Assignment: Week 1 & Week 2 Exercise, What is Data Wrangling

Name: Prashant Raghuwanshi

Date: 9/12/2021

Course: DSC540-T301 Data Preparation (2221-1)

1. Install the latest versions of either Docker or Anaconda. Your book Data Wrangling with Python uses Docker, however, you are welcome to use whichever distributer you feel comfortable with. 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.

Docker version installed : Docker version 20.10.8, build 3967b7d

packt-data-wrangling-base image is ececuting in container d6a94cf78c6a rcshubhadeep/packt-data-wrangling-base

```
In [1]: # create list
        list_example = [21, 27, 34, 76, 90, 25, -29]
        list_example
```

```
Out[1]: [21, 27, 34, 76, 90, 25, -29]
```

```
In [2]: # iterate over list
        for i, element in enumerate(list_example):
            print("list_1[{}]=>{}".format(i, element))
```

```
list_1[0]=>21
list_1[1]=>27
list_1[2]=>34
list_1[3]=>76
list_1[4]=>90
list_1[5]=>25
list_1[6]=>-29
```

```
In [3]: # sort list
        list_example.sort(reverse=True)
        list_example
```

```
Out[3]: [90, 76, 34, 27, 25, 21, -29]
```

In [4]:

```
# generate random numbers
import random
rand_num = random.randint(0, 90)
print("rand_num=", rand_num)
# add generated random number to list
list_example.append(rand_num)
list_example
```

```
rand_num= 61
```

Out[4]:

```
[90, 76, 34, 27, 25, 21, -29, 61]
```

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 [5]:

```
# import panda and read source file into dataframe
import pandas as pd
world_pop_df = pd.read_excel("/app/world-population.xlsm", sheet_name="world-p
opulation")
world_pop_df.head(5)
```

Out[5]:

	Year	Population
0	1960	3028654024
1	1961	3068356747
2	1962	3121963107
3	1963	3187471383
4	1964	3253112403

In [6]:

```
# import matplotlib and creating line graph from world population source file
import matplotlib.pyplot as plt
plt.figure()
plt.plot(world_pop_df['Year'], world_pop_df['Population'])
plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Yearly World-Population graph")
plt.show()
```

```
<Figure size 640x480 with 1 Axes>
```

3. Complete the following activities:

a. Data Wrangling with Python: Activity 1 page 17

1) create a list of 100 random number

In [7]:

```
import random
LIMIT = 100
rand_num_list = [random.randint(0, LIMIT) for x in range(0, LIMIT)]
```

2) create new list from above random number with number are divisible by 3

In [8]:

```
rand_num_div3_list = [a for a in rand_num_list if a % 3 == 0]
```

3) calculate the length of these two list and store the difference in a new variable

In [9]:

```
length_rand_num_list = len(rand_num_list)
length_rand_num_div3_list = len(rand_num_div3_list)
diff_rand_num_list = length_rand_num_list - length_rand_num_div3_list
diff_rand_num_list
```

```
Out[9]: 63
```

4) using loop, perform steps 2 & 3 and find the difference variable three times

In [10]:

```
NUMBER_OF_iterations = 3
diff_rand_num_list1 = []
for i in range(0, NUMBER_OF_iterations):
    random_number_list = [random.randint(0, LIMIT) for x in range(0, LIMIT)]
    list_with_divisible_by_3 = [a for a in random_number_list if a % 3 == 0]

    length_of_random_list = len(random_number_list)
    length_of_3_divisible_list = len(list_with_divisible_by_3)
    difference = length_of_random_list - length_of_3_divisible_list
    diff_rand_num_list1.append(difference)

diff_rand_num_list1
```

```
Out[10]: [74, 67, 67]
```

5) find the arithmetic means of these three difference values

```
In [11]: means_diff = sum(diff_rand_num_list1) / float(len(diff_rand_num_list1))
         means_diff
```

```
Out[11]: 69.33333333333333
```

b. Data Wrangling with Python: Activity 2 page 31

In [12]:

```
# create a multiline_text variable by copying the test from the first chapter
 of pride and prejudice
mutiline_text = """It is a truth universally acknowledged, that a single man i
n possession of a good fortune, must be in want of a wife.

However little known the feelings or views of such a man may be on his first e
ntering a neighbourhood, this truth is so well fixed in the minds of the surro
unding families, that he is considered the rightful property of some one or ot
her of their daughters.

"My dear Mr. Bennet," said his lady to him one day, "have you heard that Nethe
rfield Park is let at last?"

Mr. Bennet replied that he had not.

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

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 yo
ung man of large fortune from the north of England; that he came down on Monda
y in a chaise and four to see the place, and was so much delighted with it, th
at he agreed with Mr. Morris immediately; that he is to take possession before
Michaelmas, and some of his servants are to be in the house by the end of next
week."

"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 five
 thousand a year. What a fine thing for our girls!"

"How so? How can it affect them?"

"My dear Mr. Bennet," replied his wife, "how can you be so tiresome! You must
 know that I am thinking of his marrying one of them."

"Is that his design in settling here?"

"Design! Nonsense, how can you talk so! But it is very likely that he may fall
in love with one of them, and therefore you must visit him as soon as he come
s."

"I see no occasion for that. You and the girls may go, or you may send them by
themselves, which perhaps will be still better, for as you are as handsome as
 any of them, Mr. Bingley may like you the best of the party."
```

```
"My dear, you flatter me. I certainly have had my share of beauty, but I do no
t pretend to be anything extraordinary now. When a woman has five grown-up dau
ghters, she ought to give over thinking of her own beauty."

"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 n
eighbourhood."

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

"But consider your daughters. Only think what an establishment it would be for
one of them. Sir William and Lady Lucas are determined to go, merely on that a
ccount, for in general, you know, they visit no newcomers. Indeed you must go,
for it will be impossible for us to visit him if you do not."

"You are over-scrupulous, surely. I dare say Mr. Bingley will be very glad to
 see you; and I will send a few lines by you to assure him of my hearty consen
t to his marrying whichever he chooses of the girls; though I must throw in a
 good word for my little Lizzy."

"I desire you will do no such thing. Lizzy is not a bit better than the other
s; and I am sure she is not half so handsome as Jane, nor half so good-humoure
d as Lydia. But you are always giving her the preference."

"They have none of them much to recommend them," replied he; "they are all sil
ly and ignorant like other girls; but Lizzy has something more of quickness th
an her sisters."

"Mr. Bennet, how can you abuse your own children in such a way? You take delig
ht in vexing me. You have no compassion for my poor nerves."

"You mistake me, my dear. I have a high respect for your nerves. They are my o
ld friends. I have heard you mention them with consideration these last twenty
years at least."

"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 thous
and a year come into the neighbourhood."

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

"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, and
caprice, that the experience of three-and-twenty years had been insufficient t
o make his wife understand his character. Her mind was less difficult to devel
op. She was a woman of mean understanding, little information, and uncertain t
emper. When she was discontented, she fancied herself nervous. The business of
her life was to get her daughters married; its solace was visiting and news.
 """
len(mutiline_text)
```

```
Out[12]: 4475
```

In [13]:

```
# remove all newlines and symboles using the replace all function
mutiline_text = mutiline_text.replace('\n', "")
# remove special chars, punctuation etc.
import re
cleaned_multiline_text = re.sub(r'[?|$|.|!|"|,|;|:]',r'',mutiline_text)
```

In [14]:

```
# find all of the words in text by using split function
list_of_words = cleaned_multiline_text.split()
len(list_of_words)
```

```
Out[14]: 814
```

In [15]:

```
# Use set to get unique words
unique_words_as_list = list(set(list_of_words))
len(unique_words_as_list)
```

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

In [16]:

```
# Use dict to do the same
unique_words_as_dict = dict.fromkeys(list_of_words)
len(list(unique_words_as_dict.keys()))
```

```
Out[16]: 349
```

In [17]:

```
# count the number of times the nuique words has apperead in the list using the key and value in dict
for word in list_of_words:
    if unique_words_as_dict[word] is None:
        unique_words_as_dict[word] = 1
    else:
        unique_words_as_dict[word] += 1
unique_words_as_dict
```

```
Out[17]: {'It': 1,
          'is': 12,
          'a': 20,
          'truth': 2,
          'universally': 1,
          'acknowledged': 1,
          'that': 15,
          'single': 2,
          'man': 4,
          'in': 11,
          'possession': 2,
          'of': 28,
          'good': 2,
          'fortune': 3,
          'must': 7,
          'be': 11,
          'want': 3,
          '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,
          'families': 1,
          'he': 11,
          'considered': 1,
          'rightful': 1,
          'property': 1,
          'some': 2,
          'one': 5,
          'other': 2,
          'their': 1,
          'daughtersMy': 1,
          'dear': 8,
          'Mr': 6,
          'Bennet': 6,
          'said': 1,
          'lady': 1,
          'to': 22,
          'him': 4,
          'day': 1,
          'have': 7,
          'you': 23,
          'heard': 2,
          'Netherfield': 2,
          'Park': 1,
          'let': 1,
          'at': 2,
          'lastMr': 1,
          'replied': 3,
          'had': 3,
          'notBut': 1,
          'it': 9,
          'returned': 1,
          'she': 6,
          'for': 12,
          'Mrs': 2,
          'Long': 2,
          'has': 5,
          'just': 1,
          'been': 2,
          'here': 1,
          'and': 16,
          'told': 1,
          'me': 5,
          'all': 2,
          'about': 1,
          'itMr': 1,
          'made': 1,
          'no': 7,
          'answerDo': 1,
          'not': 7,
          'know': 5,
          'who': 1,
          'taken': 2,
          'cried': 1,
          'wife': 3,
          'impatientlyYou': 1,
          'tell': 1,
          'I': 15,
          'objection': 1,
          'hearing': 1,
          'itThis': 1,
          'was': 8,
          'invitation': 1,
          'enoughWhy': 1,
          'my': 10,
          'says': 1,
          'by': 4,
          'young': 2,
          'large': 2,
          'from': 1,
          'north': 1,
          'England': 1,
          'came': 1,
          'down': 1,
          'Monday': 1,
          'chaise': 1,
          'four': 3,
          'see': 5,
          'place': 1,
          'much': 3,
          'delighted': 1,
          'with': 4,
          'agreed': 1,
          'Morris': 1,
          'immediately': 1,
          'take': 2,
          'before': 1,
          'Michaelmas': 1,
          'servants': 1,
          'are': 8,
          'house': 1,
          'end': 1,
          'next': 1,
          'weekWhat': 1,
          'nameBingleyIs': 1,
          'married': 2,
          'singleOh': 1,
          'Single': 1,
          'sure': 2,
          'A': 1,
          'five': 2,
          'thousand': 2,
          'year': 2,
          'What': 1,
          'fine': 1,
          'thing': 2,
          'our': 1,
          'girlsHow': 1,
          'How': 1,
          'can': 4,
          'affect': 1,
          'themMy': 1,
          'how': 3,
          'tiresome': 1,
          'You': 4,
          'am': 2,
          'thinking': 2,
          'marrying': 2,
          'themIs': 1,
          'design': 1,
          'settling': 1,
          'hereDesign': 1,
          'Nonsense': 1,
          'talk': 1,
          'But': 2,
          'very': 2,
          'likely': 1,
          'fall': 1,
          'love': 1,
          'them': 8,
          'therefore': 1,
          'visit': 5,
          'as': 7,
          'soon': 1,
          'comesI': 1,
          'occasion': 1,
          'girls': 3,
          'go': 4,
          'send': 2,
          'themselves': 1,
          'which': 1,
          'perhaps': 1,
          'will': 9,
          'still': 1,
          'better': 2,
          'handsome': 2,
          'any': 1,
          'Bingley': 3,
          'like': 2,
          'best': 1,
          'partyMy': 1,
          'flatter': 1,
          'certainly': 1,
          'share': 1,
          'beauty': 2,
          'but': 2,
          'do': 4,
          'pretend': 1,
          'anything': 1,
          'extraordinary': 1,
          'now': 1,
          'When': 2,
          'woman': 3,
          'grown-up': 1,
          'daughters': 3,
          'ought': 1,
          'give': 1,
          'over': 2,
          'her': 5,
          'own': 2,
          'beautyIn': 1,
          'cases': 1,
          'often': 1,
          'think': 2,
          'ofBut': 1,
          'indeed': 1,
          'when': 2,
          'comes': 1,
          'into': 2,
          'neighbourhoodIt': 2,
          'more': 2,
          'than': 3,
          'engage': 1,
          'assure': 2,
          'youBut': 1,
          'consider': 1,
          'your': 3,
          'Only': 1,
          'what': 2,
          'an': 1,
          'establishment': 1,
          'would': 1,
          'Sir': 1,
          'William': 1,
          'Lady': 1,
          'Lucas': 1,
          'determined': 1,
          'merely': 1,
          'account': 1,
          'general': 1,
          'they': 2,
          'newcomers': 1,
          'Indeed': 1,
          'impossible': 1,
          'us': 2,
          'if': 2,
          'notYou': 1,
          'over-scrupulous': 1,
          'surely': 1,
          'dare': 1,
          'say': 1,
          'glad': 1,
          'few': 1,
          'lines': 1,
          'hearty': 1,
          'consent': 1,
          'whichever': 1,
          'chooses': 1,
          'though': 1,
          'throw': 1,
          'word': 1,
          'LizzyI': 1,
          'desire': 1,
          'Lizzy': 2,
          'bit': 1,
          'others': 1,
          'half': 2,
          'Jane': 1,
          'nor': 1,
          'good-humoured': 1,
          'Lydia': 1,
          'always': 1,
          'giving': 1,
          'preferenceThey': 1,
          'none': 1,
          'recommend': 1,
          'silly': 1,
          'ignorant': 1,
          'something': 1,
          'quickness': 1,
          'sistersMr': 1,
          'abuse': 1,
          'children': 1,
          'way': 1,
          'delight': 1,
          'vexing': 1,
          'compassion': 1,
          'poor': 1,
          'nervesYou': 1,
          'mistake': 1,
          'high': 1,
          'respect': 1,
          'nerves': 1,
          'They': 1,
          'old': 1,
          'friends': 1,
          'mention': 1,
          'consideration': 1,
          'these': 1,
          'last': 1,
          'twenty': 3,
          'years': 2,
          'leastAh': 1,
          'sufferBut': 1,
          'hope': 1,
          'get': 2,
          'live': 1,
          'many': 1,
          'men': 1,
          'come': 2,
          'use': 1,
          'should': 1,
          'since': 1,
          'themDepend': 1,
          'upon': 1,
          'there': 1,
          'allMr': 1,
          'odd': 1,
          'mixture': 1,
          'quick': 1,
          'parts': 1,
          'sarcastic': 1,
          'humour': 1,
          'reserve': 1,
          'caprice': 1,
          'experience': 1,
          'three-and-twenty': 1,
          'insufficient': 1,
          'make': 1,
          'understand': 1,
          'character': 1,
          'Her': 1,
          'mind': 1,
          'less': 1,
          'difficult': 1,
          'develop': 1,
          'She': 1,
          'mean': 1,
          'understanding': 1,
          'information': 1,
          'uncertain': 1,
          'temper': 1,
          'discontented': 1,
          'fancied': 1,
          'herself': 1,
          'nervous': 1,
           'The': 1,
           'business': 1,
           'life': 1,
           'its': 1,
           'solace': 1,
           'visiting': 1,
           'news': 1}
```

```
In [18]: # find the top 25 words from the unique words that you have found using the sl
         ice fucntion
         top_words = sorted(unique_words_as_dict.items(), key=lambda key_val_tuple: key
         _val_tuple[1], reverse=True)
         top_words[:25]
```

```
Out[18]: [('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

```
In [19]: # lookup the defination of permutations and dropwhile from iteration tools
         from itertools import permutations, dropwhile
```

```
In [20]: permutations?
```

```
In [21]: dropwhile?
```

In [22]:
```
# write an expression to generate all possivbles three digits numbers using 0,
1, 2
permutations(range(3))
```

Out[22]:
```
<itertools.permutations at 0x7f6d770411a8>
```

In [23]:
```
# loop over the iteratio expression you generated before. print each eliment t
hats returned by itterator.
# use assert and isinstance to make sure that the elementsare of tuple type
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)
```

In [24]:
```
# write the loop again using dropwhile with lambda expression to drop any lead
ing zeros from tuple
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]
```

In [25]:
```
# write all the logic you wrote before but this time wite a seperate function
 where you will be passing
# the list generated from dropwhile and the function will return the whole bum
ber contained in the list
import math
def convert_to_number(number_stack):
    final_number = 0
    for i in range(0, len(number_stack)):
        final_number += (number_stack.pop() * (math.pow(10, i)))
    return final_number
```

In [26]:
```
for number_tuple in permutations(range(3)):
    number_stack = list(dropwhile(lambda x: x <= 0, number_tuple))
    print(convert_to_number(number_stack))
```

```
12.0
21.0
102.0
120.0
201.0
210.0
```

d. Data Wrangling with Python: Activity 4 page 59

```
In [27]: # import zip_longest from itertools
         from itertools import zip_longest
```

```
In [28]: # deifne the return_dict function so that it contains headers ,line and fillva
         lue as none anbd add it to a dict
         def return_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]:

```
# open the accompanying sales records file and read lines and fill up the keys
value
with open("/app/sales_record.csv", "r") as fd:
    first_line = fd.readline()
    header = first_line.replace("\n", "").split(",")
    for i, line in enumerate(fd):
        # Here we loop over the first 10 lines in order to not to make the out
put too big
        line = line.replace("\n", "").split(",")
        d = return_dict_from_csv_line(header, line)
        print(d)
        if i > 10:
            break
```

```
{'Region': 'Central America and the Caribbean', 'Country': 'Antigua and Barbu
da ', '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', 'T
otal Revenue': '140914.56', 'Total Cost': '87999.84', 'Total Profit': '52914.
72'}
{'Region': 'Central America and the Caribbean', 'Country': 'Panama', 'Item Ty
pe': 'Snacks', 'Sales Channel': 'Offline', 'Order Priority': 'C', 'Order Dat
e': '7/5/2010', 'Order ID': '301644504', 'Ship Date': '7/26/2010', 'Units Sol
d': '2167', 'Unit Price': '152.58', 'Unit Cost': '97.44', 'Total Revenue': '3
30640.86', 'Total Cost': '211152.48', 'Total Profit': '119488.38'}
{'Region': 'Europe', 'Country': 'Czech Republic', 'Item Type': 'Beverages',
'Sales Channel': 'Offline', 'Order Priority': 'C', 'Order Date': '9/12/2011',
'Order ID': '478051030', 'Ship Date': '9/29/2011', 'Units Sold': '4778', 'Uni
t Price': '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 Ch
annel': 'Offline', 'Order Priority': 'L', 'Order Date': '5/13/2010', 'Order I
D': '892599952', 'Ship Date': '6/15/2010', 'Units Sold': '9016', 'Unit Pric
e': '205.70', 'Unit Cost': '117.11', 'Total Revenue': '1854591.20', 'Total Co
st': '1055863.76', 'Total Profit': '798727.44'}
{'Region': 'Asia', 'Country': 'Sri Lanka', 'Item Type': 'Snacks', 'Sales Chan
nel': 'Offline', 'Order Priority': 'C', 'Order Date': '7/20/2015', 'Order I
D': '571902596', 'Ship Date': '7/27/2015', 'Units Sold': '7542', 'Unit Pric
e': '152.58', 'Unit Cost': '97.44', 'Total Revenue': '1150758.36', 'Total Cos
t': '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 Da
te': '11/8/2010', 'Order ID': '412882792', 'Ship Date': '11/22/2010', 'Units
Sold': '48', 'Unit Price': '81.73', 'Unit Cost': '56.67', 'Total Revenue': '3
923.04', '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': '5/10/2
011', 'Units Sold': '8258', 'Unit Price': '109.28', 'Unit Cost': '35.84', 'To
tal Revenue': '902434.24', 'Total Cost': '295966.72', 'Total Profit': '60646
7.52'}
{'Region': 'Europe', 'Country': 'Bosnia and Herzegovina', 'Item Type': 'Cloth
es', 'Sales Channel': 'Online', 'Order Priority': 'M', 'Order Date': '10/14/2
013', 'Order ID': '919133651', 'Ship Date': '11/4/2013', 'Units Sold': '927',
'Unit Price': '109.28', 'Unit Cost': '35.84', 'Total Revenue': '101302.56',
'Total Cost': '33223.68', 'Total Profit': '68078.88'}
{'Region': 'Middle East and North Africa', 'Country': 'Afghanistan', 'Item Ty
pe': 'Clothes', 'Sales Channel': 'Offline', 'Order Priority': 'M', 'Order Dat
e': '8/27/2016', 'Order ID': '579814469', 'Ship Date': '10/5/2016', 'Units So
ld': '8841', 'Unit Price': '109.28', 'Unit Cost': '35.84', 'Total Revenue':
'966144.48', 'Total Cost': '316861.44', 'Total Profit': '649283.04'}
{'Region': 'Sub-Saharan Africa', 'Country': 'Ethiopia', 'Item Type': 'Baby Fo
od', 'Sales Channel': 'Online', 'Order Priority': 'M', 'Order Date': '4/13/20
15', 'Order ID': '192993152', 'Ship Date': '5/7/2015', 'Units Sold': '9817',
'Unit Price': '255.28', 'Unit Cost': '159.42', 'Total Revenue': '2506083.76',
'Total Cost': '1565026.14', 'Total Profit': '941057.62'}
{'Region': 'Middle East and North Africa', 'Country': 'Turkey', 'Item Type':
'Office Supplies', 'Sales Channel': 'Offline', 'Order Priority': 'C', 'Order
Date': '9/25/2013', 'Order ID': '557156026', 'Ship Date': '10/15/2013', 'Unit
s Sold': '3704', 'Unit Price': '651.21', 'Unit Cost': '524.96', 'Total Revenu
e': '2412081.84', 'Total Cost': '1944451.84', 'Total Profit': '467630.00'}
```

```
{'Region': 'Middle East and North Africa', 'Country': 'Oman', 'Item Type': 'C
osmetics', 'Sales Channel': 'Online', 'Order Priority': 'M', 'Order Date':
'5/12/2013', 'Order ID': '741101920', 'Ship Date': '5/17/2013', 'Units Sold':
'7382', 'Unit Price': '437.20', 'Unit Cost': '263.33', 'Total Revenue': '3227
410.40', 'Total Cost': '1943902.06', 'Total Profit': '1283508.34'}
```