p4-chlo195169

March 16, 2024

PROJET 4 DATA ANALYST

Réalisez une étude de santé publique avec R ou Python

1 OBJECTIF DE CE NOTEBOOK

Bienvenue dans l'outil plébiscité par les analystes de données Jupyter.

Il s'agit d'un outil permettant de mixer et d'alterner codes, textes et graphique.

Cet outil est formidable pour plusieurs raisons:

- il permet de tester des lignes de codes au fur et à mesure de votre rédaction, de constater immédiatement le résultat d'un instruction, de la corriger si nécessaire.
- De rédiger du texte pour expliquer l'approche suivie ou les résultats d'une analyse et de le mettre en forme grâce à du code html ou plus simple avec **Markdown**
- d'agrémenter de graphiques

Pour vous aider dans vos premiers pas à l'usage de Jupyter et de Python, nous avons rédigé ce notebook en vous indiquant les instructions à suivre.

Il vous suffit pour cela de saisir le code Python répondant à l'instruction donnée.

Vous verrez de temps à autre le code Python répondant à une instruction donnée mais cela est fait pour vous aider à comprendre la nature du travail qui vous est demandée.

Et garder à l'esprit, qu'il n'y a pas de solution unique pour résoudre un problème et qu'il y a autant de résolutions de problèmes que de développeurs ;)…

Note jeremy Est ce qu'il faut faire le calcul de la sous nutrition sur les pays qu'on a ? Est ce qu'il faut faire des graphiques ? Rajouter le soja La liste des céréales est difficile a trouver …

Etape 1 - Importation des librairies et chargement des fichiers

1.1 - Importation des librairies

[1]:
```
#Importation de la librairie Pandas
import pandas as pd
```

1.2 - Chargement des fichiers Excel

[2]:
```
#Importation du fichier population.csv
population = pd.read_csv('population.csv')
```

```
#Importation du fichier dispo_alimentaire.csv
dispo_alimentaire = pd.read_csv('dispo_alimentaire.csv')
```

[3]:
```
#Importation du fichier aide_alimentaire.csv
aide_alimentaire = pd.read_csv('aide_alimentaire.csv')
```

[4]:
```
#Importation du fichier sous_nutrition.csv
sous_nutrition = pd.read_csv('sous_nutrition.csv')
```

Etape 2 - Analyse exploratoire des fichiers

2.1 - Analyse exploratoire du fichier population

[5]:
```
#Afficher les dimensions du dataset
print("Le tableau comporte {} observation(s) ou article(s)".format(population.
  ↪shape[0]))
print("Le tableau comporte {} colonne(s)".format(population.shape[1]))
```

```
Le tableau comporte 1416 observation(s) ou article(s)
Le tableau comporte 3 colonne(s)
```

[6]:
```
#Consulter le nombre de colonnes
#La nature des données dans chacune des colonnes
#Le nombre de valeurs présentes dans chacune des colonnes
population.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1416 entries, 0 to 1415
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Zone    1416 non-null   object
 1   Année   1416 non-null   int64
 2   Valeur  1416 non-null   float64
dtypes: float64(1), int64(1), object(1)
memory usage: 33.3+ KB
```

[7]:
```
#Affichage les 5 premières lignes de la table
population.head()
```

[7]:

	Zone	Année	Valeur
0	Afghanistan	2013	32269.589
1	Afghanistan	2014	33370.794
2	Afghanistan	2015	34413.603
3	Afghanistan	2016	35383.032
4	Afghanistan	2017	36296.113

[8]:
```
#Nous allons harmoniser les unités. Pour cela, nous avons décidé de multiplier la population par 1000
#Multiplication de la colonne valeur par 1000
population['Valeur'] = population['Valeur'] * 1000
```

[9]:
```
population.head()
```

[9]:

	Zone	Année	Valeur
0	Afghanistan	2013	32269589.0
1	Afghanistan	2014	33370794.0
2	Afghanistan	2015	34413603.0
3	Afghanistan	2016	35383032.0
4	Afghanistan	2017	36296113.0

[10]:
```
#changement du nom de la colonne Valeur par Population
population.rename(columns={"Valeur": "Population"}, inplace=True)
```

[11]:
```
#Affichage les 5 premières lignes de la table pour voir les modifications
population.head()
```

[11]:

	Zone	Année	Population
0	Afghanistan	2013	32269589.0
1	Afghanistan	2014	33370794.0
2	Afghanistan	2015	34413603.0
3	Afghanistan	2016	35383032.0
4	Afghanistan	2017	36296113.0

2.2 - Analyse exploratoire du fichier disponibilité alimentaire

[12]:
```
#Afficher les dimensions du dataset
print("Le tableau comporte {} observation(s) ou article(s)".format(dispo_alimentaire.shape[0]))
print("Le tableau comporte {} colonne(s)".format(dispo_alimentaire.shape[1]))
```

```
Le tableau comporte 15605 observation(s) ou article(s)
Le tableau comporte 18 colonne(s)
```

[13]:
```
#Consulter le nombre de colonnes
dispo_alimentaire.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15605 entries, 0 to 15604
Data columns (total 18 columns):
 #   Column                                                    Non-Null
Count  Dtype
---  ------
--------------  -----
 0   Zone                                                      15605 non-
```

```
null  object
 1   Produit                                                 15605 non-
null  object
 2   Origine                                                 15605 non-
null  object
 3   Aliments pour animaux                                   2720 non-
null   float64
 4   Autres Utilisations                                     5496 non-
null   float64
 5   Disponibilité alimentaire (Kcal/personne/jour)          14241 non-
null  float64
 6   Disponibilité alimentaire en quantité (kg/personne/an)  14015 non-
null  float64
 7   Disponibilité de matière grasse en quantité (g/personne/jour)  11794 non-
null  float64
 8   Disponibilité de protéines en quantité (g/personne/jour)  11561 non-
null  float64
 9   Disponibilité intérieure                                15382 non-
null  float64
 10  Exportations - Quantité                                 12226 non-
null  float64
 11  Importations - Quantité                                 14852 non-
null  float64
 12  Nourriture                                              14015 non-
null  float64
 13  Pertes                                                  4278 non-
null   float64
 14  Production                                              9180 non-
null   float64
 15  Semences                                                2091 non-
null   float64
 16  Traitement                                              2292 non-
null   float64
 17  Variation de stock                                      6776 non-
null   float64
dtypes: float64(15), object(3)
memory usage: 2.1+ MB
```

[14]:
```
#Affichage les 5 premières lignes de la table
dispo_alimentaire.head()
```

[14]:

	Zone	Produit	Origine	Aliments pour animaux	\
0	Afghanistan	Abats Comestible	animale	NaN	
1	Afghanistan	Agrumes, Autres	vegetale	NaN	
2	Afghanistan	Aliments pour enfants	vegetale	NaN	
3	Afghanistan	Ananas	vegetale	NaN	
4	Afghanistan	Bananes	vegetale	NaN	

```
   Autres Utilisations  Disponibilité alimentaire (Kcal/personne/jour)  \
0                  NaN                                             5.0
1                  NaN                                             1.0
2                  NaN                                             1.0
3                  NaN                                             0.0
4                  NaN                                             4.0

   Disponibilité alimentaire en quantité (kg/personne/an)  \
0                                               1.72
1                                               1.29
2                                               0.06
3                                               0.00
4                                               2.70

   Disponibilité de matière grasse en quantité (g/personne/jour)  \
0                                               0.20
1                                               0.01
2                                               0.01
3                                                NaN
4                                               0.02

   Disponibilité de protéines en quantité (g/personne/jour)  \
0                                               0.77
1                                               0.02
2                                               0.03
3                                                NaN
4                                               0.05

   Disponibilité intérieure  Exportations - Quantité  Importations - Quantité  \
0                      53.0                       NaN                       NaN
1                      41.0                       2.0                      40.0
2                       2.0                       NaN                       2.0
3                       0.0                       NaN                       0.0
4                      82.0                       NaN                      82.0

   Nourriture  Pertes  Production  Semences  Traitement  Variation de stock
0        53.0     NaN        53.0       NaN         NaN                 NaN
1        39.0     2.0         3.0       NaN         NaN                 NaN
2         2.0     NaN         NaN       NaN         NaN                 NaN
3         0.0     NaN         NaN       NaN         NaN                 NaN
4        82.0     NaN         NaN       NaN         NaN                 NaN
```

[15]:
```
#remplacement des NaN dans le dataset par des 0
dispo_alimentaire.fillna(0, inplace=True)
```

[16]:
```
dispo_alimentaire.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15605 entries, 0 to 15604
Data columns (total 18 columns):
 #   Column                                                Non-Null
Count  Dtype
---  ------
--------------  -----
 0   Zone                                                  15605 non-
null  object
 1   Produit                                               15605 non-
null  object
 2   Origine                                               15605 non-
null  object
 3   Aliments pour animaux                                 15605 non-
null  float64
 4   Autres Utilisations                                   15605 non-
null  float64
 5   Disponibilité alimentaire (Kcal/personne/jour)        15605 non-
null  float64
 6   Disponibilité alimentaire en quantité (kg/personne/an)  15605 non-
null  float64
 7   Disponibilité de matière grasse en quantité (g/personne/jour)  15605 non-
null  float64
 8   Disponibilité de protéines en quantité (g/personne/jour)  15605 non-
null  float64
 9   Disponibilité intérieure                              15605 non-
null  float64
 10  Exportations - Quantité                               15605 non-
null  float64
 11  Importations - Quantité                               15605 non-
null  float64
 12  Nourriture                                            15605 non-
null  float64
 13  Pertes                                                15605 non-
null  float64
 14  Production                                            15605 non-
null  float64
 15  Semences                                              15605 non-
null  float64
 16  Traitement                                            15605 non-
null  float64
 17  Variation de stock                                    15605 non-
null  float64
dtypes: float64(15), object(3)
memory usage: 2.1+ MB
```

[17]:
```
#multiplication de toutes les lignes contenant des milliers de tonnes en Kg
colonnes_tonnes_tokg = ['Aliments pour animaux', 'Disponibilité intérieure',␣
 ↪'Exportations - Quantité', 'Importations - Quantité', 'Nourriture',␣
 ↪'Pertes', 'Production', 'Semences', 'Traitement', 'Variation de stock',␣
 ↪'Autres Utilisations']
for elt in colonnes_tonnes_tokg:
    dispo_alimentaire[elt] *= 1000000
```

[18]:
```
print("Avant conversion en kilogrammes:")
print(dispo_alimentaire)
```

```
Avant conversion en kilogrammes:
               Zone                 Produit   Origine  Aliments pour animaux  \
0       Afghanistan       Abats Comestible   animale                    0.0
1       Afghanistan        Agrumes, Autres  vegetale                    0.0
2       Afghanistan  Aliments pour enfants  vegetale                    0.0
3       Afghanistan                Ananas  vegetale                    0.0
4       Afghanistan               Bananes  vegetale                    0.0
…             …                     …       …                      …
15600  Îles Salomon       Viande de Suides   animale                    0.0
15601  Îles Salomon    Viande de Volailles   animale                    0.0
15602  Îles Salomon          Viande, Autre   animale                    0.0
15603  Îles Salomon                    Vin  vegetale                    0.0
15604  Îles Salomon         Épices, Autres  vegetale                    0.0

       Autres Utilisations  Disponibilité alimentaire (Kcal/personne/jour)  \
0                      0.0                                             5.0
1                      0.0                                             1.0
2                      0.0                                             1.0
3                      0.0                                             0.0
4                      0.0                                             4.0
…                      …                                               …
15600                  0.0                                            45.0
15601                  0.0                                            11.0
15602                  0.0                                             0.0
15603                  0.0                                             0.0
15604                  0.0                                             4.0

       Disponibilité alimentaire en quantité (kg/personne/an)  \
0                                                        1.72
1                                                        1.29
2                                                        0.06
3                                                        0.00
4                                                        2.70
…                                                           …
15600                                                    4.70
15601                                                    3.34
```

```
15602                                                0.06
15603                                                0.07
15604                                                0.48

       Disponibilité de matière grasse en quantité (g/personne/jour)  \
0                                                    0.20
1                                                    0.01
2                                                    0.01
3                                                    0.00
4                                                    0.02
…                                                     …
15600                                                4.28
15601                                                0.69
15602                                                0.00
15603                                                0.00
15604                                                0.21

       Disponibilité de protéines en quantité (g/personne/jour)  \
0                                                    0.77
1                                                    0.02
2                                                    0.03
3                                                    0.00
4                                                    0.05
…                                                     …
15600                                                1.41
15601                                                1.14
15602                                                0.04
15603                                                0.00
15604                                                0.15

       Disponibilité intérieure  Exportations - Quantité  \
0                    53000000.0                       0.0
1                    41000000.0                 2000000.0
2                     2000000.0                       0.0
3                           0.0                       0.0
4                    82000000.0                       0.0
…                           …                         …
15600                 3000000.0                       0.0
15601                 2000000.0                       0.0
15602                       0.0                       0.0
15603                       0.0                       0.0
15604                       0.0                       0.0

       Importations - Quantité  Nourriture     Pertes  Production  Semences  \
0                          0.0  53000000.0        0.0  53000000.0       0.0
1                   40000000.0  39000000.0  2000000.0   3000000.0       0.0
2                    2000000.0   2000000.0        0.0         0.0       0.0
3                          0.0         0.0        0.0         0.0       0.0
```

```
4                   82000000.0  82000000.0        0.0         0.0       0.0
…                          …           …          …           …         …
15600                      0.0   3000000.0        0.0   2000000.0       0.0
15601                2000000.0   2000000.0        0.0         0.0       0.0
15602                      0.0         0.0        0.0         0.0       0.0
15603                      0.0         0.0        0.0         0.0       0.0
15604                      0.0         0.0        0.0         0.0       0.0

       Traitement  Variation de stock
0             0.0                 0.0
1             0.0                 0.0
2             0.0                 0.0
3             0.0                 0.0
4             0.0                 0.0
…             …                   …
15600         0.0                 0.0
15601         0.0                 0.0
15602         0.0                 0.0
15603         0.0                 0.0
15604         0.0                 0.0

[15605 rows x 18 columns]
```

[19]:
```
#Affichage les 5 premières lignes de la table
dispo_alimentaire.head()
```

[19]:
```
          Zone                Produit   Origine  Aliments pour animaux  \
0  Afghanistan       Abats Comestible   animale                    0.0
1  Afghanistan        Agrumes, Autres  vegetale                    0.0
2  Afghanistan  Aliments pour enfants  vegetale                    0.0
3  Afghanistan                Ananas  vegetale                    0.0
4  Afghanistan               Bananes  vegetale                    0.0

   Autres Utilisations  Disponibilité alimentaire (Kcal/personne/jour)  \
0                  0.0                                             5.0
1                  0.0                                             1.0
2                  0.0                                             1.0
3                  0.0                                             0.0
4                  0.0                                             4.0

   Disponibilité alimentaire en quantité (kg/personne/an)  \
0                                                1.72
1                                                1.29
2                                                0.06
3                                                0.00
4                                                2.70
```

```
   Disponibilité de matière grasse en quantité (g/personne/jour)  \
0                                                       0.20
1                                                       0.01
2                                                       0.01
3                                                       0.00
4                                                       0.02

   Disponibilité de protéines en quantité (g/personne/jour)  \
0                                                       0.77
1                                                       0.02
2                                                       0.03
3                                                       0.00
4                                                       0.05

   Disponibilité intérieure  Exportations - Quantité  Importations - Quantité  \
0                53000000.0                     0.0                      0.0
1                41000000.0                2000000.0                40000000.0
2                 2000000.0                     0.0                 2000000.0
3                       0.0                     0.0                      0.0
4                82000000.0                     0.0                82000000.0

   Nourriture     Pertes  Production  Semences  Traitement  Variation de stock
0  53000000.0        0.0  53000000.0       0.0         0.0                 0.0
1  39000000.0  2000000.0   3000000.0       0.0         0.0                 0.0
2   2000000.0        0.0         0.0       0.0         0.0                 0.0
3         0.0        0.0         0.0       0.0         0.0                 0.0
4  82000000.0        0.0         0.0       0.0         0.0                 0.0
```

[20]:
```
print("\nAprès conversion en kilogrammes:")
print(dispo_alimentaire['Aliments pour animaux'])
```

```
Après conversion en kilogrammes:
0        0.0
1        0.0
2        0.0
3        0.0
4        0.0
        ...
15600    0.0
15601    0.0
15602    0.0
15603    0.0
15604    0.0
Name: Aliments pour animaux, Length: 15605, dtype: float64
```

[21]:
```
dispo_alimentaire[dispo_alimentaire['Aliments pour␣
 ↪animaux']>0][['Zone','Aliments pour animaux']]
```

[21]:

	Zone	Aliments pour animaux
29	Afghanistan	1.230000e+08
31	Afghanistan	4.000000e+06
32	Afghanistan	2.000000e+08
40	Afghanistan	3.600000e+08
50	Afghanistan	8.100000e+07
…	…	…
15421	États-Unis d'Amérique	2.351000e+09
15428	États-Unis d'Amérique	1.600000e+07
15432	États-Unis d'Amérique	3.300000e+07
15480	Éthiopie	1.800000e+07
15483	Éthiopie	6.670000e+08

[1790 rows x 2 columns]

2.3 - Analyse exploratoire du fichier aide alimentaire

[22]:
```
#Afficher les dimensions du dataset
print("Le tableau comporte {} observation(s) ou article(s)".
 ↪format(aide_alimentaire.shape[0]))
print("Le tableau comporte {} colonne(s)".format(aide_alimentaire.shape[1]))
```

```
Le tableau comporte 1475 observation(s) ou article(s)
Le tableau comporte 4 colonne(s)
```

[23]:
```
#Consulter le nombre de colonnes
aide_alimentaire.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1475 entries, 0 to 1474
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Pays bénéficiaire  1475 non-null   object
 1   Année              1475 non-null   int64
 2   Produit            1475 non-null   object
 3   Valeur             1475 non-null   int64
dtypes: int64(2), object(2)
memory usage: 46.2+ KB
```

[24]:
```
#Aaide_alimentaire.head()ffichage les 5 premières lignes de la table
aide_alimentaire.head()
```

[24]:
	Pays bénéficiaire	Année	Produit	Valeur
0	Afghanistan	2013	Autres non-céréales	682
1	Afghanistan	2014	Autres non-céréales	335
2	Afghanistan	2013	Blé et Farin	39224
3	Afghanistan	2014	Blé et Farin	15160
4	Afghanistan	2013	Céréales	40504

[25]:
```
#changement du nom de la colonne Pays bénéficiaire par Zone
aide_alimentaire.rename(columns={'Pays bénéficiaire': 'Zone'}, inplace=True)
```

[26]:
```
#Multiplication de la colonne Aide_alimentaire qui contient des tonnes par 1000␣
 ↪pour avoir des kg
aide_alimentaire['Valeur'] = aide_alimentaire['Valeur'] * 1000
```

[27]:
```
#Affichage les 5 premières lignes de la table
aide_alimentaire.head()
```

[27]:
	Zone	Année	Produit	Valeur
0	Afghanistan	2013	Autres non-céréales	682000
1	Afghanistan	2014	Autres non-céréales	335000
2	Afghanistan	2013	Blé et Farin	39224000
3	Afghanistan	2014	Blé et Farin	15160000
4	Afghanistan	2013	Céréales	40504000

2.3 - Analyse exploratoire du fichier sous nutrition

[28]:
```
#Afficher les dimensions du dataset
print("Le tableau comporte {} observation(s) ou article(s)".
 ↪format(sous_nutrition.shape[0]))
print("Le tableau comporte {} colonne(s)".format(sous_nutrition.shape[1]))
```

```
Le tableau comporte 1218 observation(s) ou article(s)
Le tableau comporte 3 colonne(s)
```

[29]:
```
#Consulter le nombre de colonnes
sous_nutrition.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1218 entries, 0 to 1217
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Zone    1218 non-null   object
 1   Année   1218 non-null   object
 2   Valeur  624 non-null    object
dtypes: object(3)
memory usage: 28.7+ KB
```

[30]:
```
#Afficher les 5 premières lignes de la table
sous_nutrition.head()
```

[30]:

	Zone	Année	Valeur
0	Afghanistan	2012-2014	8.6
1	Afghanistan	2013-2015	8.8
2	Afghanistan	2014-2016	8.9
3	Afghanistan	2015-2017	9.7
4	Afghanistan	2016-2018	10.5

[31]:
```
sous_nutrition[sous_nutrition['Valeur']=='<0.1']
```

[31]:

	Zone	Année	Valeur
60	Arménie	2012-2014	<0.1
61	Arménie	2013-2015	<0.1
62	Arménie	2014-2016	<0.1
63	Arménie	2015-2017	<0.1
64	Arménie	2016-2018	<0.1
…	…	…	…
1183	Vanuatu	2013-2015	<0.1
1184	Vanuatu	2014-2016	<0.1
1185	Vanuatu	2015-2017	<0.1
1186	Vanuatu	2016-2018	<0.1
1187	Vanuatu	2017-2019	<0.1

[120 rows x 3 columns]

[32]:
```
#Conversion de la colonne sous nutrition en numérique
sous_nutrition['Valeur'] = pd.to_numeric(sous_nutrition['Valeur'])
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
File ~\AppData\Local\anaconda3\Lib\site-packages\pandas\_libs\lib.pyx:2280, in␣
 ↪pandas._libs.lib.maybe_convert_numeric()

ValueError: Unable to parse string "<0.1"

During handling of the above exception, another exception occurred:

ValueError                                Traceback (most recent call last)
Cell In[32], line 2
      1 #Conversion de la colonne sous nutrition en numérique
----> 2 sous_nutrition['Valeur'] = pd.to_numeric(sous_nutrition['Valeur'])

File ~\AppData\Local\anaconda3\Lib\site-packages\pandas\core\tools\numeric.py:
 ↪217, in to_numeric(arg, errors, downcast, dtype_backend)
    215 coerce_numeric = errors not in ("ignore", "raise")
```

```
    216 try:
--> 217     values, new_mask = lib.maybe_convert_numeric(  # type:␣
 ↪ignore[call-overload]  # noqa
    218         values,
    219         set(),
    220         coerce_numeric=coerce_numeric,
    221         convert_to_masked_nullable=dtype_backend is not lib.no_default
    222         or isinstance(values_dtype, StringDtype),
    223     )
    224 except (ValueError, TypeError):
    225     if errors == "raise":

File ~\AppData\Local\anaconda3\Lib\site-packages\pandas\_libs\lib.pyx:2322, in␣
 ↪pandas._libs.lib.maybe_convert_numeric()

ValueError: Unable to parse string "<0.1" at position 60
```

[33]:
```
sous_nutrition['Valeur'] = pd.to_numeric(sous_nutrition['Valeur'],␣
 ↪errors='coerce')
```

[34]:
```
sous_nutrition.isna().sum()
```

[34]:
```
Zone        0
Année       0
Valeur    714
dtype: int64
```

[35]:
```
#Conversion de la colonne (avec l'argument errors=coerce qui permet de␣
 ↪convertir automatiquement les lignes qui ne sont pas des nombres en NaN)
#Puis remplacement des NaN en 0
sous_nutrition.fillna(0,inplace=True)
```

[36]:
```
#changement du nom de la colonne Valeur par sous_nutrition
sous_nutrition.rename(columns={'Valeur': 'sous_nutrition'}, inplace=True)
```

[37]:
```
sous_nutrition.head()
```

[37]:

	Zone	Année	sous_nutrition
0	Afghanistan	2012-2014	8.6
1	Afghanistan	2013-2015	8.8
2	Afghanistan	2014-2016	8.9
3	Afghanistan	2015-2017	9.7
4	Afghanistan	2016-2018	10.5

[38]:
```
sous_nutrition.dtypes
```

[38]:
```
Zone               object
Année              object
sous_nutrition    float64
dtype: object
```

[39]:
```
#Multiplication de la colonne sous_nutrition par 1000000
sous_nutrition['sous_nutrition'] = sous_nutrition['sous_nutrition'] * 1000000
```

[40]:
```
#Afficher les 5 premières lignes de la table
sous_nutrition.head()
```

[40]:

	Zone	Année	sous_nutrition
0	Afghanistan	2012-2014	8600000.0
1	Afghanistan	2013-2015	8800000.0
2	Afghanistan	2014-2016	8900000.0
3	Afghanistan	2015-2017	9700000.0
4	Afghanistan	2016-2018	10500000.0

[41]:
```
# Il faut tout d'abord faire une jointure entre la table population et la table␣
↪sous nutrition, en ciblant l'année 2017
df_pop_sous_nut = pd.merge(population[population["Année"]==␣
↪2017],sous_nutrition[sous_nutrition["Année"]== "2016-2018"], on ="Zone")
```

3.1 - Proportion de personnes en sous nutrition

[42]:
```
df_pop_sous_nut.rename(columns= {"Année_x":"Année_pop","Valeur_x":
↪"Nb_pop","Année_y":"Année_sous_nut","Valeur_y":"sous_nutrition"},inplace =␣
↪True)
```

[43]:
```
#Affichage du dataset
df_pop_sous_nut.head()
```

[43]:

	Zone	Année_pop	Population	Année_sous_nut	sous_nutrition
0	Afghanistan	2017	36296113.0	2016-2018	10500000.0
1	Afrique du Sud	2017	57009756.0	2016-2018	3100000.0
2	Albanie	2017	2884169.0	2016-2018	100000.0
3	Algérie	2017	41389189.0	2016-2018	1300000.0
4	Allemagne	2017	82658409.0	2016-2018	0.0

[44]:
```
#Calcul et affichage du nombre de personnes en état de sous nutrition
nb_sous_nut=df_pop_sous_nut["sous_nutrition"].sum()
```

[45]:
```
nb_pop=df_pop_sous_nut["Population"].sum()
```

[46]:
```
pourcentage_sous_nut=round(nb_sous_nut*100/nb_pop,2)
```

[47]:
```
print("Proportion de personnes en état de sous nutrition :", "{:.2f}".
↪format(pourcentage_sous_nut), "%")
```

```
Proportion de personnes en état de sous nutrition : 7.10 %
```

3.2 - Nombre théorique de personne qui pourrait être nourries

[48]:
```
#Combien mange en moyenne un être humain ? Source => Les besoins alimentaires␣
↪d'un être humain sont évalués en moyenne à 2 100 kcal par jour (PAM).
↪Médecins sans frontières
```

[49]:
```
pop_2017= population.loc[population['Année'] == 2017,["Zone", "Population"]]
```

[50]:
```
#On commence par faire une jointure entre le data frame population et␣
↪Dispo_alimentaire afin d'ajouter dans ce dernier la population
colonne_commune = "Zone"
df_pop_dispo_ali = pd.merge(dispo_alimentaire, pop_2017, on=colonne_commune)
```

[51]:
```
#Affichage du nouveau dataframe
df_pop_dispo_ali.head()
```

[51]:

	Zone	Produit	Origine	Aliments pour animaux	Autres Utilisations	Disponibilité alimentaire (Kcal/personne/jour)	Disponibilité alimentaire en quantité (kg/personne/an)	Disponibilité de matière grasse en quantité (g/personne/jour)
0	Afghanistan	Abats Comestible	animale	0.0	0.0	5.0	1.72	0.20
1	Afghanistan	Agrumes, Autres	vegetale	0.0	0.0	1.0	1.29	0.01
2	Afghanistan	Aliments pour enfants	vegetale	0.0	0.0	1.0	0.06	0.01
3	Afghanistan	Ananas	vegetale	0.0	0.0	0.0	0.00	0.00
4	Afghanistan	Bananes	vegetale	0.0	0.0	4.0	2.70	0.02

```
   Disponibilité de protéines en quantité (g/personne/jour)  \
0                                               0.77
1                                               0.02
2                                               0.03
3                                               0.00
4                                               0.05

   Disponibilité intérieure  Exportations - Quantité  Importations - Quantité  \
0                53000000.0                      0.0                      0.0
1                41000000.0                2000000.0               40000000.0
2                 2000000.0                      0.0                2000000.0
3                       0.0                      0.0                      0.0
4                82000000.0                      0.0               82000000.0

   Nourriture     Pertes  Production  Semences  Traitement  \
0  53000000.0        0.0  53000000.0       0.0         0.0
1  39000000.0  2000000.0   3000000.0       0.0         0.0
2   2000000.0        0.0         0.0       0.0         0.0
3         0.0        0.0         0.0       0.0         0.0
4  82000000.0        0.0         0.0       0.0         0.0

   Variation de stock  Population
0                 0.0  36296113.0
1                 0.0  36296113.0
2                 0.0  36296113.0
3                 0.0  36296113.0
4                 0.0  36296113.0
```

[52]:
```
#Création de la colonne dispo_kcal avec calcul des kcal disponibles mondialement
df_pop_dispo_ali['dispo_kcal'] = df_pop_dispo_ali['Population'] *␣
↪df_pop_dispo_ali['Disponibilité alimentaire (Kcal/personne/jour)'] * 365
```

[53]:
```
df_pop_dispo_ali.head()
```

[53]:
```
          Zone                Produit   Origine  Aliments pour animaux  \
0  Afghanistan       Abats Comestible   animale                    0.0
1  Afghanistan        Agrumes, Autres  vegetale                    0.0
2  Afghanistan  Aliments pour enfants  vegetale                    0.0
3  Afghanistan                Ananas  vegetale                    0.0
4  Afghanistan               Bananes  vegetale                    0.0

   Autres Utilisations  Disponibilité alimentaire (Kcal/personne/jour)  \
0                  0.0                                             5.0
1                  0.0                                             1.0
2                  0.0                                             1.0
3                  0.0                                             0.0
```

```
4                  0.0                                                4.0

   Disponibilité alimentaire en quantité (kg/personne/an)  \
0                                                 1.72
1                                                 1.29
2                                                 0.06
3                                                 0.00
4                                                 2.70

   Disponibilité de matière grasse en quantité (g/personne/jour)  \
0                                                 0.20
1                                                 0.01
2                                                 0.01
3                                                 0.00
4                                                 0.02

   Disponibilité de protéines en quantité (g/personne/jour)  \
0                                                 0.77
1                                                 0.02
2                                                 0.03
3                                                 0.00
4                                                 0.05

   Disponibilité intérieure  Exportations - Quantité  Importations - Quantité  \
0               53000000.0                       0.0                       0.0
1               41000000.0                 2000000.0                40000000.0
2                2000000.0                       0.0                 2000000.0
3                      0.0                       0.0                       0.0
4               82000000.0                       0.0                82000000.0

   Nourriture     Pertes  Production  Semences  Traitement  \
0  53000000.0        0.0  53000000.0       0.0         0.0
1  39000000.0  2000000.0   3000000.0       0.0         0.0
2   2000000.0        0.0         0.0       0.0         0.0
3         0.0        0.0         0.0       0.0         0.0
4  82000000.0        0.0         0.0       0.0         0.0

   Variation de stock  Population    dispo_kcal
0                 0.0  36296113.0  6.624041e+10
1                 0.0  36296113.0  1.324808e+10
2                 0.0  36296113.0  1.324808e+10
3                 0.0  36296113.0  0.000000e+00
4                 0.0  36296113.0  5.299232e+10
```

[54]:
```python
#Calcul du nombre d'humains pouvant être nourris
total_h_kcal = round(df_pop_dispo_ali['dispo_kcal'].sum()/(2100*365))
print("Total d'être humain pouvant être nourris :", total_h_kcal)
```

```
print("Proportion :", "{:.2f}".format(total_h_kcal*100/population.
  ↪loc[population['Année'] == 2017,"Population"].sum()), "%")
```

```
Total d'être humain pouvant être nourris : 9961421251
Proportion : 131.97 %
```

3.3 - Nombre théorique de personne qui pourrait être nourrie avec les produits végétaux

[55]:
```
#Transfert des données avec les végétaux dans un nouveau dataframe
vegetaux_df = df_pop_dispo_ali.loc[df_pop_dispo_ali['Origine'] == "vegetale",:]
```

[56]:
```
vegetaux_df['Origine'].unique()
```

[56]:
```
array(['vegetale'], dtype=object)
```

[57]:
```
df_pop_dispo_ali.head()
```

[57]:

	Zone	Produit	Origine	Aliments pour animaux	Autres Utilisations	Disponibilité alimentaire (Kcal/personne/jour)	Disponibilité alimentaire en quantité (kg/personne/an)	Disponibilité de matière grasse en quantité (g/personne/jour)	Disponibilité de protéines en quantité (g/personne/jour)
0	Afghanistan	Abats Comestible	animale	0.0	0.0	5.0	1.72	0.20	0.77
1	Afghanistan	Agrumes, Autres	vegetale	0.0	0.0	1.0	1.29	0.01	0.02
2	Afghanistan	Aliments pour enfants	vegetale	0.0	0.0	1.0	0.06	0.01	
3	Afghanistan	Ananas	vegetale	0.0	0.0	0.0	0.00	0.00	
4	Afghanistan	Bananes	vegetale	0.0	0.0	4.0	2.70	0.02	

```
2                                                0.03
3                                                0.00
4                                                0.05

   Disponibilité intérieure  Exportations - Quantité  Importations - Quantité  \
0                53000000.0                      0.0                      0.0
1                41000000.0                2000000.0                40000000.0
2                 2000000.0                      0.0                 2000000.0
3                       0.0                      0.0                       0.0
4                82000000.0                      0.0                82000000.0

   Nourriture     Pertes  Production  Semences  Traitement  \
0  53000000.0        0.0  53000000.0       0.0         0.0
1  39000000.0  2000000.0   3000000.0       0.0         0.0
2   2000000.0        0.0         0.0       0.0         0.0
3         0.0        0.0         0.0       0.0         0.0
4  82000000.0        0.0         0.0       0.0         0.0

   Variation de stock  Population    dispo_kcal
0                 0.0  36296113.0  6.624041e+10
1                 0.0  36296113.0  1.324808e+10
2                 0.0  36296113.0  1.324808e+10
3                 0.0  36296113.0  0.000000e+00
4                 0.0  36296113.0  5.299232e+10
```

[58]:
```
#Calcul du nombre de kcal disponible pour les végétaux
kcal_disponibles_vegetaux = vegetaux_df['dispo_kcal'].sum()
```

[59]:
```
print("Nombre total de kilocalories disponibles pour les végétaux :",␣
↪kcal_disponibles_vegetaux)
```

```
Nombre total de kilocalories disponibles pour les végétaux : 6300178937197865.0
```

[60]:
```
consommation_moyenne_quotidienne = 2100
```

[61]:
```
#Calcul du nombre d'humains pouvant être nourris avec les végétaux
nombre_humains_nourris_vegetaux = round(kcal_disponibles_vegetaux/(2100*365))
```

[62]:
```
print("Nombre d'humains pouvant être nourris avec les végétaux :",␣
↪nombre_humains_nourris_vegetaux)
```

```
Nombre d'humains pouvant être nourris avec les végétaux : 8219411529
```

[63]:
```
print("Proportion d'humains que l'on peut nourrir avec les végétaux :", "{:.
↪2f}".format(nombre_humains_nourris_vegetaux*100/population.
↪loc[population['Année'] == 2017,"Population"].sum()), "%")
```

```
Proportion d'humains que l'on peut nourrir avec les végétaux : 108.89 %
```

3.4 - Utilisation de la disponibilité intérieure

[64]:
```
#Calcul de la disponibilité totale
disponibilite_totale = df_pop_dispo_ali['Disponibilité intérieure'].sum()
```

[65]:
```
print("Disponibilité totale :", disponibilite_totale)
```

```
Disponibilité totale : 9733927000000.0
```

[66]:
```
#création d'une boucle for pour afficher les différentes valeurs en fonction␣
 ↪des colonnes aliments pour animaux, pertes, nourritures,
colonnes_a_explorer = ['Aliments pour animaux', 'Pertes',␣
 ↪'Nourriture','Semences', 'Traitement', 'Autres Utilisations']
```

[67]:
```
for colonne in colonnes_a_explorer:
    valeurs_uniques = df_pop_dispo_ali[colonne].sum()*100/disponibilite_totale
    print("Proportion de", colonne, ":", "{:.2f}".format(valeurs_uniques), "%")
```

```
Proportion de Aliments pour animaux : 13.23 %
Proportion de Pertes : 4.65 %
Proportion de Nourriture : 49.37 %
Proportion de Semences : 1.58 %
Proportion de Traitement : 22.45 %
Proportion de Autres Utilisations : 8.82 %
```

[68]:
```
Tout n'est pas utilisé dans la nourriture humaine
```

```
  Cell In[68], line 1
    Tout n'est pas utilisé dans la nourriture humaine
          ^
SyntaxError: unterminated string literal (detected at line 1)
```

3.5 - Utilisation des céréales

[69]:
```
#Création d'une liste avec toutes les variables
liste_cereales = ["Blé", "Riz (Eq Blanchi)", "Orge", "Maïs", "Seigle",
                  "Avoine", "Millet", "Sorgho", "Céréales, Autres"]
```

[70]:
```
cereales = dispo_alimentaire.loc[dispo_alimentaire['Produit'].
 ↪isin(liste_cereales),:]
```

[71]:
```
print("Liste des produits :")
for produit in liste_cereales:
    print(produit)
```

```
Liste des produits :
Blé
```

```
Riz (Eq Blanchi)
Orge
Maïs
Seigle
Avoine
Millet
Sorgho
Céréales, Autres
```

[72]:
```
#Affichage de la proportion d'alimentation animale
aliments_pour_animaux_total = cereales['Aliments pour animaux'].sum()
```

[73]:
```
#Affichage de la proportion d'alimentation animale
print("Proportion d'alimentation animale :", aliments_pour_animaux_total/
 ↪cereales['Disponibilité intérieure'].sum())
```

```
Proportion d'alimentation animale : 0.36291456706047653
```

[74]:
```
aliments_pour_humain_total = cereales['Nourriture'].sum()
```

[75]:
```
#Affichage de la proportion d'alimentation humaine
print("Proportion d'alimentation humaine :", aliments_pour_humain_total/
 ↪cereales['Disponibilité intérieure'].sum())
```

```
Proportion d'alimentation humaine : 0.4275074480712289
```

3.6 - Pays avec la proportion de personnes sous-alimentée la plus forte en 2017

[76]:
```
#Création de la colonne proportion par pays
df_pop_sous_nut['proportion']= round(df_pop_sous_nut['sous_nutrition'] * 100 /␣
 ↪df_pop_sous_nut['Population'],2)
```

[77]:
```
df_pop_sous_nut.head(10)
```

[77]:

	Zone	Année_pop	Population	Année_sous_nut	sous_nutrition	proportion
0	Afghanistan	2017	36296113.0	2016-2018	10500000.0	28.93
1	Afrique du Sud	2017	57009756.0	2016-2018	3100000.0	
2	Albanie	2017	2884169.0	2016-2018	100000.0	
3	Algérie	2017	41389189.0	2016-2018	1300000.0	
4	Allemagne	2017	82658409.0	2016-2018	0.0	
5	Andorre	2017	77001.0	2016-2018	0.0	
6	Angola	2017	29816766.0	2016-2018	5800000.0	
7	Antigua-et-Barbuda	2017	95426.0	2016-2018	0.0	
8	Arabie saoudite	2017	33101179.0	2016-2018	1600000.0	
9	Argentine	2017	43937140.0	2016-2018	1500000.0	

```
1        5.44
2        3.47
3        3.14
4        0.00
5        0.00
6       19.45
7        0.00
8        4.83
9        3.41
```

[78]:
```
#affichage après trie des 10 pires pays
print("Les 10 pires pays en termes de proportion d'alimentation :")
df_pop_sous_nut[['Zone', "proportion"]].sort_values(by="proportion",␣
 ↪ascending=False).head(10)
```

```
Les 10 pires pays en termes de proportion d'alimentation :
```

[78]:

	Zone	proportion
78	Haïti	48.26
157	République populaire démocratique de Corée	47.19
108	Madagascar	41.06
103	Libéria	38.28
100	Lesotho	38.25
183	Tchad	37.96
161	Rwanda	35.06
121	Mozambique	32.81
186	Timor-Leste	32.17
0	Afghanistan	28.93

3.7 - Pays qui ont le plus bénéficié d'aide alimentaire depuis 2013

[79]:
```
#calcul du total de l'aide alimentaire par pays
total_aide_alimentaire_par_pays = aide_alimentaire.groupby('Zone')['Valeur'].
 ↪sum().reset_index()
```

[80]:
```
print("Total de l'aide alimentaire par pays :")
total_aide_alimentaire_par_pays
```

```
Total de l'aide alimentaire par pays :
```

[80]:

	Zone	Valeur
0	Afghanistan	185452000
1	Algérie	81114000
2	Angola	5014000
3	Bangladesh	348188000
4	Bhoutan	2666000
..	…	…
71	Zambie	3026000

```
72     Zimbabwe    62570000
73       Égypte     1122000
74     Équateur     1362000
75     Éthiopie  1381294000

[76 rows x 2 columns]
```

[81]:
```
#affichage après trie des 10 pays qui ont bénéficié le plus de l'aide␣
 ↪alimentaire
total_aide_alimentaire_par_pays.sort_values(by="Valeur", ascending=False).
 ↪head(10)
```

[81]:
```
                                Zone      Valeur
50         République arabe syrienne  1858943000
75                           Éthiopie  1381294000
70                              Yémen  1206484000
61                      Soudan du Sud   695248000
60                             Soudan   669784000
30                              Kenya   552836000
3                           Bangladesh   348188000
59                             Somalie   292678000
53  République démocratique du Congo   288502000
43                              Niger   276344000
```

3.8 - Evolution des 5 pays qui ont le plus bénéficiés de l'aide alimentaire entre 2013 et 2016

[82]:
```
#Création d'un dataframe avec la zone, l'année et l'aide alimentaire puis␣
 ↪groupby sur zone et année
Zone_année_aide = aide_alimentaire[['Zone','Valeur']].groupby("Zone").sum().
 ↪reset_index()
```

[83]:
```
Zone_année_aide
```

[83]:
```
           Zone      Valeur
0   Afghanistan   185452000
1       Algérie    81114000
2        Angola     5014000
3    Bangladesh   348188000
4       Bhoutan     2666000
..          …          …
71       Zambie     3026000
72     Zimbabwe    62570000
73       Égypte     1122000
74     Équateur     1362000
75     Éthiopie  1381294000

[76 rows x 2 columns]
```

[84]:
```
#Création d'une liste contenant les 5 pays qui ont le plus bénéficiées de␣
 ↪l'aide alimentaire
pays_les_plus_beneficiaires = Zone_année_aide.groupby('Zone')['Valeur'].sum().
 ↪nlargest(5)
```

[85]:
```
print("Les 5 pays qui ont le plus bénéficié de l'aide alimentaire :")
print(pays_les_plus_beneficiaires)
```

```
Les 5 pays qui ont le plus bénéficié de l'aide alimentaire :
Zone
République arabe syrienne    1858943000
Éthiopie                     1381294000
Yémen                        1206484000
Soudan du Sud                 695248000
Soudan                        669784000
Name: Valeur, dtype: int64
```

[86]:
```
#On filtre sur le dataframe avec notre liste
df_5_pays_aide=aide_alimentaire.loc[(aide_alimentaire['Zone'].isin(['République␣
 ↪arabe syrienne','Éthiopie','Yémen','Soudan du␣
 ↪Sud','Soudan']))&(aide_alimentaire['Année']>=2013)&(aide_alimentaire['Année']<=2016)]
```

[87]:
```
df_5_pays_aide=aide_alimentaire.loc[(aide_alimentaire['Zone'].isin(['République␣
 ↪arabe syrienne','Éthiopie','Yémen','Soudan du␣
 ↪Sud','Soudan']))&(aide_alimentaire['Année']>=2013)&(aide_alimentaire['Année']<=2016)]
```

[88]:
```
df_5_pays_aide_gp = df_5_pays_aide.groupby(['Zone','Année'])['Valeur'].sum().
 ↪reset_index()
```

[89]:
```
# Affichage des pays avec l'aide alimentaire par année
df_5_pays_aide_gp
```

[89]:

	Zone	Année	Valeur
0	République arabe syrienne	2013	563566000
1	République arabe syrienne	2014	651870000
2	République arabe syrienne	2015	524949000
3	République arabe syrienne	2016	118558000
4	Soudan	2013	330230000
5	Soudan	2014	321904000
6	Soudan	2015	17650000
7	Soudan du Sud	2013	196330000
8	Soudan du Sud	2014	450610000
9	Soudan du Sud	2015	48308000
10	Yémen	2013	264764000
11	Yémen	2014	103840000
12	Yémen	2015	372306000
13	Yémen	2016	465574000

```
14                  Éthiopie   2013  591404000
15                  Éthiopie   2014  586624000
16                  Éthiopie   2015  203266000
```

3.9 - Pays avec le moins de disponibilité par habitant

[90]:
```
#Calcul de la disponibilité en kcal par personne par jour par pays
dispo_ali_total = dispo_alimentaire[['Zone', 'Produit','Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)']].groupby('Zone').sum().reset_index()
```

[91]:
```
dispo_ali_total
```

[91]:
```
                     Zone                                            Produit  \
0              Afghanistan  Abats ComestibleAgrumes, AutresAliments pour e…
1           Afrique du Sud  Abats ComestibleAgrumes, AutresAlcool, non Com…
2                  Albanie  Abats ComestibleAgrumes, AutresAlcool, non Com…
3                  Algérie  Abats ComestibleAgrumes, AutresAlcool, non Com…
4                Allemagne  Abats ComestibleAgrumes, AutresAlcool, non Com…
..                     …                                                 …
169    Émirats arabes unis  Abats ComestibleAgrumes, AutresAlcool, non Com…
170               Équateur  Abats ComestibleAgrumes, AutresAlcool, non Com…
171  États-Unis d'Amérique  Abats ComestibleAgrumes, AutresAlcool, non Com…
172               Éthiopie  Abats ComestibleAgrumes, AutresAlcool, non Com…
173           Îles Salomon  Abats ComestibleAgrumes, AutresAlcool, non Com…

     Disponibilité alimentaire (Kcal/personne/jour)
0                                            2087.0
1                                            3020.0
2                                            3188.0
3                                            3293.0
4                                            3503.0
..                                              …
169                                          3275.0
170                                          2346.0
171                                          3682.0
172                                          2129.0
173                                          2383.0

[174 rows x 3 columns]
```

[92]:
```
#Affichage des 10 pays qui ont le moins de dispo alimentaire par personne
pays_moins_disponibilite_alimentaire = dispo_ali_total[['Zone', 'Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)']].sort_values(by='Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)', ascending=True)
```

[93]:
```
top_10_pays_disponibilite_alimentaire_moins =␣
 ↪pays_moins_disponibilite_alimentaire.head(10)
```

[94]:
```
print("Les 10 pays qui ont le moins de disponibilité alimentaire par personne :
 ↪")
print(top_10_pays_disponibilite_alimentaire_moins)
```

```
Les 10 pays qui ont le moins de disponibilité alimentaire par personne :
                                        Zone  \
128                  République centrafricaine
166                                   Zambie
91                                 Madagascar
0                                  Afghanistan
65                                       Haïti
133  République populaire démocratique de Corée
151                                     Tchad
167                                  Zimbabwe
114                                   Ouganda
154                                Timor-Leste

     Disponibilité alimentaire (Kcal/personne/jour)
128                                          1879.0
166                                          1924.0
91                                           2056.0
0                                            2087.0
65                                           2089.0
133                                          2093.0
151                                          2109.0
167                                          2113.0
114                                          2126.0
154                                          2129.0
```

3.10 - Pays avec le plus de disponibilité par habitant

[95]:
```
#Affichage des 10 pays qui ont le plus de dispo alimentaire par personne
top_10_pays_disponibilite_alimentaire = dispo_ali_total[['Zone', 'Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)']].sort_values(by='Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)', ascending=False)
```

[96]:
```
print("Les 10 pays ayant la plus grande disponibilité alimentaire par personne :
 ↪")
print(top_10_pays_disponibilite_alimentaire[['Zone', 'Disponibilité alimentaire␣
 ↪(Kcal/personne/jour)']].head(10))
```

```
Les 10 pays ayant la plus grande disponibilité alimentaire par personne :
                      Zone  Disponibilité alimentaire (Kcal/personne/jour)
11                Autriche                                          3770.0
16                Belgique                                          3737.0
159                Turquie                                          3708.0
171  États-Unis d'Amérique                                          3682.0
74                  Israël                                          3610.0
```

```
72                 Irlande                                    3602.0
75                  Italie                                    3578.0
89              Luxembourg                                    3540.0
168                 Égypte                                    3518.0
4                Allemagne                                    3503.0
```

3.11 - Exemple de la Thaïlande pour le Manioc

[97]:
```
#création d'un dataframe avec uniquement la Thaïlande
df_thailande = df_pop_sous_nut.loc[df_pop_sous_nut['Zone'] == 'Thaïlande']
```

[98]:
```
df_thailande
```

[98]:
```
           Zone  Année_pop  Population Année_sous_nut  sous_nutrition  \
185  Thaïlande       2017  69209810.0      2016-2018       6200000.0

     proportion
185        8.96
```

[99]:
```
#Calcul de la sous nutrition en Thaïlande['population']
sous_nutrition_thailande = df_thailande['sous_nutrition']/
 ↪df_thailande['Population']*100
```

[100]:
```
print("Proportion de la population en sous-nutrition en Thaïlande :",␣
 ↪sous_nutrition_thailande)
```

```
Proportion de la population en sous-nutrition en Thaïlande : 185    8.958268
dtype: float64
```

[101]:
```
# On calcule la proportion exportée en fonction de la proportion
thai_manioc = dispo_alimentaire.loc[(dispo_alimentaire['Produit'] == "Manioc")␣
 ↪& (dispo_alimentaire['Zone'] == "Thaïlande"),:]
print('Proportion de manioc exportée :', "{:.2f}".
 ↪format(thai_manioc['Exportations - Quantité'].iloc[0]*100 /
 ↪thai_manioc['Production'].iloc[0]), "%")
```

```
Proportion de manioc exportée : 83.41 %
```

[102]:
```
disponibilite_par_habitant_thailande =␣
 ↪dispo_alimentaire[(dispo_alimentaire['Zone'] == 'Thaïlande')]['Disponibilité␣
 ↪alimentaire (Kcal/personne/jour)'].sum()
print("La disponibilité alimentaire par habitant en Thaïlande est de:",␣
 ↪disponibilite_par_habitant_thailande, "(Kcal/personne/jour)")
```

```
La disponibilité alimentaire par habitant en Thaïlande est de: 2785.0
(Kcal/personne/jour)
```

Etape 6 - Analyse complémentaires

[ ]:
```
#Rajouter en dessous toutes les analyses complémtaires suite à la demande de␣
↪mélanie :
#"et toutes les infos que tu trouverais utiles pour mettre en relief les pays␣
↪qui semblent être
#le plus en difficulté au niveau alimentaire"
```

[ ]:
```
Japon kilocalories négatives ?
```

[114]:
```
#création d'un dataframe avec uniquement la France
df_Inde = df_pop_sous_nut.loc[df_pop_sous_nut['Zone'] == 'Inde']
```

[115]:
```
#Calcul de la sous nutrition en France['population']
sous_nutrition_Inde = df_Inde['sous_nutrition']/df_Inde['Population']*100
```

[116]:
```
print("Proportion de la population en sous-nutrition en Inde:",␣
↪sous_nutrition_Inde)
```

```
Proportion de la population en sous-nutrition en Inde: 84    14.20059
dtype: float64
```

[ ]:
```

```