# Puzzle-1

To count the row of csv file

import csv

file = open(“csv-sample.csv”)

reader = csv.reader(file)

lines = len(list(reader))

print(lines)

# Puzzle-sample puzzle-1

Test\_puzzle is the function which is used to check the correct number. We receive number, check number, correct\_num, correct\_num\_position as an argument. Then we send each number to the test\_puzzle function. Inside function we check whether the placement of the number is correct or not. If number is correct then zfill is inserted with 000 at first and we push the correct number to the lnumber. Only if the position and number are correct.

def test\_puzzle(number,checknumber,correct\_num,correct\_num\_position):

count\_correct\_num = 0

count\_correct\_num\_position = 0

lnumber = str(number).zfill(len(checknumber))

for i in range(len(checknumber)):

if lnumber[i] in checknumber:

count\_correct\_num += 1

if lnumber[i] == checknumber[i]:

count\_correct\_num\_position +=1

return correct\_num == count\_correct\_num and correct\_num\_position == count\_correct\_num\_position

for number in range(1000):

if test\_puzzle(number,'147',1,0) and test\_puzzle(number,'189',1,1) and test\_puzzle(number,'964',2,0) and test\_puzzle(number,'523',0,0) and test\_puzzle(number,'286',1,0):

print("Solution to puzzle: " + str(number).zfill(3))

# Puzzle –Sample puzzle-2

Test\_puzzle is the function which is used to check the correct number. We receive number, check number, correct\_num, correct\_num\_position as an argument. Then we send each number to the test\_puzzle function. Inside function we check whether the placement of the number is correct or not. If number is correct then zfill is inserted with 000 at first and we push the correct number to the lnumber. Only if the position and number are correct.

def test\_puzzle(number,checknumber,correct\_num,correct\_num\_position):

count\_correct\_num = 0

count\_correct\_num\_position = 0

lnumber = str(number).zfill(len(checknumber))

for i in range(len(checknumber)):

if lnumber[i] in checknumber:

count\_correct\_num += 1

if lnumber[i] == checknumber[i]:

count\_correct\_num\_position +=1

return correct\_num == count\_correct\_num and \

correct\_num\_position == count\_correct\_num\_position

for number in range(1000):

if test\_puzzle(number,'682',1,1) and test\_puzzle(number,'614',1,0) and test\_puzzle(number,'206',2,0) and test\_puzzle(number,'738',0,0) and test\_puzzle(number,'380',1,0):

print("Solution to puzzle: " + str(number).zfill(3))