Questions to Answer from QUIZ - STAGE A.

1. Select columns ‘Y2017’ and ‘Area’, Perform a groupby operation on ‘Area’. Which of these Areas had the highest sum in 2017?
2. What is the total number and percentage of missing data in 2014 to 3 decimal places?
3. Answer the following questions based on the African food production dataset provided by the FAO website already provided

What is the total sum of Wine produced in 2015 and 2018 respectively?

4. Select columns ‘Y2017’ and ‘Area’, Perform a groupby operation on ‘Area’. Which of these Areas had the 7th lowest sum in 2017?
5. Perform a groupby operation on ‘Element’. What year has the highest sum of Stock Variation?
6. Which year had the least correlation with ‘Element Code’?
7. Perform a groupby operation on ‘Element’. What is the total number of the sum of Processing in 2017?
8. What is the total Protein supply quantity in Madagascar in 2015?
9. What is the mean and standard deviation across the whole dataset for the year 2017 to 2 decimal places?

```
#importing libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
#importing data from pc to colab
from google.colab import files
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
# Reading our dataset
food_data = pd.read_excel("foodbalancesheets_e_africa_noflag.xlsx")
food_data.head()
```

	Area Code	Area	Item Code	Item	Element Code	Element	Unit	Y2014	
0	4	Algeria	2501	Population	511	Total Population - Both sexes	1000 persons	38924.00	397
1	4	Algeria	2501	Population	5301	Domestic supply quantity	1000 tonnes	0.00	
2	4	Algeria	2901	Grand Total	664	Food supply (kcal/capita/day)	kcal/capita/day	3377.00	33

```
#viewing samples of the dataset.
food_data.sample(20)
```

	Area Code	Area	Item Code	Item	Element Code	Element	Unit	Y2
18344	209	Eswatini	2601	Tomatoes and products	645	Food supply quantity (kg/capita/yr)	kg	5
33893	133	Mali	2745	Honey	5511	Production	1000 tonnes	12
34095	133	Mali	2563	Olives (including preserved)	674	Protein supply quantity (g/capita/day)	g/capita/day	0
35893	136	Mauritania	2949	Eggs	5072	Stock Variation	1000 tonnes	0
50785	202	South Africa	2645	Spices, Other	674	Protein supply quantity (g/capita/day)	g/capita/day	0
6094	233	Burkina Faso	2582	Maize Germ Oil	5911	Export Quantity	1000 tonnes	0
48279	195	Senegal	2737	Fats, Animals, Raw	5301	Domestic supply quantity	1000 tonnes	4
45914	193	Sao Tome and Principe	2805	Rice and products	5171	Tourist consumption	1000 tonnes	0
17062	59	Egypt	2613	Grapefruit and products	5072	Stock Variation	1000 tonnes	0
39230	144	Mozambique	2549	Pulses, Other and products	5521	Feed	1000 tonnes	22
1132	4	Algeria	2744	Eggs	5072	Stock Variation	1000 tonnes	0
48296	195	Senegal	2781	Fish, Body Oil	5142	Food	1000 tonnes	0
34310	133	Mali	2611	Oranges, Mandarines	5072	Stock Variation	1000 tonnes	1
7173	35	Cabo Verde	2556	Groundnuts (Shelled Eq)	5142	Food	1000 tonnes	0
17790	209	Eswatini	2514	Maize and products	5072	Stock Variation	1000 tonnes	0
58415	251	Zambia	2745	Honey	5301	Domestic supply quantity	1000 tonnes	0
11414	39	Chad	2573	Sunflowerseed Oil	5154	Other uses (non-food)	1000 tonnes	0
						Protein		

```
food_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60943 entries, 0 to 60942
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Area Code     60943 non-null  int64  
 1   Area          60943 non-null  object 
 2   Item Code     60943 non-null  int64  
 3   Item          60943 non-null  object 
 4   Element Code  60943 non-null  int64  
 5   Element       60943 non-null  object 
 6   Unit          60943 non-null  object 
 7   Y2014         59354 non-null  float64
 8   Y2015         59395 non-null  float64
 9   Y2016         59408 non-null  float64
 10  Y2017         59437 non-null  float64
 11  Y2018         59507 non-null  float64
dtypes: float64(5), int64(3), object(4)
memory usage: 5.6+ MB
```

```
food_data.describe()
```

	Area Code	Item Code	Element Code	Y2014	Y2015	
count	60943.000000	60943.000000	60943.000000	59354.000000	59395.000000	59408
mean	134.265576	2687.176706	3814.856456	134.196282	135.235966	136
std	72.605709	146.055739	2212.007033	1567.663696	1603.403984	1640
min	4.000000	2501.000000	511.000000	-1796.000000	-3161.000000	-3225
25%	74.000000	2562.000000	684.000000	0.000000	0.000000	0
50%	136.000000	2630.000000	5142.000000	0.090000	0.080000	0
75%	195.000000	2775.000000	5511.000000	8.340000	8.460000	8

```
#checking for duplicates
print(food_data.duplicated().sum())
```

```
0
```

```
#checking for missing values
print(food_data.isnull().sum())
```

```
Area Code          0
Area               0
Item Code          0
Item               0
Element Code       0
Element            0
```

```
Unit              0
Y2014          1589
Y2015          1548
Y2016          1535
Y2017          1506
Y2018          1436
dtype: int64
```

Question 1: Select columns ‘Y2017’ and ‘Area’, Perform a groupby operation on ‘Area’. Which of these Areas had the highest sum in 2017?

```
# Area with the highest sum in 2017 grouping by Y2017 and Area.
highest_sum_area = food_data.groupby('Area')['Y2017'].sum().sort_values(ascending=Fal
print(f"Area with highest sum in 2017: {highest_sum_area}")
```

```
Area with highest sum in 2017: Nigeria
```

Question 2: What is the total number and percentage of missing data in 2014 to 3 decimal places?

```
#the total number and percentage of missing data in 2014 to 3 decimal places

total_missing_2014 = food_data['Y2014'].isnull().sum()
total_data_points = len(food_data['Y2014'])
missing_percentage_2014 = (total_missing_2014 / total_data_points) * 100

print(f"Total missing data in 2014: {total_missing_2014}")
print(f"Percentage of missing data in 2014: {missing_percentage_2014:.3%}")
```

```
Total missing data in 2014: 1589
Percentage of missing data in 2014: 260.735%
```

Question 3: Answer the following questions based on the African food production dataset provided by the FAO website already provided

What is the total sum of Wine produced in 2015 and 2018 respectively?

```
#the total sum of Wine produced in 2015 and 2018 respectively

wine_production_2015 = food_data[food_data['Item'] == 'Wine'].loc[:, 'Y2015'].sum()
wine_production_2018 = food_data[food_data['Item'] == 'Wine'].loc[:, 'Y2018'].sum()

print(f"Total Wine production in 2015: {wine_production_2015}")
print(f"Total Wine production in 2018: {wine_production_2018}")
```

```
Total Wine production in 2015: 4251.8099999999995
Total Wine production in 2018: 4039.3199999999997
```

Question 4: Select columns ‘Y2017’ and ‘Area’, Perform a groupby operation on ‘Area’. Which of these Areas had the 7th lowest sum in 2017?

```
# Area that had the 7th lowest sum in 2017 grouping by Y2017 and Area.
seventh_lowest_area = food_data.groupby('Area')['Y2017'].sum().sort_values().index[6]
print(f"Area with 7th lowest sum in 2017: {seventh_lowest_area}")
```

```
Area with 7th lowest sum in 2017: Guinea-Bissau
```

Question 5: Perform a groupby operation on ‘Element’. What year has the highest sum of Stock Variation?

```
food_data.columns
```

```
Index(['Area Code', 'Area', 'Item Code', 'Item', 'Element Code', 'Element',
       'Unit', 'Y2014', 'Y2015', 'Y2016', 'Y2017', 'Y2018'],
      dtype='object')
```

```
element_groups = food_data.groupby('Element')['Y2017'].sum()

year_highest_sum = element_groups.idxmax()

print("Year with the highest sum of Stock Variation for each element:", year_highest_s
Year with the highest sum of Stock Variation for each element: Domestic supply qua
```

Question 6: Which year had the least correlation with ‘Element Code’?

```
# correlations between 'Element Code' and all year columns
correlations = food_data[['Element Code'] + [col for col in food_data.columns if col.s

least_correlated_year = correlations['Element Code'].abs().sort_values().index[1]

print(f"Year with least correlation with 'Element Code': {least_correlated_year}")
```

```
Year with least correlation with 'Element Code': Y2015
```

Question 7: Perform a groupby operation on ‘Element’. What is the total number of the sum of Processing in Y2017?

```
# a groupby operation on ‘Element’ to find the total number of the sum of Processing i

element_groups = food_data.groupby('Element')['Y2017'].sum()
total_processing = element_groups['Processing'].sum()
print(f"Total sum of Processing in 2017: {total_processing}")
```

```
Total sum of Processing in 2017: 292836.0
```

Question 8: What is the total Protein supply quantity in Madagascar in 2015?

```
# total Protein supply quantity in Madagascar in Y2015

protein_supply_madagascar_2015 = food_data[
    (food_data['Item'] == 'Protein supply quantity (g/capita/day)') & (food_data['Area'
]['Y2015'].sum()
```