**PYTHON ASSIGNMENT 4**

1. **Write a Python program to read a file line by line and store it into a list**

IN []:

file1 = open("file.txt","a")

file1.writelines("this is the first line\nthis is the second line\nthis is the third line")

file1 = open("file.txt","r")

print(file1.readlines())

type(file1.readlines())

OUT []:

['this is the first line\n', 'this is the second line\n', 'this is the third line']

list

1. **Write a Python program to read a file line by line store it into an array.**

IN [] :

import numpy as np

file1 = open("file.txt","r")

read\_data\_lines = file1.readlines()

arr = np.array(read\_data\_lines)

print(arr)

print(type(arr))

OUT []:

['this is the first line\n' 'this is the second line\n'

'this is the third line']

<class 'numpy.ndarray'>

1. **Write a Python program to read a random line from a file.**

IN []:

import random

def random\_line(file):

line = open(file).readlines()

return random.choice(line)

print(random\_line('file.txt'))

OUT []:

this is the sixth line

1. **Write a Python program to combine each line from first file with the corresponding line in second file.**

IN []:

file1 = open('file.txt')

file2 = open('myfile.txt')

for line1, line2 in zip(file1, file2):

print(line1+line2)

OUT []:

this is the first line

line 1

this is the second line

line 2

this is the third line

line 3

1. **Write a Python program to generate 26 text files named A.txt, B.txt, and so on up to Z.txt.**

IN []:

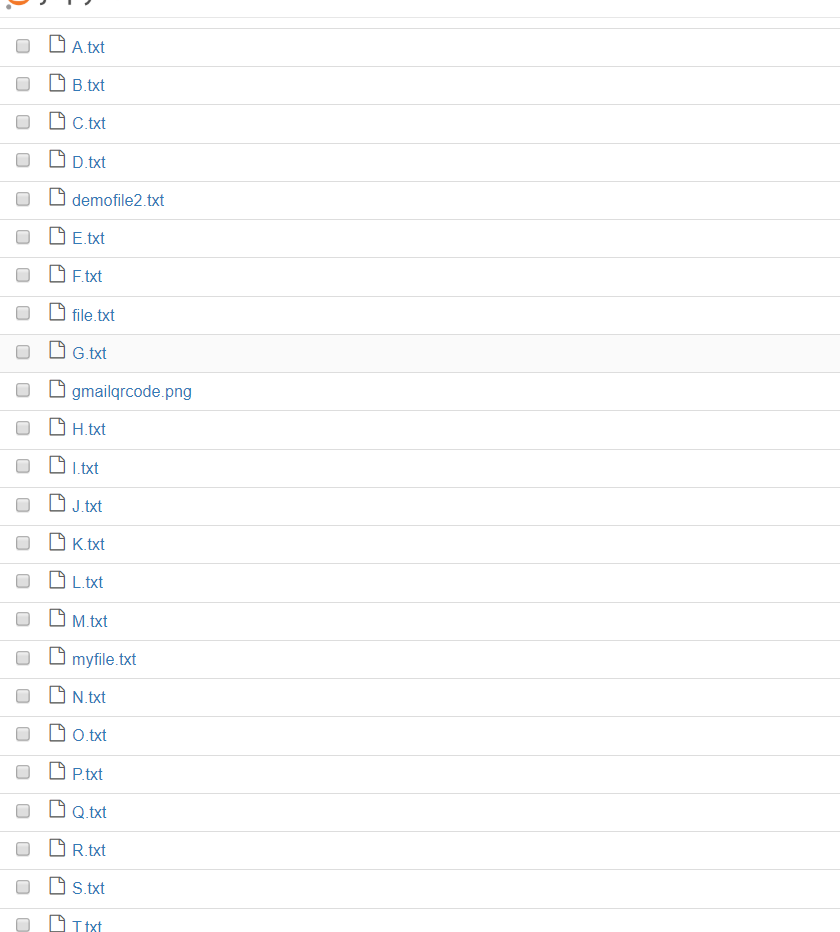
name = "A"

for i in range(0,26):

file = open(name+".txt", "w+")

name = chr(ord(name) + 1)

OUT []:



1. **Write a Python program to create a file where all letters of English alphabet are listed by specified number of letters on each line.**

IN []:

n = int(input())

lis = []

list = []

file = open("A.txt","w")

alpha = "A"

for i in range(26): #list of all 26 alphabets

lis.append(alpha)

alpha = chr(ord(alpha)+1)

for j in range(0,26,n): #list of series of divided alphabets on each line

list.append(lis[j:n+j])

print("\n")

with open('A.txt', 'w') as f: #writing to the file

for \_list in list:

f.write(str(\_list) + '\n')

OUT []:

Given the value of n = 4



**7. #### MAIN TASK ####**

**- To scrap data from worldometer example: INDIA Data and run it on live mode.**

**- Print Additionally total number of Coronavirus Cases, Deaths, Recovered.**

IN []:

import requests

from bs4 import BeautifulSoup

import bs4

req = requests.get("https://www.worldometers.info/coronavirus/")

page = req.text

soup = bs4.BeautifulSoup(page)

#heading of the page

soup.find("title").get\_text()

OUT []:

'Coronavirus Update (Live): 4,041,446 Cases and 276,914 Deaths from COVID-19 Virus Pandemic - Worldometer'

IN []:

#printing the table from the web using HTML package

html\_soup = soup.find\_all('table')

type(html\_soup)

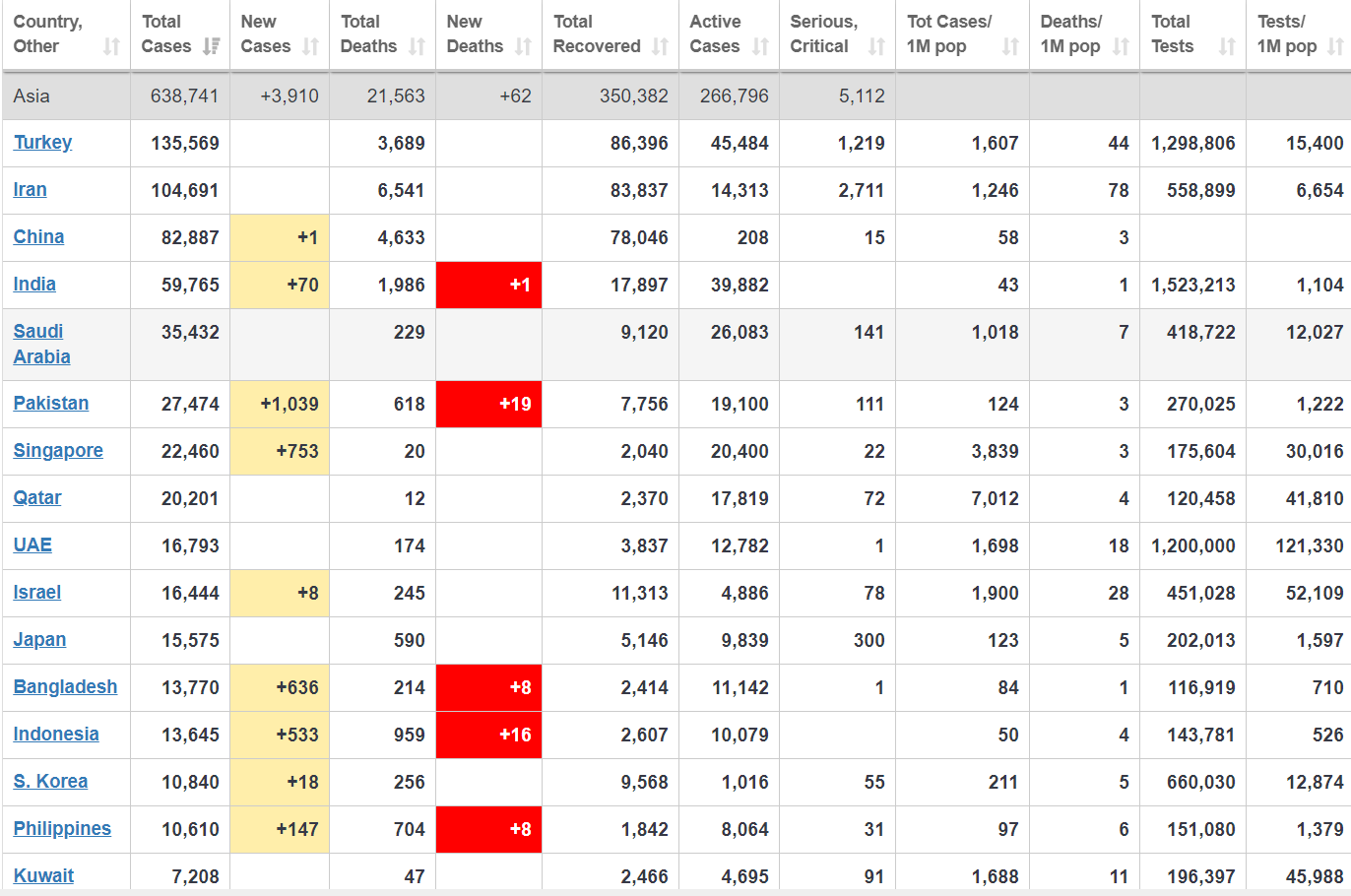
table\_html = str(html\_soup)

type(table\_html)

from IPython.core.display import HTML

HTML(table\_html)

OUT []:



IN []:

#total cases in asia

data = []

table\_rows=soup.find\_all('tr')

for tr in table\_rows:

td=tr.find\_all('td')

row=[i.text for i in td]

data.append(row)

data[4]

OUT []:

['\nAsia\n',

'643,867',

'+9,036',

'21,628',

'+127',

'353,106',

'269,133',

'5,101',

'',

'',

'',

'',

'Asia']

IN []:

#extracting all the columns of the table

keys = [key.get\_text() for key in html\_soup[0].find\_all('th')]

keys = keys[:7]

keys

OUT []:

['Country,Other',

'TotalCases',

'NewCases',

'TotalDeaths',

'NewDeaths',

'TotalRecovered',

'ActiveCases']

IN []:

#calculating the index value of 'India'

values = [ind.get\_text().replace('\n','') for ind in html\_soup[0].find\_all("td")[:]]

values.index('India')

#pair of values from index 260 to 260+(the total number of columns; 12)

values = values[260:272][:7]

values

OUT []:

['India', '59,881', '+186', '1,990 ', '+5', '17,956', '39,935']

IN []:

data\_generated = {}

for key in keys:

for value in values:

data\_generated[key] = value

values.remove(value)

break

data\_generated

OUT []:

{'Country,Other': 'India',

'TotalCases': '59,881',

'NewCases': '+186',

'TotalDeaths': '1,990 ',

'NewDeaths': '+5',

'TotalRecovered': '17,956',

'ActiveCases': '39,935'}