Sailunsi Chen

Lab 4 Submission

=========== Data Wrangler Script: 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",

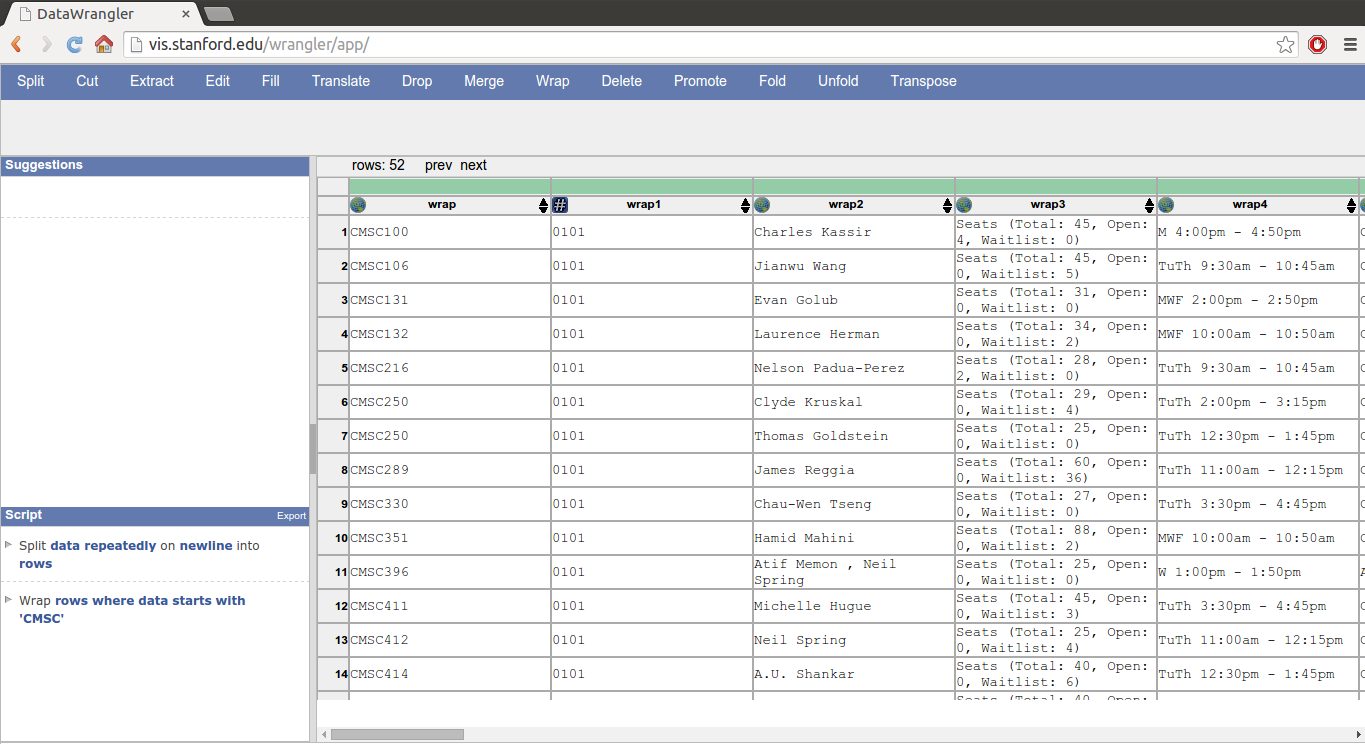
drop=False,

lcol="data",

value="CMSC",

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

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



=========== Data Wrangler Script: 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))

# Wrap rows where data contains '|-'

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

table=0,

status="active",

drop=False,

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

lcol="data",

value="|-",

op\_str="contains")])))

# Drop wrap

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

table=0,

status="active",

drop=True))

# 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 wrap1 between 'fb|' and '}'

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

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))

# Drop wrap1

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

table=0,

status="active",

drop=True))

# Split wrap2 repeatedly on ',' into rows

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

table=0,

status="active",

drop=True,

result="row",

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 wrap2 between '[\[' and ' FIFA'

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

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 wrap2

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

table=0,

status="active",

drop=True))

# Edit extract1 row 1 to ' 1958, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[0])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None,

to="1958, 1",

update\_method=None))

# Edit extract1 row 2 to ' 1962, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[1])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None,

to="1962, 1",

update\_method=None))

# Edit extract1 row 3 to ' 1970, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[2])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None,

to="1970, 1",

update\_method=None))

# Edit extract1 row 4 to ' 1994, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[3])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None,

to="1994, 1",

update\_method=None))

# Edit extract1 row 5 to ' 2002, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[4])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

max=1,

positions=None,

to="2002, 1",

update\_method=None))

# Edit extract1 row 6 to ' 1954, 1 '

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

table=0,

status="active",

drop=False,

result="column",

update=True,

insert\_position="right",

row=dw.Row(column=[],

table=0,

status="active",

drop=False,

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

table=0,

status="active",

drop=False,

indices=[5])]),

on=None,

before=None,

after=None,

ignore\_between=None,

which=1,

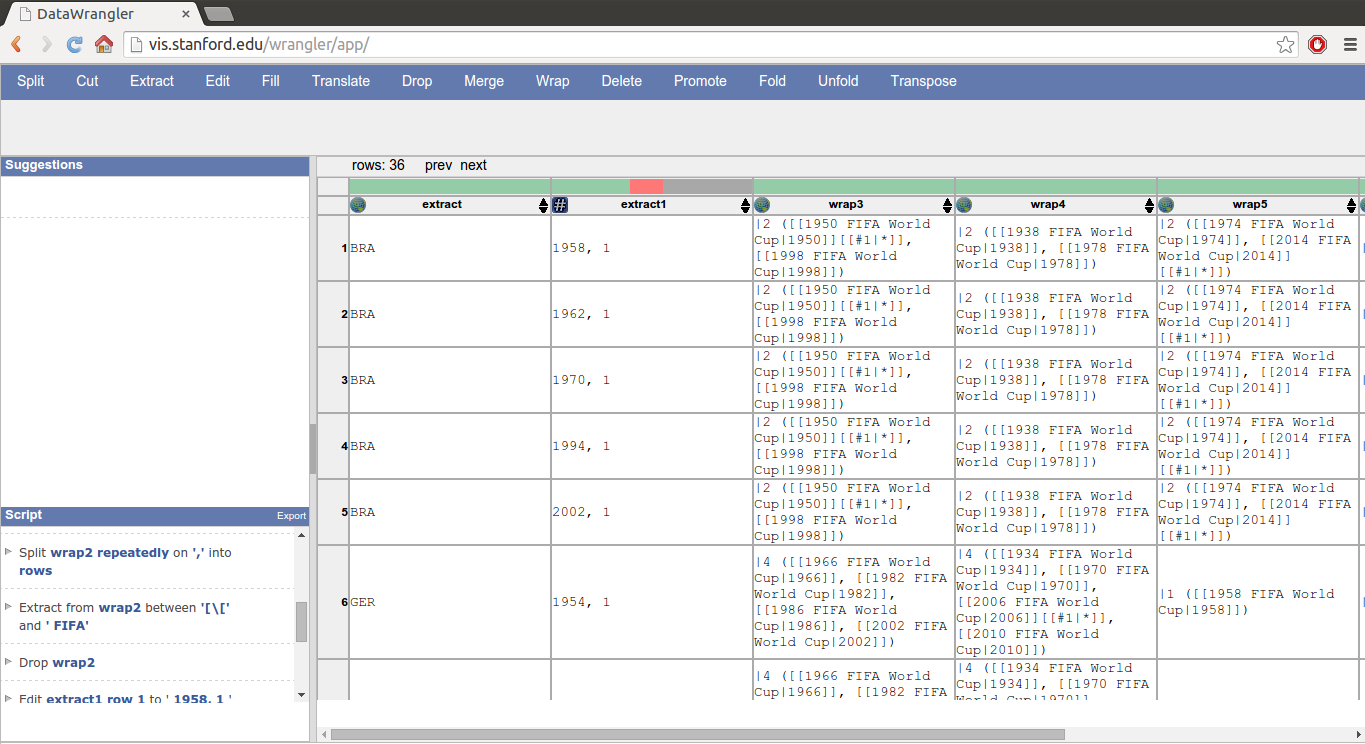
max=1,

positions=None,

to="1954, 1",

update\_method=None))

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

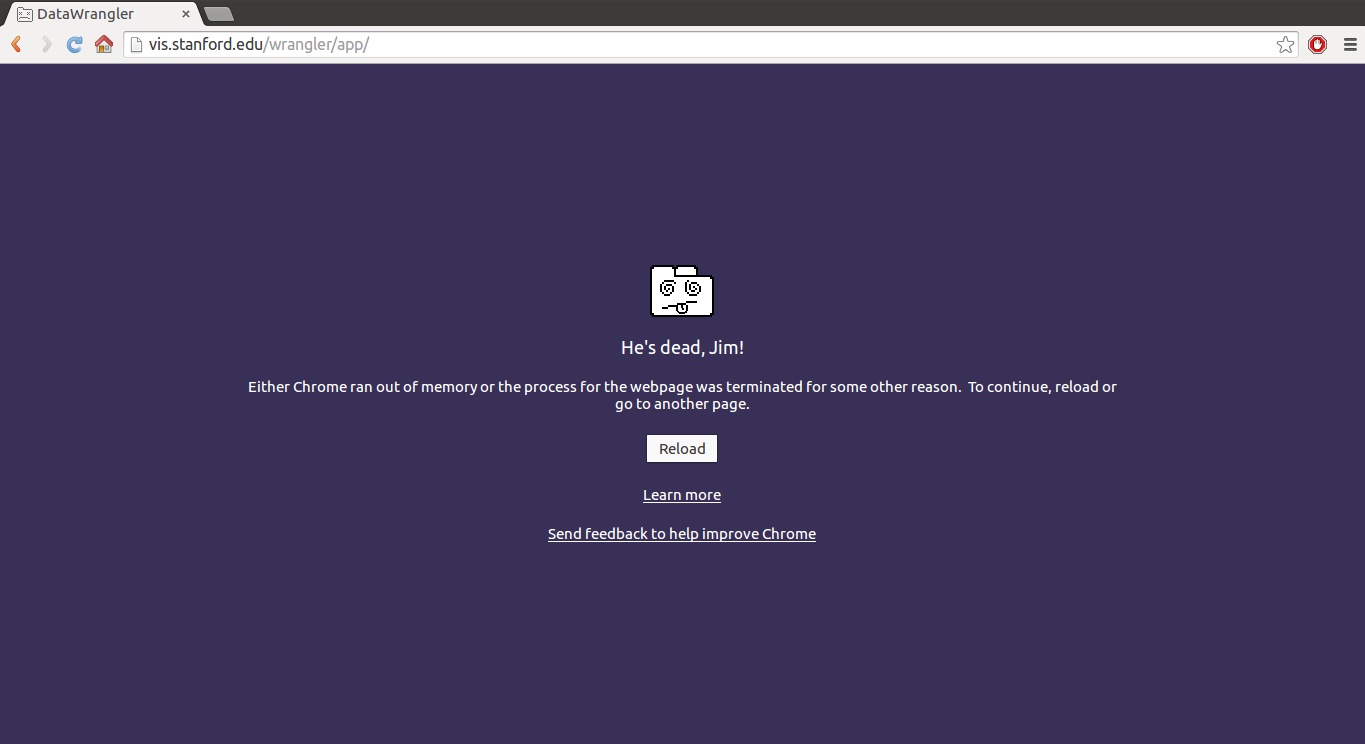


=========== Data Wrangler Script: World Cup 2

Data wrangler keeps crashing before I can finish World Cup 1.

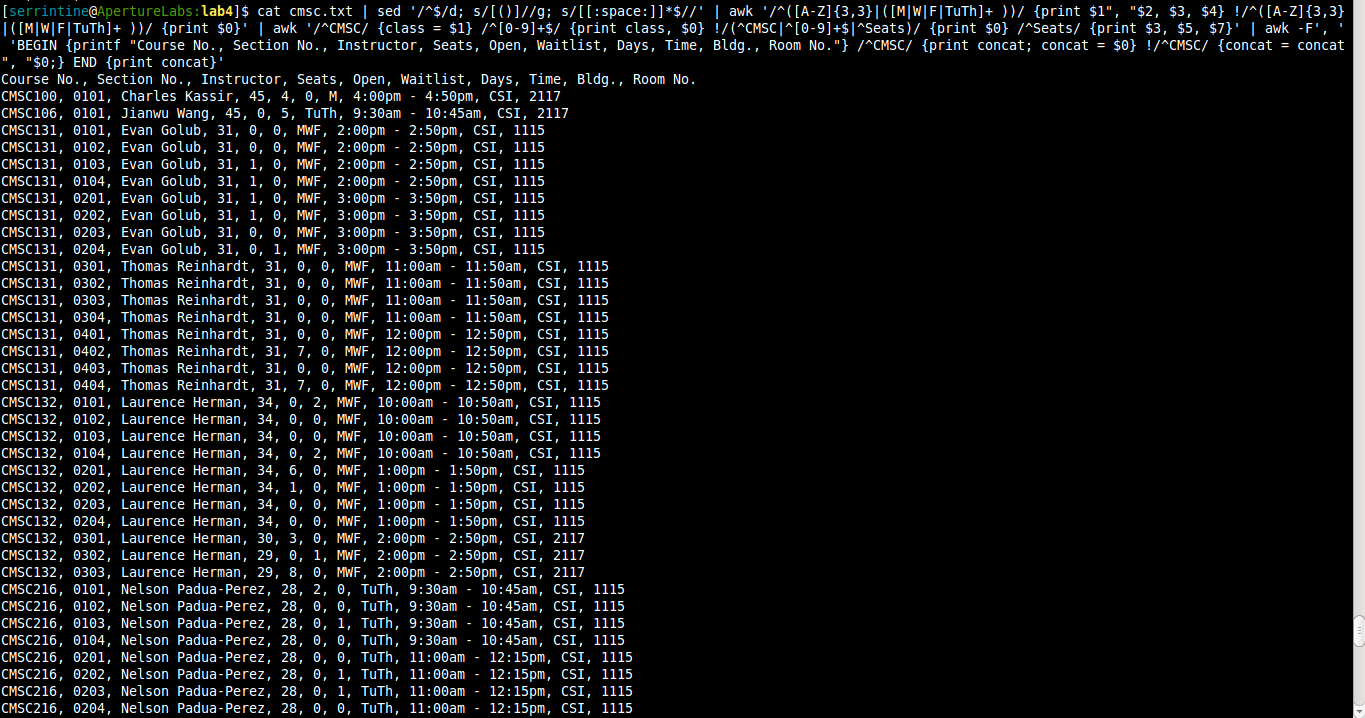
I have given up on moving forward with it. This is not even

remotely close to a competent tool for cleaning up data.



=========== UNIX Tools Command: CMSC

cat cmsc.txt | sed '/^$/d; s/[()]//g; s/[[:space:]]\*$//' | awk '/^([A-Z]{3,3}|([M|W|F|TuTh]+ ))/ {print $1", "$2, $3, $4} !/^([A-Z]{3,3}|([M|W|F|TuTh]+ ))/ {print $0}' | awk '/^CMSC/ {class = $1} /^[0-9]+$/ {print class, $0} !/(^CMSC|^[0-9]+$|^Seats)/ {print $0} /^Seats/ {print $3, $5, $7}' | awk -F', ' 'BEGIN {printf "Course No., Section No., Instructor, Seats, Open, Waitlist, Days, Time, Bldg., Room No."} /^CMSC/ {print concat; concat = $0} !/^CMSC/ {concat = concat", "$0;} END {print concat}'



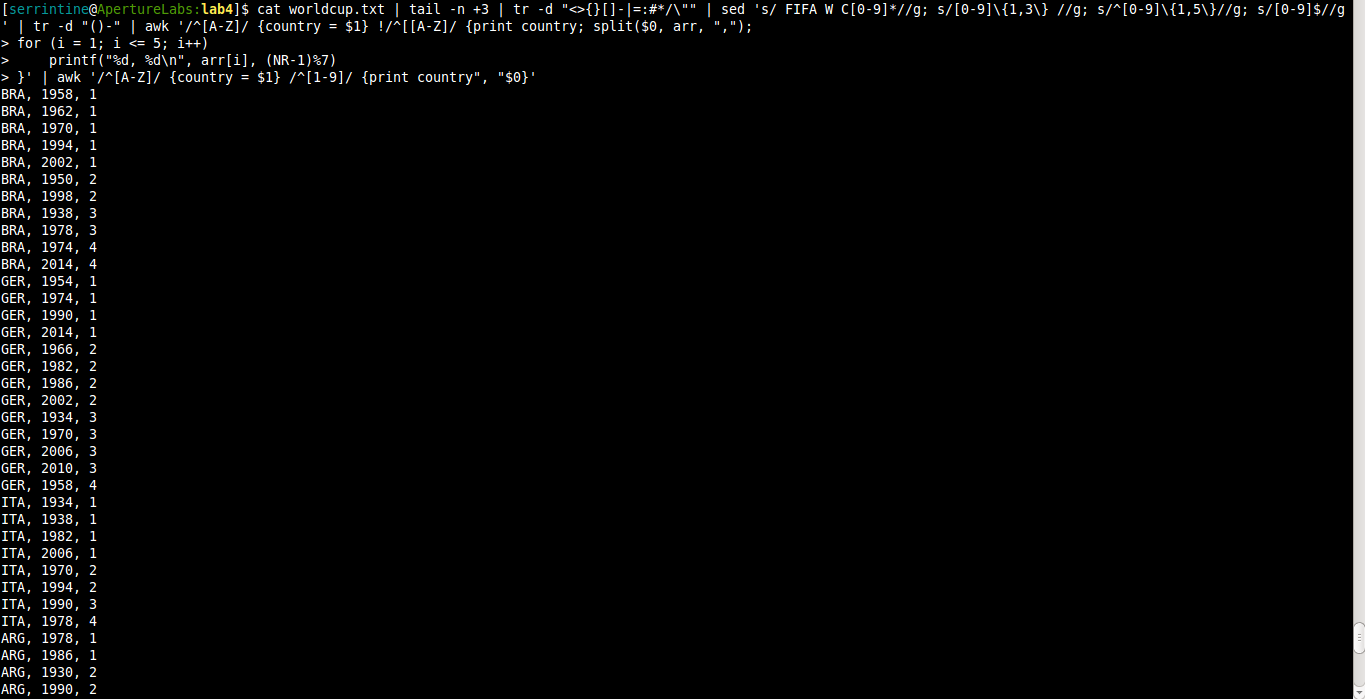
=========== UNIX Tools Command: World Cup 1

cat worldcup.txt | tail -n +3 | tr -d "<>{}[]-|=:#\*/\"" | sed 's/ FIFA W C[0-9]\*//g; s/[0-9]\{1,3\} //g; s/^[0-9]\{1,5\}//g; s/[0-9]$//g' | tr -d "()-" | awk '/^[A-Z]/ {country = $1} !/^[[A-Z]/ {print country; split($0, arr, ",");

for (i = 1; i <= 5; i++)

printf("%d, %d\n", arr[i], (NR-1)%7)

}' | awk '/^[A-Z]/ {country = $1} /^[1-9]/ {print country", "$0}'



=========== Python Script: CMSC

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

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

line = f.readline().strip()

while line:

course = line

line = f.readline().strip()

while line:

out = []

out.append(course)

out.append(line)

# out.append('"' + f.readline().strip() + '"')

out.append(f.readline().strip())

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

out.append(line[1].split(",")[0])

out.append(line[2].split(",")[0])

out.append(line[3].split(")")[0])

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

out.append(line[0])

out.append(' '.join(line[1:]))

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

out.append(line[0])

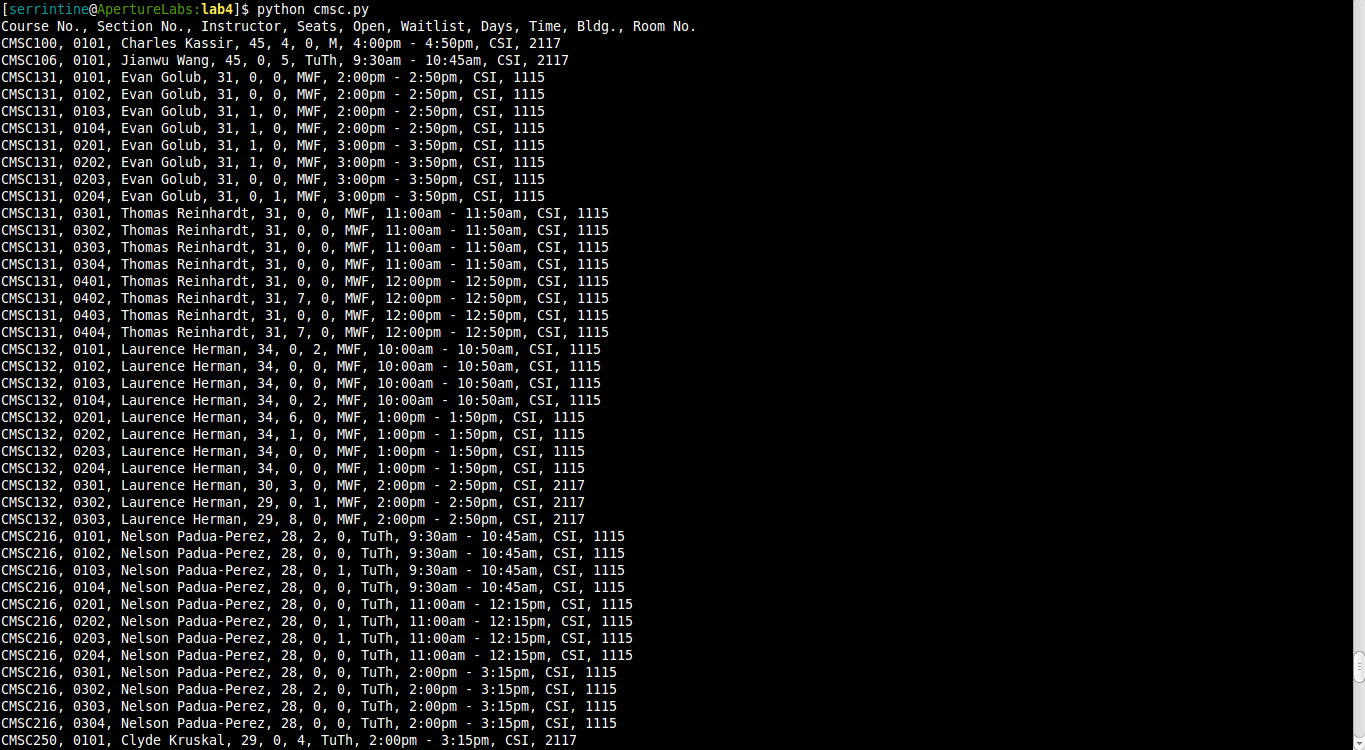
out.append(line[1])

print(', '.join(map(str,out)))

line = f.readline().strip()

line = f.readline().strip()

f.close()



=========== Python Script: World Cup 1

import re

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

print("Country, Year, Title")

f.readline()

f.readline()

line = f.readline().strip()

while line:

for i in range(4):

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

for pos in position:

out = []

out.append(re.search(r"[A-Z]{3}", line).group(0))

out.append(pos[1:-2])

out.append(i+1)

print(', '.join(map(str,out)))

f.readline()

f.readline()

line = f.readline().strip()

f.close()



=========== Python Script: World Cup 2

import sys

import pandas as pd

worldcup = pd.read\_csv(sys.stdin, header=0, names=['Country', 'Year', 'Title'])

pivoted = worldcup.pivot(index='Country', columns='Year', values='Title')

print pivoted.to\_string(na\_rep='-', index\_names=False)

