**Lab 4 Submission**

**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))

# Extract from data on 'CMSC any number '

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on="CMSC\\d+",

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None))

# Fill extract with values from above

w.add(dw.Fill(column=["extract"],

table=0,

status="active",

drop=False,

direction="down",

method="copy",

row=None))

# Wrap rows where data starts with '0'

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="0",

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

# 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])])))

# Drop wrap3, wrap5, wrap9, wrap7

w.add(dw.Drop(column=["wrap3","wrap5","wrap9","wrap7"],

table=0,

status="active",

drop=True))

# Drop wrap10, wrap11, wrap12, wrap13

w.add(dw.Drop(column=["wrap10","wrap11","wrap12","wrap13"],

table=0,

status="active",

drop=True))

# Set wrap1 name to Course No.

w.add(dw.SetName(column=["wrap1"],

table=0,

status="active",

drop=True,

names=["Course No."],

header\_row=None))

# Set wrap name to Section No.

w.add(dw.SetName(column=["wrap"],

table=0,

status="active",

drop=True,

names=["Section No."],

header\_row=None))

# Set Section\_No. name to Section No

w.add(dw.SetName(column=["Section\_No."],

table=0,

status="active",

drop=True,

names=["Section No"],

header\_row=None))

# Set Section\_No name to Section No

w.add(dw.SetName(column=["Section\_No"],

table=0,

status="active",

drop=True,

names=["Section No"],

header\_row=None))

# Set Section\_No1 name to Section No

w.add(dw.SetName(column=["Section\_No1"],

table=0,

status="active",

drop=True,

names=["Section No"],

header\_row=None))

# Set Course\_No. name to Course\_No

w.add(dw.SetName(column=["Course\_No."],

table=0,

status="active",

drop=True,

names=["Course\_No"],

header\_row=None))

# Extract from wrap8 after ' '

w.add(dw.Extract(column=["wrap8"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=None,

after=" ",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from wrap8 on ' any word '

w.add(dw.Extract(column=["wrap8"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on="[a-zA-Z]+",

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop wrap8

w.add(dw.Drop(column=["wrap8"],

table=0,

status="active",

drop=True))

# Extract from wrap6 after ' any word '

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=None,

after="[a-zA-Z]+ ",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from wrap6 on ' any word '

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on="[a-zA-Z]+",

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop wrap6

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

table=0,

status="active",

drop=True))

# Extract from wrap4 between ' Waitlist: ' and ')'

w.add(dw.Extract(column=["wrap4"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before="\\)",

after=" Waitlist: ",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from wrap4 between 'Open: ' and ','

w.add(dw.Extract(column=["wrap4"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=",",

after="Open: ",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from wrap4 between ': ' and ','

w.add(dw.Extract(column=["wrap4"],

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 wrap4

w.add(dw.Drop(column=["wrap4"],

table=0,

status="active",

drop=True))

# Set wrap2 name to Instructor

w.add(dw.SetName(column=["wrap2"],

table=0,

status="active",

drop=True,

names=["Instructor"],

header\_row=None))

# Set extract6 name to Seats

w.add(dw.SetName(column=["extract6"],

table=0,

status="active",

drop=True,

names=["Seats"],

header\_row=None))

# Set extract5 name to Open

w.add(dw.SetName(column=["extract5"],

table=0,

status="active",

drop=True,

names=["Open"],

header\_row=None))

# Set extract4 name to Wait List

w.add(dw.SetName(column=["extract4"],

table=0,

status="active",

drop=True,

names=["Wait List"],

header\_row=None))

# Set extract3 name to Days

w.add(dw.SetName(column=["extract3"],

table=0,

status="active",

drop=True,

names=["Days"],

header\_row=None))

# Set extract2 name to Times

w.add(dw.SetName(column=["extract2"],

table=0,

status="active",

drop=True,

names=["Times"],

header\_row=None))

# Set Times name to Time

w.add(dw.SetName(column=["Times"],

table=0,

status="active",

drop=True,

names=["Time"],

header\_row=None))

# Set extract1 name to Building

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

table=0,

status="active",

drop=True,

names=["Building"],

header\_row=None))

# Set Building name to Bldg.

w.add(dw.SetName(column=["Building"],

table=0,

status="active",

drop=True,

names=["Bldg."],

header\_row=None))

# Set extract name to Room No

w.add(dw.SetName(column=["extract"],

table=0,

status="active",

drop=True,

names=["Room No"],

header\_row=None))

# Merge Course\_No, Section\_No with glue ,

w.add(dw.Merge(column=["Course\_No","Section\_No"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

glue=","))

# Extract from merge after ','

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=None,

after=",",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from merge on 'CMSC any number '

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on="CMSC\\d+",

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop Section\_No, merge, Course\_No

w.add(dw.Drop(column=["Section\_No","merge","Course\_No"],

table=0,

status="active",

drop=True))

# Set extract7 name to Course No

w.add(dw.SetName(column=["extract7"],

table=0,

status="active",

drop=True,

names=["Course No"],

header\_row=None))

# Set extract name to Section No

w.add(dw.SetName(column=["extract"],

table=0,

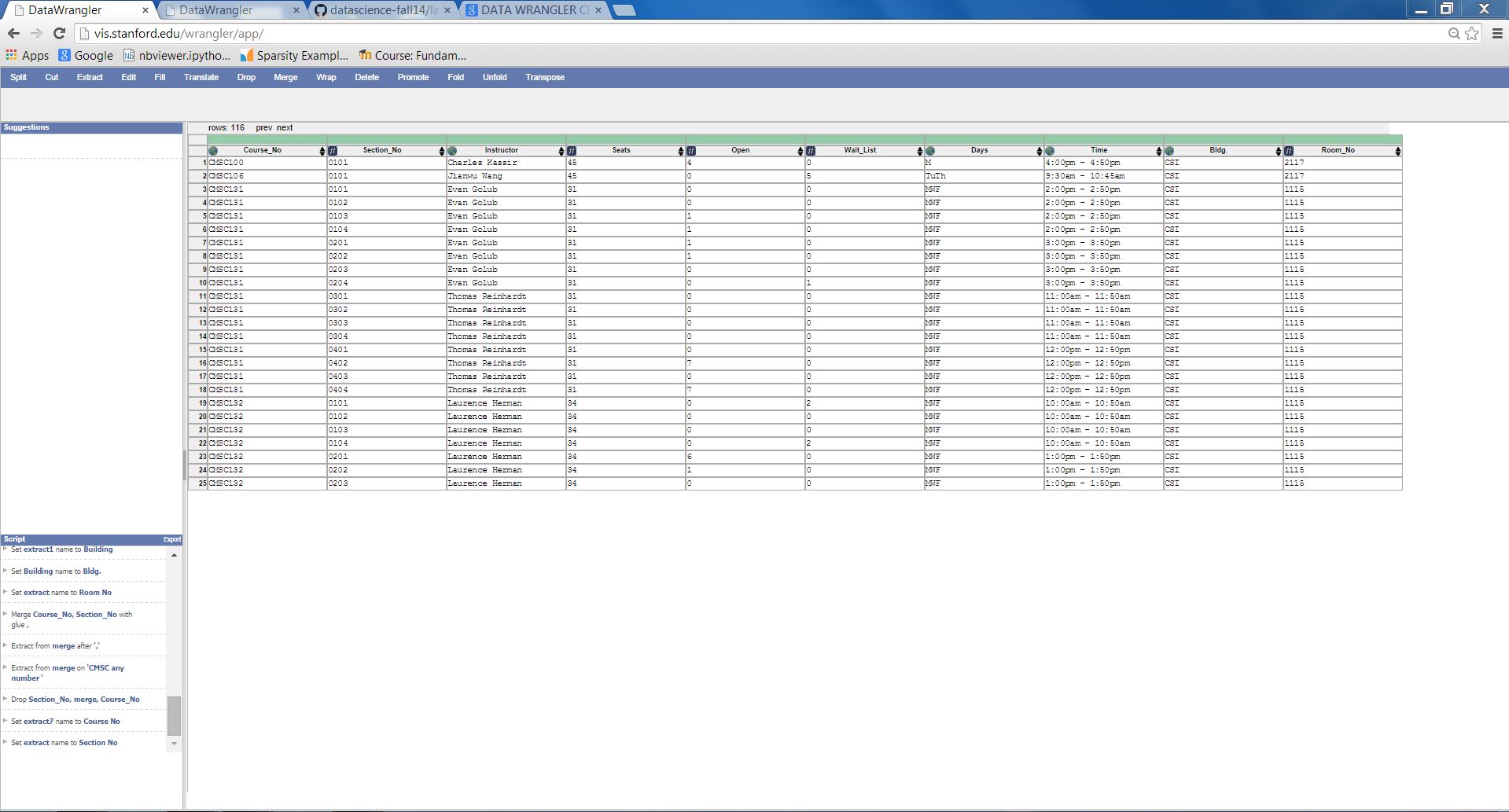
status="active",

drop=True,

names=["Section No"],

header\_row=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))

# Split data repeatedly on '|'

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

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="\""))

# 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))

# Drop split

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

table=0,

status="active",

drop=True))

# Extract from split1 between '[\[' and ' FIFA'

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

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 '[\[' and ' FIFA'

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop split2

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

table=0,

status="active",

drop=True))

# Extract from split3 before '}}'

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before="}}",

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None))

# Extract from split3 between '[\[' and ' FIFA'

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

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 between '[\[' and ' FIFA'

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

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

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 split5 between '[\[' and ' FIFA'

w.add(dw.Extract(column=["split5"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop split5

w.add(dw.Drop(column=["split5"],

table=0,

status="active",

drop=True))

# Extract from split6 between '[\[' and ' FIFA'

w.add(dw.Extract(column=["split6"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

on=".\*",

before=" FIFA",

after="\\[\\[",

ignore\_between=None,

which=1,

max=1,

positions=None))

# Drop split6

w.add(dw.Drop(column=["split6"],

table=0,

status="active",

drop=True))

# Drop split7

w.add(dw.Drop(column=["split7"],

table=0,

status="active",

drop=True))

# Fill extract2 with values from above

w.add(dw.Fill(column=["extract2"],

table=0,

status="active",

drop=False,

direction="down",

method="copy",

row=None))

# Delete empty rows

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

table=0,

status="active",

drop=False,

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

conditions=[dw.Empty(column=[],

table=0,

status="active",

drop=False,

percent\_valid=0,

num\_valid=0)])))

# Merge extract2, extract, extract1, extract3... with glue ,

w.add(dw.Merge(column=["extract2","extract","extract1","extract3","extract4","extract5","extract6"],

table=0,

status="active",

drop=False,

result="column",

update=False,

insert\_position="right",

row=None,

glue=","))

# Drop extract, extract1, extract3, extract2...

w.add(dw.Drop(column=["extract","extract1","extract3","extract2","extract4","extract5","extract6"],

table=0,

status="active",

drop=True))

# Split merge repeatedly on ','

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

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))

# Delete rows where split8 is null

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

table=0,

status="active",

drop=False,

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

conditions=[dw.IsNull(column=[],

table=0,

status="active",

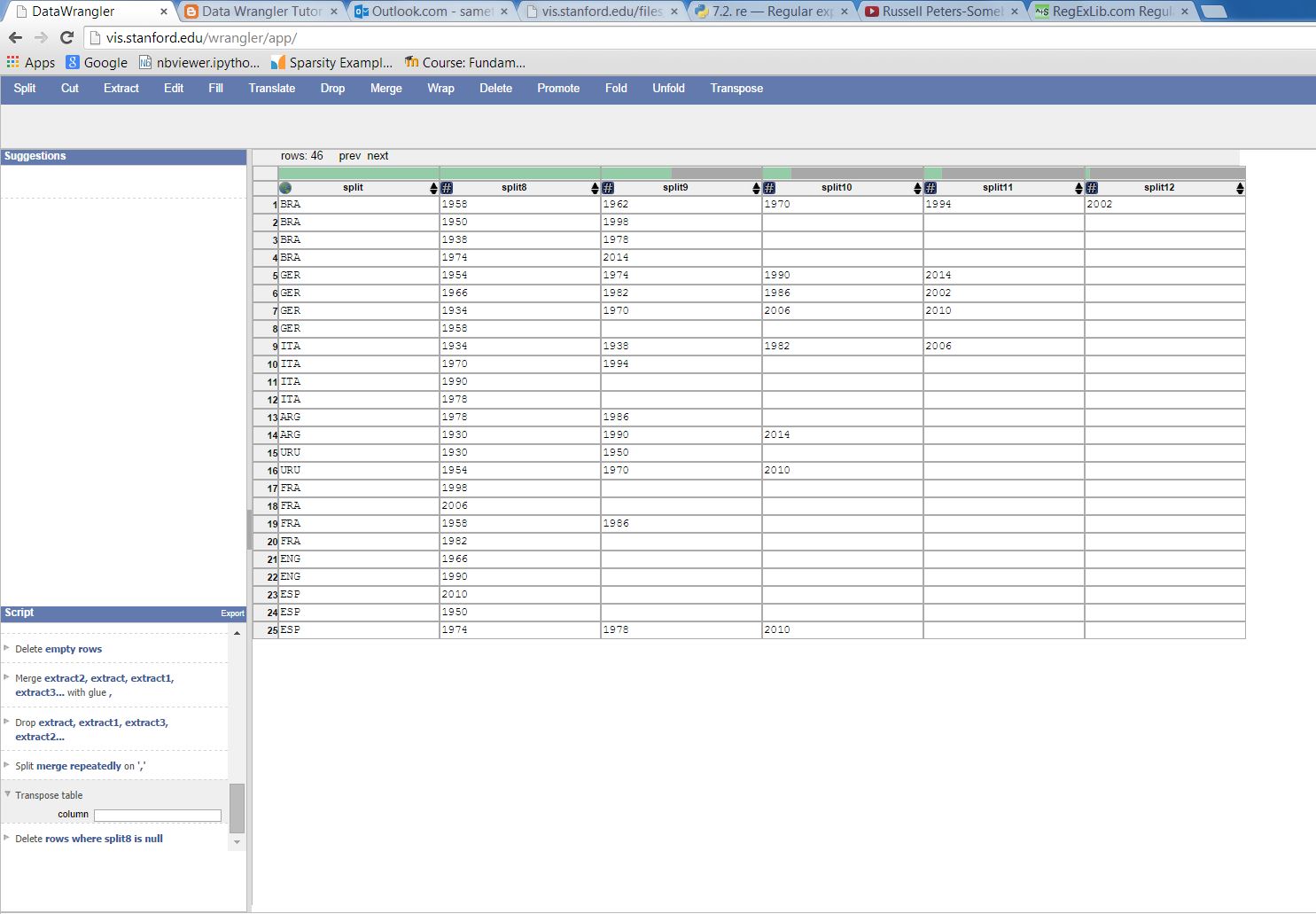
drop=False,

lcol="split8",

value=None,

op\_str="is null")])))

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 | grep -v '^$' | awk -v OFS=', ' '/^(CSI|AVW|ITV|MTH|JMP)/ {print $1, $2} !/^(CSI|AVW|ITV|MTH|JMP) / {print $0}' | awk -F')' '/^Seats/ {print $1} !/^Seats/ {print $0}' | awk -F',' '/^Seats/ {print $1, $2, $3} !/^Seats/ {print $0}' | awk -v OFS=', ' '/^CMSC/ {Class = $1} /^0/ {print Class, $0} !/(^CMSC|^0|^Seats)/ {print $0} /^Seats/ {print $3,$5,$7} ' | awk '/^CMSC/ {print combined; combined = $0} !/^CMSC/ {combined = combined", "$0;} END {print combined}'

**UNIX Tools Command: World Cup 1**

cat worldcup.txt | tail +3 | sed 's/{{//g' | sed 's/|style="background:#fff68f"|//g' | sed 's/}}/\ /g' | sed 's/}}/\ /g' | sed 's/(fb\||\|fb\|)//g' | awk '/^(fb|\|fb)/ {count = 0;print $0} !/^(fb|\|fb)/{count = count + 1; if(count < 5) print count," ",$0}' | sed 's/\|fb\|//g' | sed 's/fb\|//g' | awk '/^[A-Z]/ {country = $1} !/^[A-Z]/ {print country, $0}' | grep -v 'sort dash' | grep -v '-' | sed 's/\|[0-9][0-9][0-9][0-9]//g' | sed 's/FIFA\ World\ Cup//g' | sed 's/(\[\[//g' | sed 's/]])//g' | sed 's/\[\[//g' | sed 's/]]\*//g' | sed 's/#\*//g' | sed 's/\\*//g' | sed 's/\,//g' | sed 's/[0-9]\|//g' | awk -v OFS=', ' '// { country = $1; title = $2; for(i = 4; i <= NF; i++) {print country, $i, title;}}'

**Python Script: CMSC**

import unicodecsv

fw = open("cmsc-python-output.csv","w")

writer = unicodecsv.writer(fw, encoding="utf8", lineterminator="\n")

writer.writerow(("Course No.","Section No.","Instructor","Seats","Open","Waitlist","Days","Time","Bldg.","Room No."))

f = open("cmsc.txt","r")

line = f.readline().strip()

while line!='':

course = line

line = f.readline().strip()

while line != '':

section = line

instr = f.readline().strip()

line = f.readline().strip().split(": ")

total = line[1].split(",")[0]

open = line[2].split(",")[0]

wait = line[3].split(")")[0]

line = f.readline().strip().split()

day = line[0]

time = ' '.join(line[1:])

line = f.readline().strip().split()

bldg = line[0]

room = line[1]

writer.writerow((course, section, instr, total, open, wait, day, time, bldg, room))

line = f.readline().strip()

line = f.readline().strip()

f.close()

fw.close()

**Python Script: World Cup 1**

import unicodecsv

import re

fw = open("worldcup-python-output1.csv","w")

writer = unicodecsv.writer(fw, encoding="utf8", lineterminator="\n")

writer.writerow(("Team","Year","Position"))

f = open("worldcup.txt","r")

f.readline()

line = f.readline().strip()

while line!="|}":

line = f.readline().strip()

country = line.split("{{fb|")[1].split("}}")[0]

for i in range(4):

position = re.findall("\|\d{4}]]", f.readline().strip())

for position in position:

writer.writerow((country, position[1:-2], i+1))

f.readline()

line = f.readline().strip()

fw.close()

f.close()

**Python Script: World Cup 2**

import unicodecsv

import re

fw = open("worldcup-python-output2.csv","w")

writer = unicodecsv.writer(fw, encoding="utf8", lineterminator="\n")

titles = {}

f = open("worldcup.txt","r")

f.readline()

line = f.readline().strip()

while line!="|}":

line = f.readline().strip()

country = line.split("{{fb|")[1].split("}}")[0]

countrytitles = titles.get(country, {})

for i in range(4):

position = re.findall("\|\d{4}]]", f.readline().strip())

for position in position:

countrytitles[position[1:-2]] = i+1

titles[country] = countrytitles

f.readline()

line = f.readline().strip()

f.close()

writer.writerow([i for i in range(1930,2015,4)])

for k in titles:

m = [k]

for i in range(1930, 2015, 4):

m.append('-' if str(i) not in titles[k].keys() else titles[k][str(i)])

writer.writerow(m)

fw.close()