Weeks 1 & 2 Exercises

Name : Ragunath Gunasekaran

1. Create a Jupyter notebook where you create a list, iterate over the list and sort your results, generate random numbers, add to the list, and then print your results

In [1]:

```python
# Create a list
listname1 = [1,60,80,35,49,23,68,85,24,76,38,92]
print("\n Elements in the listname1 List are :")
print(listname1)

# iterate over the list by using while and Forloop
# Geting the length of the list
lengthoflist = len(listname1)
i = 0

# Sorting the List
listname1.sort()

print("############ While Loop - Iteration #########################")
while i < lengthoflist:
    print(listname1[i])
    i += 1
'''
print("############ For Loop #########################")
for i in range(lengthoflist):
    print(listname1[i])
'''
```

```
 Elements in the listname1 List are :
[1, 60, 80, 35, 49, 23, 68, 85, 24, 76, 38, 92]
############ While Loop - Iteration #########################
1
23
24
35
38
49
60
68
76
80
85
92
```

Out[1]:

```
'\nprint("############ For Loop #########################")   \nfor i in range
(lengthoflist):\n    print(listname1[i])\n'
```

In [2]:

```
import random

#generate random numbers
limitofnumbers = 30
listname2 = [random.randint(0, limitofnumbers) for x in range(0, limitofnumb

# using list.extend() to concat
listname1.extend(listname2)
listname1.sort()

# Printing concatenated list
print ("Concatenated the two lists - with Duplicates")
print (str(listname1))
# Removing Duplicates
print ("Concatenated the two lists - Removed Duplicates")
print (list(dict.fromkeys(listname1)))
```

```
Concatenated the two lists - with Duplicates
[0, 0, 1, 1, 2, 2, 3, 4, 5, 6, 7, 7, 7, 8, 8, 9, 10, 12, 14, 16, 17, 20, 20, 2
3, 23, 24, 24, 24, 25, 26, 26, 27, 29, 35, 38, 49, 60, 68, 76, 80, 85, 92]
Concatenated the two lists - Removed Duplicates
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 16, 17, 20, 23, 24, 25, 26, 27, 29,
35, 38, 49, 60, 68, 76, 80, 85, 92]
```

2. Create a line chart with Matplotlib and the following data file.

a. Data file: world-population.xlsm

b. (Hint: Python for Data Analysis: Page 19-50 & Data Wrangling with Python: Preface)

In [3]:

```
import pandas as pd

# Data worldpopulation Data
WorldPopulation_DF = pd.read_excel('C:/Users/ragun/Documents/GitHub/DSC540/W

#verify any Nulls
WorldPopulation_DF.isnull().sum()
```

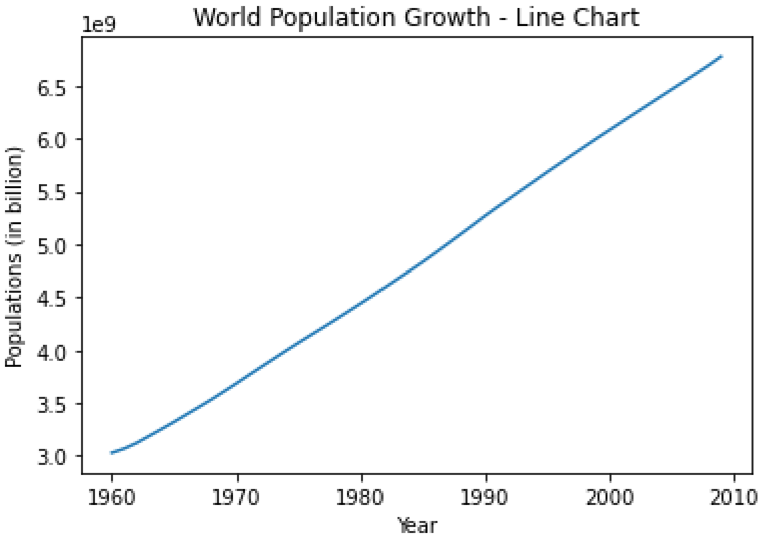
Out[3]:

```
Year          0
Population    0
dtype: int64
```

In [4]:
```
import matplotlib.pyplot as plt

plt.plot(WorldPopulation_DF.Year, WorldPopulation_DF.Population)
plt.xlabel("Year")  # add X-axis label
plt.ylabel("Populations (in billion)")  # add Y-axis label
plt.title("World Population Growth - Line Chart")  # add title
plt.show()

```



3. Complete the following activities:

a. Data Wrangling with Python: Activity 1 page 17

Create a list of random numbers and then create another list from this one whose elements are divisible by three. Also repeat the experiment few times (at least three times) and calculate the arithimetic mean of the difference of length of the two lists

Task 1 - Create list of 100 Random Numbers

In [5]:

```
LimitNumbers = 100
List_RandomNumber = [random.randint(0, LimitNumbers) for x in range(0, Limit
List_RandomNumber
```

Out[5]:

```
[32,
 72,
 35,
 50,
 78,
 29,
 76,
 46,
 44,
 13,
 56,
 78,
 4,
 57,
 100,
 58,
 65,
 19,
 28,
 73,
 92,
 33,
 74,
 55,
 29,
 62,
 87,
 84,
 73,
 58,
 83,
 85,
 71,
 52,
 28,
 40,
 54,
 53,
 6,
 67,
 88,
 6,
 40,
 22,
 72,
 0,
 39,
 28,
 78,
 93,
 21,
 18,
 53,
 82,
 28,
 35,
 85,
 40,
 22,
 100,
 60,
 89,
 99,
 45,
 66,
 15,
 70,
 66,
 89,
 10,
 57,
 83,
 21,
 1,
 5,
 33,
 78,
 45,
 64,
 40,
 84,
 83,
 68,
 87,
 70,
 83,
 70,
 75,
 34,
 30,
 33,
 69,
 38,
 10,
 72,
 86,
 34,
 41,
 71,
 99]
```

Task-2

Generate a second list from the above one. The condition of membership in the second list is divisibility by 3.

In [6]:

```
List_DivByThree=[number for number in List_RandomNumber if number % 3 == 0]
List_DivByThree
```

Out[6]:

```
[72,
 78,
 78,
 57,
 33,
 87,
 84,
 54,
 6,
 6,
 72,
 0,
 39,
 78,
 93,
 21,
 18,
 60,
 99,
 45,
 66,
 15,
 66,
 57,
 21,
 33,
 78,
 45,
 84,
 87,
 75,
 30,
 33,
 69,
 72,
 99]
```

Task-3

- Use the `len` function to measure the length of the first list and the second list
- Store both in two different variables
- Calculate the difference of length between them

In [7]:

```
DiffOfLen = len(List_RandomNumber) - len(List_DivByThree)
DiffOfLen
```

Out[7]:

```
64
```

Task-4

- Pack `Task-2` and `Task-3` in a single while loop and perform them few times in such a way that at the end you have a list with difference of length
- End the while loop when desired number of experiments are finished (at least three, please feel free to do more)
- Calculate the arithmetic mean (common average) on the difference of length that you have. (How to sum all values of a list?)

In [8]:

```
No_of_expts=10

ListDifference =[]

for i in range(0, No_of_expts):
    ListRandom = [random.randint(0, No_of_expts) for x in range(0, No_of_exp
    ListDivByThree = [number for number in ListRandom if number % 3 == 0]
    difference = len(ListRandom) - len(ListDivByThree)
    ListDifference.append(difference)

ListDifference
```

Out[8]:

```
[7, 7, 9, 3, 7, 6, 5, 8, 4, 4]
```

In [9]:

```
from statistics import mean

# Calculate the mean of the list - ListDifference
mean(ListDifference)

# Ref - https://www.geeksforgeeks.org/find-average-list-python/
```

Out[9]:

```
6
```

b. Data Wrangling with Python: Activity 2 - Analyze Multiline String and Generate the unique Word Count

Task 1 - Create multiline_text and assign p and p file data

In [10]:

```
multiline_text = """It is a truth universally acknowledged, that a single ma

However little known the feelings or views of such a man may be on his first

"My dear Mr. Bennet," said his lady to him one day, "have you heard that Net

Mr. Bennet replied that he had not.

"But it is," returned she; "for Mrs. Long has just been here, and she told m

Mr. Bennet made no answer.

"Do you not want to know who has taken it?" cried his wife impatiently.

"You want to tell me, and I have no objection to hearing it."

This was invitation enough.

"Why, my dear, you must know, Mrs. Long says that Netherfield is taken by a

"What is his name?"

"Bingley."

"Is he married or single?"

"Oh! Single, my dear, to be sure! A single man of large fortune; four or fiv

"How so? How can it affect them?"

"My dear Mr. Bennet," replied his wife, "how can you be so tiresome! You mus

"Is that his design in settling here?"

"Design! Nonsense, how can you talk so! But it is very likely that he may fa

"I see no occasion for that. You and the girls may go, or you may send them

"My dear, you flatter me. I certainly have had my share of beauty, but I do

"In such cases, a woman has not often much beauty to think of."

"But, my dear, you must indeed go and see Mr. Bingley when he comes into the

"It is more than I engage for, I assure you."

"But consider your daughters. Only think what an establishment it would be f

"You are over-scrupulous, surely. I dare say Mr. Bingley will be very glad t

"I desire you will do no such thing. Lizzy is not a bit better than the othe

"They have none of them much to recommend them," replied he; "they are all s

"Mr. Bennet, how can you abuse your own children in such a way? You take del

```

```
"You mistake me, my dear. I have a high respect for your nerves. They are my

"Ah, you do not know what I suffer."

"But I hope you will get over it, and live to see many young men of four tho

"It will be no use to us, if twenty such should come, since you will not vis

"Depend upon it, my dear, that when there are twenty, I will visit them all.

Mr. Bennet was so odd a mixture of quick parts, sarcastic humour, reserve, a
```

Task 2 - Find the length and type of the multiline_text string

In [11]:

```
type(multiline_text)
```

Out[11]:

```
str
```

In [12]:

```
len(multiline_text)
```

Out[12]:

```
4475
```

Task 3 - Remove all new lines by using Replace function

In [13]:

```
multiline_text = multiline_text.replace('\n', "")
```

Task 4 - Find all words in string by using split function

In [14]:
```
import re
#Removing special chars
multiline_text = re.sub(r'[?|$|.|!|"|,|;|:]',r'',multiline_text)

#Split the words based on the Space

list_of_words = multiline_text.split()
list_of_words

```

Out[14]:
```
['It',
 'is',
 'a',
 'truth',
 'universally',
 'acknowledged',
 'that',
 'a',
 'single',
 'man',
 'in',
 'possession',
 'of',
 'a',
 'good',
 'fortune',
 'must',
 'be',
 'in',
```

Task 5 - create list only contains unique values (Remove duplicates)

In [15]:
```
unique_words = dict.fromkeys(list_of_words)
len(list(unique_words.keys()))
```

Out[15]:
```
349
```

Task 6 - Count number of times the unique word apperared

In [16]:

```
for word in list_of_words:
    if unique_words[word] is None:
        unique_words[word] = 1
    else:
        unique_words[word] += 1
unique_words

#
```

```
 'wifeHowever': 1,
 'little': 3,
 'known': 1,
 'the': 17,
 'feelings': 1,
 'or': 5,
 'views': 1,
 'such': 5,
 'may': 5,
 'on': 3,
 'his': 11,
 'first': 1,
 'entering': 1,
 'neighbourhood': 1,
 'this': 1,
 'so': 8,
 'well': 1,
 'fixed': 1,
 'minds': 1,
 'surrounding': 1,
```

Task 7 - Top 25

In [17]:

```
top_words = sorted(unique_words.items(), key=lambda key_val_tuple: key_val_t
top_words[:25]

# Ref - https://www.geeksforgeeks.org/python-n-largest-values-in-dictionary/
```

Out[17]:

```
[('of', 28),
 ('you', 23),
 ('to', 22),
 ('a', 20),
 ('the', 17),
 ('and', 16),
 ('that', 15),
 ('I', 15),
 ('is', 12),
 ('for', 12),
 ('in', 11),
 ('be', 11),
 ('his', 11),
 ('he', 11),
 ('my', 10),
 ('it', 9),
 ('will', 9),
 ('so', 8),
 ('dear', 8),
 ('was', 8),
 ('are', 8),
 ('them', 8),
 ('must', 7),
 ('have', 7),
 ('no', 7)]
```

c. Data Wrangling with Python: Activity 3 page 49

Task 1

Look up the definition of `permutations` and `dropwhile` from itertools.

In [18]:

```
from itertools import permutations, dropwhile
```

In [19]:

```
permutations?
```

In [20]:

```
dropwhile?
```

Task 2

Write an expression to generate all the possible three digit numbers using 0, 1, and 2

In [21]:

```
permutations(range(3))
```

Out[21]:

```
<itertools.permutations at 0x1ad1a005450>
```

Task 3

Loop over the iterator expression you generated before. Use print to print each element returned by the iterator. Use `assert` and `isinstance` to make sure that the elements are of type tuple

In [22]:

```
for number_tuple in permutations(range(3)):
    print(number_tuple)
    assert isinstance(number_tuple, tuple)
```

```
(0, 1, 2)
(0, 2, 1)
(1, 0, 2)
(1, 2, 0)
(2, 0, 1)
(2, 1, 0)
```

Task 4

Write the loop again. But this time use `dropwhile` with a lambda expression to drop any leading zeros from the tuples. As an example `(0, 1, 2)` will become `[0, 2]` . Also cast the output of the dropwhile to a list.

Extra task can be to check the actual type that dropwhile returns without the casting asked above

In [23]:

```
for number_tuple in permutations(range(3)):
    print(list(dropwhile(lambda x: x <= 0, number_tuple)))
```

```
[1, 2]
[2, 1]
[1, 0, 2]
[1, 2, 0]
[2, 0, 1]
[2, 1, 0]
```

d. Data Wrangling with Python: Activity 4 - Activity 04 Design your own CSV parser

In [24]:

```
from itertools import zip_longest
```

In [25]:

```
def dict_from_csv_line(header, line):
    # Zip them
    zipped_line = zip_longest(header, line, fillvalue=None)
    # Use dict comprehension to generate the final dict
    ret_dict = {kv[0]: kv[1] for kv in zipped_line}
    return ret_dict
```

In [29]:

```
### For this Activity we are not gonna make smaller subtasks like the ones b
### Write the whole activity's code bellow this line

# copied the file and pasted in my local

with open("C:/Users/ragun/Documents/GitHub/DSC540/Weeks 1 & 2/sales_record.c
    firstLine = fd.readline()
    header = firstLine.replace("\n", "").split(",")
    for i, line in enumerate(fd):
        # Here we loop over the first 10 lines in order to not to make the o
        line = line.replace("\n", "").split(",")
        dict_output = dict_from_csv_line(header, line)
        print(dict_output)

        if i > 5:
            break
```

```
{'Region': 'Central America and the Caribbean', 'Country': 'Antigua and Barbuda
', 'Item Type': 'Baby Food', 'Sales Channel': 'Online', 'Order Priority': 'M',
'Order Date': '12/20/2013', 'Order ID': '957081544', 'Ship Date': '1/11/2014',
'Units Sold': '552', 'Unit Price': '255.28', 'Unit Cost': '159.42', 'Total Reve
nue': '140914.56', 'Total Cost': '87999.84', 'Total Profit': '52914.72'}
{'Region': 'Central America and the Caribbean', 'Country': 'Panama', 'Item Typ
e': 'Snacks', 'Sales Channel': 'Offline', 'Order Priority': 'C', 'Order Date':
'7/5/2010', 'Order ID': '301644504', 'Ship Date': '7/26/2010', 'Units Sold': '2
167', 'Unit Price': '152.58', 'Unit Cost': '97.44', 'Total Revenue': '330640.8
6', 'Total Cost': '211152.48', 'Total Profit': '119488.38'}
{'Region': 'Europe', 'Country': 'Czech Republic', 'Item Type': 'Beverages', 'Sa
les Channel': 'Offline', 'Order Priority': 'C', 'Order Date': '9/12/2011', 'Ord
er ID': '478051030', 'Ship Date': '9/29/2011', 'Units Sold': '4778', 'Unit Pric
e': '47.45', 'Unit Cost': '31.79', 'Total Revenue': '226716.10', 'Total Cost':
'151892.62', 'Total Profit': '74823.48'}
{'Region': 'Asia', 'Country': 'North Korea', 'Item Type': 'Cereal', 'Sales Chan
nel': 'Offline', 'Order Priority': 'L', 'Order Date': '5/13/2010', 'Order ID':
'892599952', 'Ship Date': '6/15/2010', 'Units Sold': '9016', 'Unit Price': '20
5.70', 'Unit Cost': '117.11', 'Total Revenue': '1854591.20', 'Total Cost': '105
5863.76', 'Total Profit': '798727.44'}
{'Region': 'Asia', 'Country': 'Sri Lanka', 'Item Type': 'Snacks', 'Sales Channe
l': 'Offline', 'Order Priority': 'C', 'Order Date': '7/20/2015', 'Order ID': '5
71902596', 'Ship Date': '7/27/2015', 'Units Sold': '7542', 'Unit Price': '152.5
8', 'Unit Cost': '97.44', 'Total Revenue': '1150758.36', 'Total Cost': '734892.
48', 'Total Profit': '415865.88'}
{'Region': 'Middle East and North Africa', 'Country': 'Morocco', 'Item Type':
'Personal Care', 'Sales Channel': 'Offline', 'Order Priority': 'L', 'Order Dat
e': '11/8/2010', 'Order ID': '412882792', 'Ship Date': '11/22/2010', 'Units Sol
d': '48', 'Unit Price': '81.73', 'Unit Cost': '56.67', 'Total Revenue': '3923.0
4', 'Total Cost': '2720.16', 'Total Profit': '1202.88'}
{'Region': 'Australia and Oceania', 'Country': 'Federated States of Micronesi
a', 'Item Type': 'Clothes', 'Sales Channel': 'Offline', 'Order Priority': 'H',
'Order Date': '3/28/2011', 'Order ID': '932776868', 'Ship Date': '5/10/2011',
'Units Sold': '8258', 'Unit Price': '109.28', 'Unit Cost': '35.84', 'Total Reve
nue': '902434.24', 'Total Cost': '295966.72', 'Total Profit': '606467.52'}
```

In [ ]:

```

```