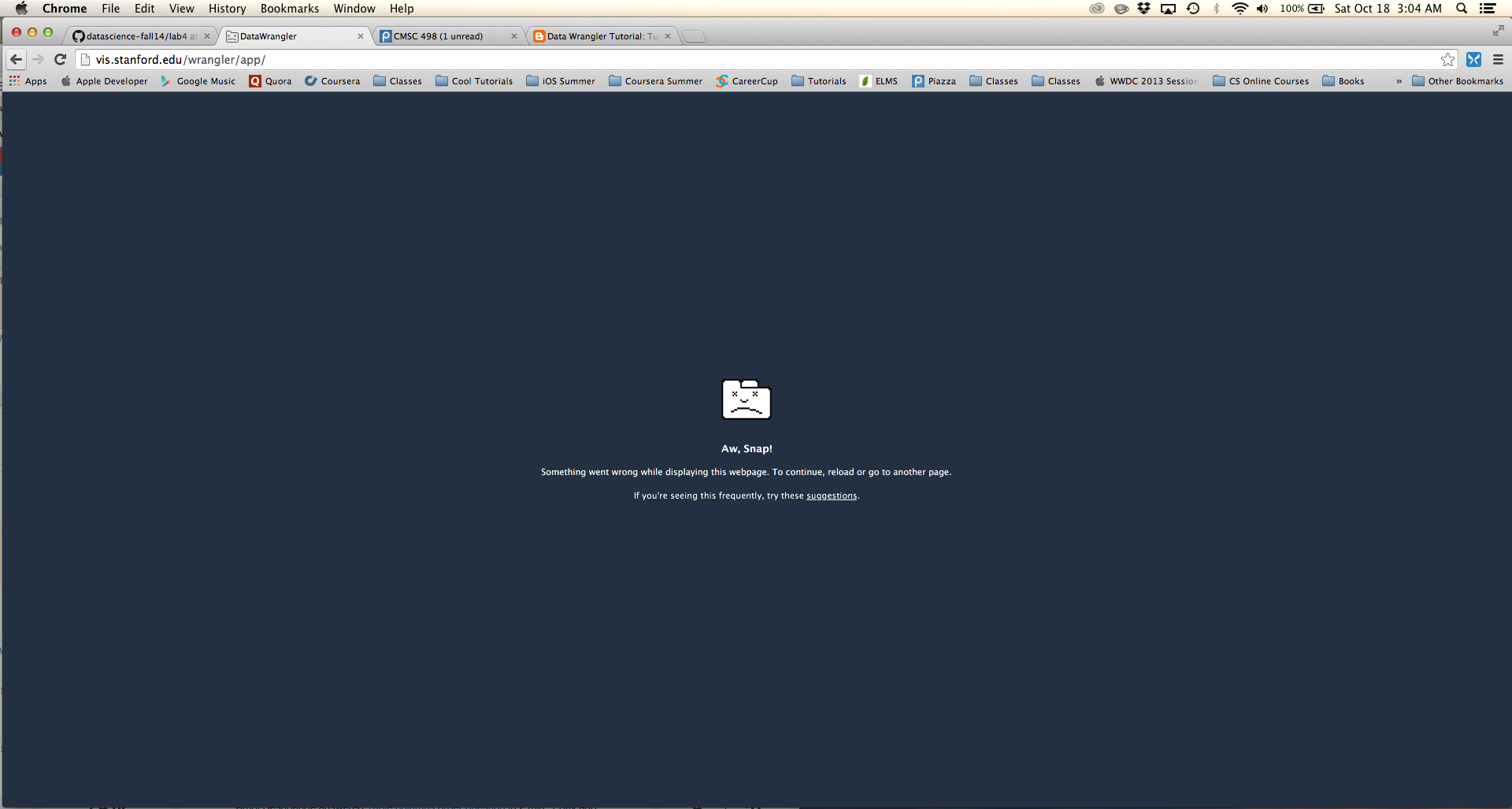
**table=0,**

**status="active",**

**drop=True))**

**w.apply\_to\_file(sys.argv[1]).print\_csv(sys.argv[2])**

**Data Wrangler Script and Screenshot: World Cup 2**

****

**UNIX Tools Command: CMSC**

**cat cmsc.txt | sed 's/[()]//g' | awk '/^(CMSC[0-9]{3})/ {c=$0;}**

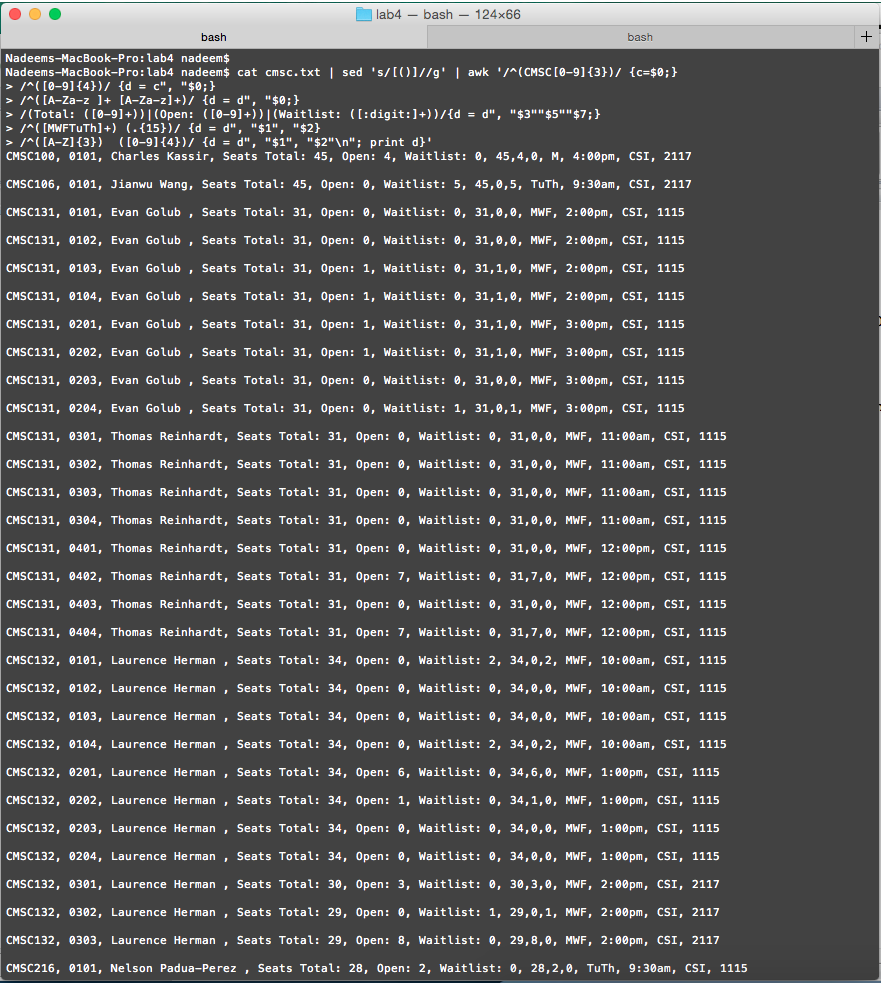
**/^([0-9]{4})/ {d = c", "$0;}**

**/^([A-Za-z ]+ [A-Za-z]+)/ {d = d", "$0;}**

**/(Total: ([0-9]+))|(Open: ([0-9]+))|(Waitlist: ([:digit:]+))/{d = d", "$3""$5""$7;}**

**/^([MWFTuTh]+) (.{15})/ {d = d", "$1", "$2}**

**/^([A-Z]{3}) ([0-9]{4})/ {d = d", "$1", "$2"\n"; print d}'**

****

**UNIX Tools Command: World Cup 1**

**cat worldcup.txt | awk '{gsub(".style=\".+\"", "");print}'|**

**sed 's/|{{fb|\('[A-Z]'\{3\}\)}}/\1/' | awk '{gsub("\\[|\\]", "");print}' | awk '{gsub("FIFA World Cup\\|[0-9]{4}","");print}' | awk '{gsub("#..\\\*","");print}' | awk '{gsub("(\\|)\*[0-9]{1,}(\\|)+[0-9]{1,}(\\|)+[0-9]{1,}","");print}' | awk '{gsub("(\\|)|-","");print}' | awk '{gsub("style=whitespace:nowrapTCH<sup>","");print}' | awk '{gsub("<sup>|#.\*","");print}' | awk '{gsub("\\(|\\)|,|\\}","");print}' |**

**awk '{gsub("([0-9])\*\\{\\{sort dash","");print}' |**

**awk 'BEGIN {ctr=0} /^([A-Z]{3})/ {c=$0; d=$0;}**

**/^([0-9]) ([0-9]{4})+/ {ctr = ctr+1;}**

**{ i = 1**

**x = 2**

**l = $1**

**if(l ~ /[A-Z]{3}/) {**

**ctr = 0**

**d = $1**

**}else if(l ~ /align=center/) {**

**ctr++**

**}**

**else {**

**for(i = 1; i <= l; i++) {**

**d = c", "$(i+1)", " ctr;**

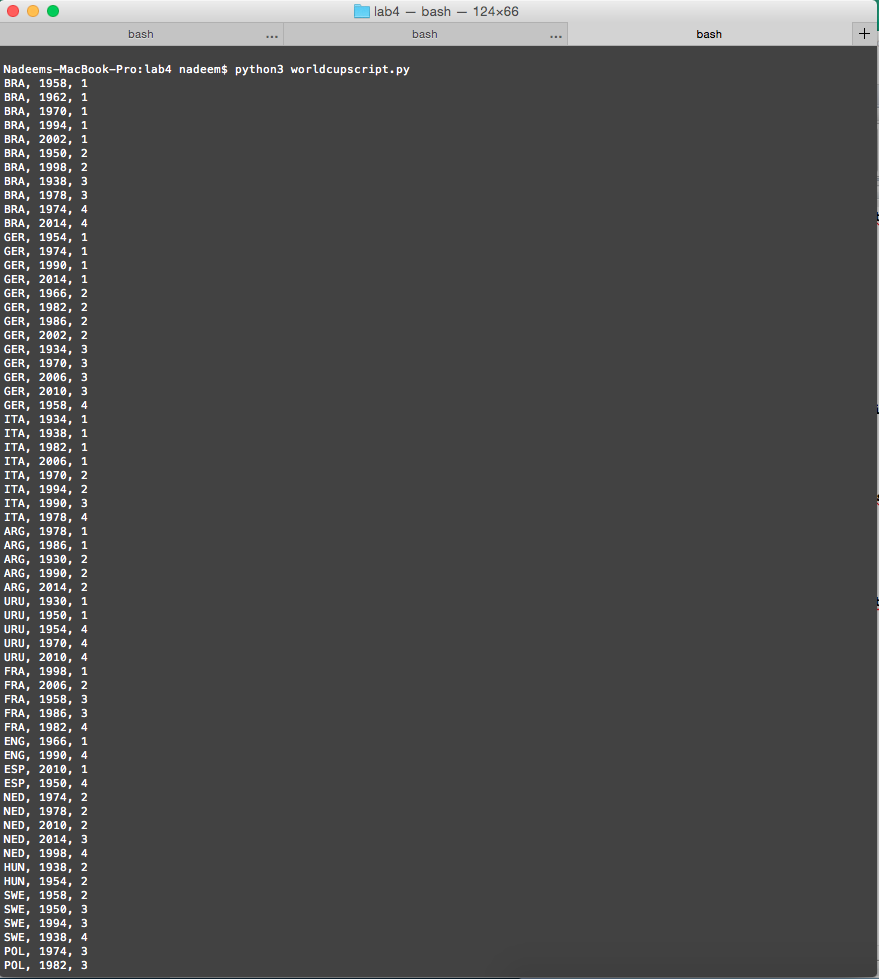
**print d;**

**}**

**d = c;**

**}**

**} END {d=""}'**

****

**Python Script: CMSC**

**import re**

**file\_name = "cmsc.txt"**

**with open(file\_name) as f:**

**lines = f.readlines()**

**num\_lines = len(lines)**

**i = 0**

**while i < num\_lines:**

**l = lines[i]**

**if re.match('^CMSC[0-9]{3}', l):**

**class\_name = l.rstrip()**

**info = class\_name + ", "**

**while(i + 1 < num\_lines and re.match('^[0-9]{4}', lines[i+1].rstrip())):**

**section = lines[i+1].rstrip();**

**prof\_name = lines[i+2].rstrip()**

**seats = lines[i+3].rstrip()**

**days\_time = lines[i+4].rstrip().split(" ")**

**days = days\_time[0]**

**days\_time = days\_time[1] + " - " + days\_time[3]**

**building, room = lines[i+5].rstrip().split(" ")**

**m\_seats = re.match('\d+',seats)**

**t, o, w = (seats[7:-1].split(", "))**

**t, o, w = t.split(": ")[1], o.split(": ")[1], w.split(": ")[1]**

**print(class\_name + ", " + section + ", " + \**

**prof\_name + ", " + t + ", " + o + ", " + w + ", " + \**

**days + "," + days\_time + ", " + building + ", " + room)**

**i = i + 5**

**else:**

**i += 1**

**Python Script: World Cup 1**

**import re**

**import fileinput**

**file\_name = "worldcup.txt"**

**import fileinput**

**lines = []**

**for line in fileinput.input(file\_name, inplace=False):**

**line = re.sub('(\\|)+(\d)+(\\|)+.+', "", line.rstrip())**

**line = re.sub('\\||{|}|fb|style=\".\*\"', "", line.rstrip())**

**line = re.sub('\\[|\\]', "", line.rstrip())**

**line = re.sub('FIFA World Cup(\d){4}(#\d\\\*)\*', "", line.rstrip())**

**line = re.sub('#\d\\^', "", line.rstrip())**

**line = re.sub('(^style=.+)|-|<sup>...</sup>|\\(|\\)', "", line.rstrip())**

**line = re.sub(' , ', " ", line.rstrip())**

**lines.append(line)**

**num\_lines = len(lines)**

**i = 0**

**while(not re.match('[A-Z]{3}', lines[i])):**

**i += 1**

**while(i < num\_lines and re.match('[A-Z]{3}', lines[i])):**

**country = lines[i].rstrip()**

**x = [1,2,3,4]**

**for placing in x:**

**l2 = lines[i+placing]**

**if(not re.match('align=centersort dash', l2)):**

**for year in l2.split(" "):**

**if len(year) > 1:**

**print (country + ", " + year + ", " + str(placing))**

**i += 5**

**while(i < num\_lines and not re.match('[A-Z]{3}', lines[i])):**

**i += 1**

**Python Script: World Cup 2**

**import re**

**import fileinput**

**import csv**

**import pandas as pd**

**import numpy as np**

**file\_name = "worldcup.txt"**

**import fileinput**

**lines = []**

**for line in fileinput.input(file\_name, inplace=False):**

**line = re.sub('(\\|)+(\d)+(\\|)+.+', "", line.rstrip())**

**line = re.sub('\\||{|}|fb|style=\".\*\"', "", line.rstrip())**

**line = re.sub('\\[|\\]', "", line.rstrip())**

**line = re.sub('FIFA World Cup(\d){4}(#\d\\\*)\*', "", line.rstrip())**

**line = re.sub('#\d\\^', "", line.rstrip())**

**line = re.sub('(^style=.+)|-|<sup>...</sup>|\\(|\\)', "", line.rstrip())**

**line = re.sub(' , ', " ", line.rstrip())**

**lines.append(line)**

**num\_lines = len(lines)**

**i = 0**

**outputFile = open("output.csv", 'wb')**

**wr = csv.writer(outputFile);**

**while(not re.match('[A-Z]{3}', lines[i])):**

**i += 1**

**csv = []**

**csv.append(["country", "year", "placing"])**

**while(i < num\_lines and re.match('[A-Z]{3}', lines[i])):**

**country = lines[i].rstrip()**

**x = [1,2,3,4]**

**for placing in x:**

**l2 = lines[i+placing]**

**if(not re.match('align=centersort dash', l2)):**

**for year in l2.split(" "):**

**if len(year) > 1:**

**csv.append([country, year, str(placing)])**

**i += 5**

**while(i < num\_lines and not re.match('[A-Z]{3}', lines[i])):**

**i += 1**

**wr.writerows(csv)**

**# set options to display the full table**

**pd.set\_option('display.max\_rows', 100)**

**pd.set\_option('display.max\_columns', 100)**

**pd.set\_option('display.width', 200)**

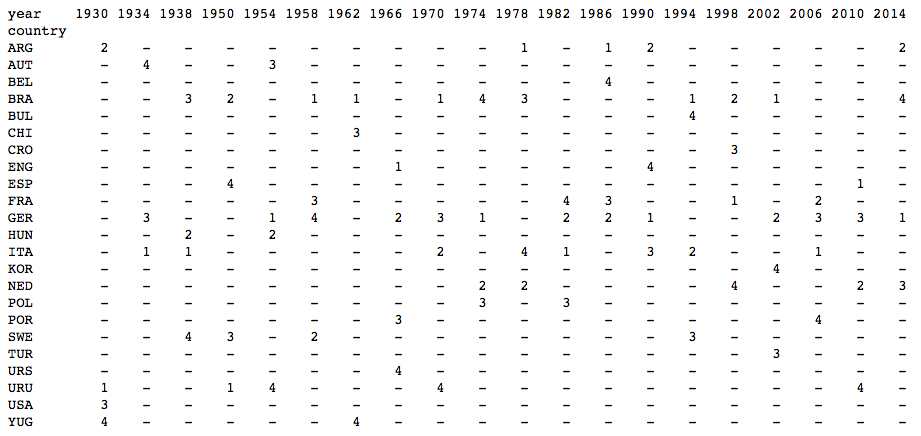
**# Read into pandas**

**df = pd.read\_csv('output.csv')**

**table = df.pivot(index='country', columns='year', values='placing')**

**table.fillna("-", inplace=True)**

**print table**

****