Institute of Technology of Cambodia (ITC)

Master of Data Science

Subject: Programming for Data Science

Final Report of Programming for Data Science

Name: ENG KHUN

ID: M080101

In [1]:
```
import pandas as pd
import math
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
# load data from csv file
df = pd.read_csv('uncomtrade.csv')
df
```

Out[3]:

	Classification	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Trade Flow	Reporter Code	Rep
0	H3	2010	2010	2010	4	0	0	Import	116	Camb
1	H3	2010	2010	2010	4	0	0	Import	116	Camb
2	H3	2010	2010	2010	4	0	0	Import	116	Camb
3	H3	2010	2010	2010	4	0	0	Import	116	Camb
4	H3	2010	2010	2010	4	0	0	Export	116	Camb
...	...	...	...	...	...	...	...	...	...	...
18625	H5	2021	2021	2021	4	0	1	Import	116	Camb
18626	H5	2021	2021	2021	4	0	1	Import	116	Camb
18627	H5	2021	2021	2021	2	0	1	Import	116	Camb
18628	H5	2021	2021	2021	2	0	2	Export	116	Camb
18629	H5	2021	2021	2021	2	0	4	Import	116	Camb

18630 rows × 37 columns

In [11]:
```
#Sort the first 20 rows
df.head(n=20)
```

Out[11]:

	Classification	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Trade Flow	Reporter Code	Reporte
0	H3	2010	2010	2010	4	0	0	Import	116	Cambodi
1	H3	2010	2010	2010	4	0	0	Import	116	Cambodi
2	H3	2010	2010	2010	4	0	0	Import	116	Cambodi
3	H3	2010	2010	2010	4	0	0	Import	116	Cambodi
4	H3	2010	2010	2010	4	0	0	Export	116	Cambodi
5	H3	2011	2011	2011	4	0	0	Import	116	Cambodi
6	H3	2011	2011	2011	4	0	0	Import	116	Cambodi
7	H3	2011	2011	2011	4	0	0	Export	116	Cambodi
8	H3	2011	2011	2011	4	0	0	Import	116	Cambodi
9	H3	2011	2011	2011	4	0	0	Import	116	Cambodi
10	H3	2011	2011	2011	4	0	0	Export	116	Cambodi
11	H4	2012	2012	2012	4	0	0	Import	116	Cambodi
12	H4	2012	2012	2012	4	0	0	Export	116	Cambodi
13	H4	2012	2012	2012	4	0	0	Import	116	Cambodi
14	H4	2012	2012	2012	4	0	0	Import	116	Cambodi
15	H4	2012	2012	2012	4	0	0	Export	116	Cambodi
16	H4	2013	2013	2013	4	0	0	Import	116	Cambodi
17	H4	2013	2013	2013	4	0	0	Import	116	Cambodi
18	H4	2013	2013	2013	4	0	0	Import	116	Cambodi
19	H4	2013	2013	2013	4	0	0	Export	116	Cambodi

20 rows × 37 columns

In [6]:
```
#Sort the last 20 rows
df.tail(n=20)
```

Out[6]:

	Classification	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Trade Flow	Reporter Code	Rep
18610	H5	2021	2021	2021	4	0	4	Import	116	Camb
18611	H5	2021	2021	2021	4	0	1	Import	116	Camb
18612	H5	2021	2021	2021	4	0	1	Import	116	Camb
18613	H5	2021	2021	2021	4	0	1	Import	116	Camb
18614	H5	2021	2021	2021	4	0	1	Import	116	Camb
18615	H5	2021	2021	2021	4	0	2	Export	116	Camb
18616	H5	2021	2021	2021	4	0	1	Import	116	Camb
18617	H5	2021	2021	2021	4	0	2	Export	116	Camb
18618	H5	2021	2021	2021	4	0	1	Import	116	Camb
18619	H5	2021	2021	2021	4	0	1	Import	116	Camb
18620	H5	2021	2021	2021	4	0	2	Export	116	Camb
18621	H5	2021	2021	2021	4	0	1	Import	116	Camb
18622	H5	2021	2021	2021	4	0	2	Export	116	Camb
18623	H5	2021	2021	2021	4	0	1	Import	116	Camb
18624	H5	2021	2021	2021	4	0	1	Import	116	Camb
18625	H5	2021	2021	2021	4	0	1	Import	116	Camb
18626	H5	2021	2021	2021	4	0	1	Import	116	Camb
18627	H5	2021	2021	2021	2	0	1	Import	116	Camb
18628	H5	2021	2021	2021	2	0	2	Export	116	Camb
18629	H5	2021	2021	2021	2	0	4	Import	116	Camb

20 rows × 37 columns

In [8]:
```
#The varible of dataset check
df.columns
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18630 entries, 0 to 18629
Data columns (total 37 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Classification          18630 non-null  object 
 1   Year                    18630 non-null  int64  
 2   Period                  18630 non-null  int64  
 3   Period Desc.            18630 non-null  int64  
 4   Aggregate Level         18630 non-null  int64  
 5   Is Leaf Code            18630 non-null  int64  
 6   Trade Flow Code         18630 non-null  int64  
 7   Trade Flow              18630 non-null  object 
 8   Reporter Code           18630 non-null  int64  
 9   Reporter                18630 non-null  object 
 10  Reporter ISO            18630 non-null  object 
 11  Partner Code            18630 non-null  int64  
 12  Partner                 18630 non-null  object 
 13  Partner ISO             18630 non-null  object 
 14  2nd Partner Code        16504 non-null  float64
 15  2nd Partner             16504 non-null  object 
 16  2nd Partner ISO         16504 non-null  object 
 17  Customs Proc. Code      16504 non-null  object 
 18  Customs                 16504 non-null  object 
 19  Mode of Transport Code  16504 non-null  float64
 20  Mode of Transport       16504 non-null  object 
 21  Commodity Code          18630 non-null  int64  
 22  Gen_Code                18630 non-null  int64  
 23  2-Digit                 18630 non-null  int64  
 24  Commodity               18630 non-null  object 
 25  Qty Unit Code           18630 non-null  int64  
 26  Qty Unit                14947 non-null  object 
 27  Qty                     16268 non-null  float64
 28  Alt Qty Unit Code       16504 non-null  float64
 29  Alt Qty Unit            2097 non-null   object 
 30  Alt Qty                 6491 non-null   float64
 31  Netweight (kg)          17652 non-null  float64
 32  Gross weight (kg)       6491 non-null   float64
 33  Trade Value (US$)       18630 non-null  int64  
 34  CIF Trade Value (US$)   11687 non-null  float64
 35  FOB Trade Value (US$)   6389 non-null   float64
 36  Flag                    18630 non-null  int64  
dtypes: float64(9), int64(14), object(14)
memory usage: 5.3+ MB
```

In [13]:
```
df.isnull().values.any()
```

Out[13]: True

In [14]:
```
df.isnull().sum().sum()
```

Out[14]: 84026

In [15]:
```
df['Commodity'].isnull().values.any()
```

Out[15]: False

In [16]:
```
df['2-Digit'].isnull().values.any()
```

Out[16]: False

In [24]:
```
df['Netweight (kg)'].isnull().values.any()
df['Netweight (kg)'].isnull().sum().sum()
```

Out[24]: 978

In [26]:
```
df['Trade Value (US$)'].isnull().values.any()
df['Trade Value (US$)'].isnull().sum().sum()
```

Out[26]: False

In [27]:
```
df.describe(include = 'number').round(decimals = 2)
```

Out[27]:

	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Reporter Code	Partner Code	Part C
count	18630.00	18630.00	18630.00	18630.00	18630.00	18630.00	18630.0	18630.0	165
mean	2015.73	2015.73	2015.73	4.00	0.00	0.16	116.0	0.0	
std	3.40	3.40	3.40	0.07	0.03	0.48	0.0	0.0	
min	2010.00	2010.00	2010.00	2.00	0.00	0.00	116.0	0.0	
25%	2013.00	2013.00	2013.00	4.00	0.00	0.00	116.0	0.0	
50%	2016.00	2016.00	2016.00	4.00	0.00	0.00	116.0	0.0	
75%	2019.00	2019.00	2019.00	4.00	0.00	0.00	116.0	0.0	
max	2021.00	2021.00	2021.00	6.00	1.00	4.00	116.0	0.0	

8 rows × 23 columns

In [28]:
```
df.describe(include = 'object')
```

Out[28]:

	Classification	Trade Flow	Reporter	Reporter ISO	Partner	Partner ISO	2nd Partner	2nd Partner ISO	Custo Pr Co
count	18630	18630	18630	18630	18630	18630	16504	16504	165
unique	3	2	1	1	1	2	1	1	
top	H5	Import	Cambodia	KHM	World	W00	World	W00	C
freq	8322	12577	18630	18630	18630	16504	16504	16504	165

In [29]:

```
# Linear (Pearson) correlation
df_corr = df.corr(method = "pearson").round(decimals = 2)
df_corr
```

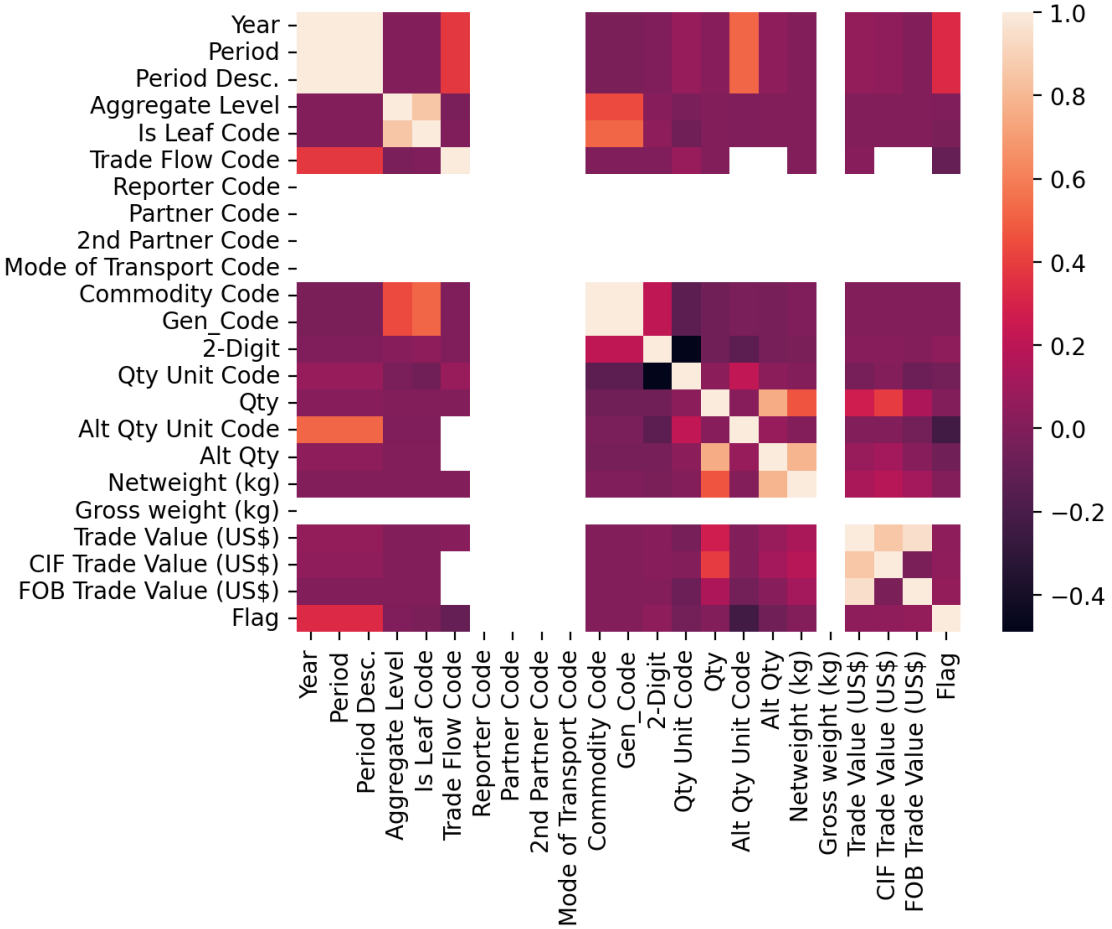
Out[29]:

	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Reporter Code	Partner Code	2nd Partner Code	Mc Tran
Year	1.00	1.00	1.00	0.00	-0.00	0.38	NaN	NaN	NaN	
Period	1.00	1.00	1.00	0.00	-0.00	0.38	NaN	NaN	NaN	
Period Desc.	1.00	1.00	1.00	0.00	-0.00	0.38	NaN	NaN	NaN	
Aggregate Level	0.00	0.00	0.00	1.00	0.85	-0.03	NaN	NaN	NaN	
Is Leaf Code	-0.00	-0.00	-0.00	0.85	1.00	-0.01	NaN	NaN	NaN	
Trade Flow Code	0.38	0.38	0.38	-0.03	-0.01	1.00	NaN	NaN	NaN	
Reporter Code	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
Partner Code	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2nd Partner Code	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
Mode of Transport Code	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
Commodity Code	-0.02	-0.02	-0.02	0.44	0.52	-0.01	NaN	NaN	NaN	
Gen_Code	-0.02	-0.02	-0.02	0.44	0.52	-0.01	NaN	NaN	NaN	
2-Digit	-0.01	-0.01	-0.01	0.02	0.04	-0.01	NaN	NaN	NaN	
Qty Unit Code	0.07	0.07	0.07	-0.03	-0.06	0.08	NaN	NaN	NaN	
Qty	0.02	0.02	0.02	-0.00	-0.00	0.00	NaN	NaN	NaN	
Alt Qty Unit Code	0.52	0.52	0.52	-0.01	-0.01	NaN	NaN	NaN	NaN	
Alt Qty	0.04	0.04	0.04	-0.00	-0.00	NaN	NaN	NaN	NaN	
Netweight (kg)	0.00	0.00	0.00	-0.00	-0.00	0.00	NaN	NaN	NaN	
Gross weight (kg)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
Trade Value (US$)	0.06	0.06	0.06	-0.00	-0.00	0.02	NaN	NaN	NaN	
CIF Trade Value (US$)	0.05	0.05	0.05	-0.00	-0.00	NaN	NaN	NaN	NaN	
FOB Trade Value (US$)	0.00	0.00	0.00	-0.00	-0.00	NaN	NaN	NaN	NaN	
Flag	0.33	0.33	0.33	-0.01	-0.02	-0.10	NaN	NaN	NaN	

23 rows × 23 columns

In [37]:
```
%config InlineBackend.figure_format='retina'
```

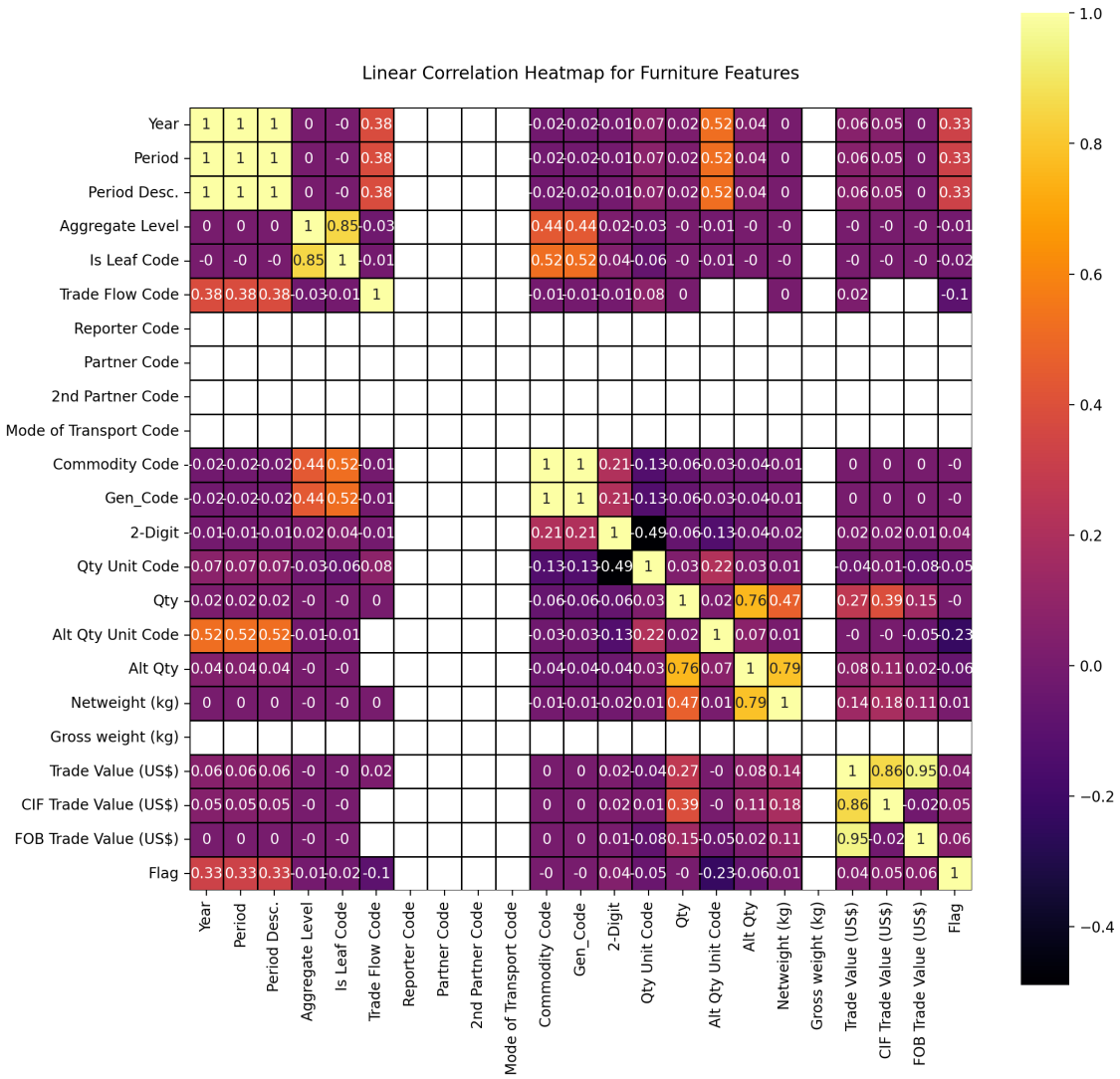
In [58]:
```
# Instantiating a heatmap
sns.heatmap(df_corr)

# Displaying the plot
plt.show()
```



In [51]:

```
# Specifying plot size (making it bigger)
fig, ax = plt.subplots(figsize=(12,12))


# Developing a spicy heatmap
sns.heatmap(data       = df_corr, # the correlation matrix
            cmap       = 'inferno',     # changing to SPICY colors
            square     = True,          # tightening the layout
            annot      = True,          # should there be numbers in the he
            linecolor  = 'black',       # lines between boxes
            linewidths = 0.5)          # how thick should the lines be?


# Title and displaying the plot
plt.title("""
Linear Correlation Heatmap for Furniture Features
""")
```

Out[51]: Text(0.5, 1.0, '\nLinear Correlation Heatmap for Furniture Features\n')



In [64]:
```
df
```

3	H3	2010	2010	2010	4	0	0	Import	Cambodia
4	H3	2010	2010	2010	4	0	0	Export	Cambodia
...	...	...	...	...	...	...	...	...	...
18625	H5	2021	2021	2021	4	0	1	Import	Cambodia
18626	H5	2021	2021	2021	4	0	1	Import	Cambodia
18627	H5	2021	2021	2021	2	0	1	Import	Cambodia
18628	H5	2021	2021	2021	2	0	2	Export	Cambodia
18629	H5	2021	2021	2021	2	0	4	Import	Cambodia

18630 rows × 32 columns

In [74]:

```
# Drop non-related column in dataframe
df.drop(['Classification'], axis=1, inplace=True)
```

```
---------------------------------------------------------------------------
--
KeyError                                  Traceback (most recent call las
t)
~\AppData\Local\Temp\ipykernel_12592\1615959713.py in <module>
      1 # Drop non-related column in dataframe
----> 2 df.drop(['Classification','Trade Flow Code','Period', 'Period Des
c.', 'Aggregate Level', 'Is Leaf Code'], axis=1, inplace=True)

C:\ProgramData\Anaconda3\lib\site-packages\pandas\util\_decorators.py in
wrapper(*args, **kwargs)
    309                     stacklevel=stacklevel,
    310                 )
--> 311             return func(*args, **kwargs)
    312
    313         return wrapper

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\frame.py in drop(s
elf, labels, axis, index, columns, level, inplace, errors)
   4955                 weight  1.0     0.8
   4956         """
-> 4957         return super().drop(
   4958             labels=labels,
   4959             axis=axis,

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in drop
(self, labels, axis, index, columns, level, inplace, errors)
   4265         for axis, labels in axes.items():
   4266             if labels is not None:
-> 4267                 obj = obj._drop_axis(labels, axis, level=level, e
rrors=errors)
   4268
   4269         if inplace:

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in _dro
p_axis(self, labels, axis, level, errors, consolidate, only_slice)
   4309                 new_axis = axis.drop(labels, level=level, errors=
errors)
   4310             else:
-> 4311                 new_axis = axis.drop(labels, errors=errors)
   4312             indexer = axis.get_indexer(new_axis)
   4313

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexes\base.py in
drop(self, labels, errors)
   6659         if mask.any():
   6660             if errors != "ignore":
-> 6661                 raise KeyError(f"{list(labels[mask])} not found i
n axis")
   6662             indexer = indexer[~mask]
   6663         return self.delete(indexer)

KeyError: "['Classification'] not found in axis"
```

In [76]:
```
df.columns
```

Out[76]:
```
Index(['Year', 'Period', 'Period Desc.', 'Aggregate Level', 'Is Leaf Cod
e',
       'Trade Flow Code', 'Trade Flow', 'Reporter', 'Reporter ISO', 'Part
ner',
       'Partner ISO', '2nd Partner', '2nd Partner ISO', 'Customs Proc. Co
de',
       'Customs', 'Mode of Transport', 'Commodity Code', 'Gen_Code', '2-D
igit',
       'Commodity', 'Qty Unit Code', 'Qty Unit', 'Qty', 'Alt Qty Unit Cod
e',
       'Alt Qty Unit', 'Alt Qty', 'Netweight (kg)', 'Trade Value (US$)',
       'CIF Trade Value (US$)', 'FOB Trade Value (US$)', 'Flag'],
      dtype='object')
```

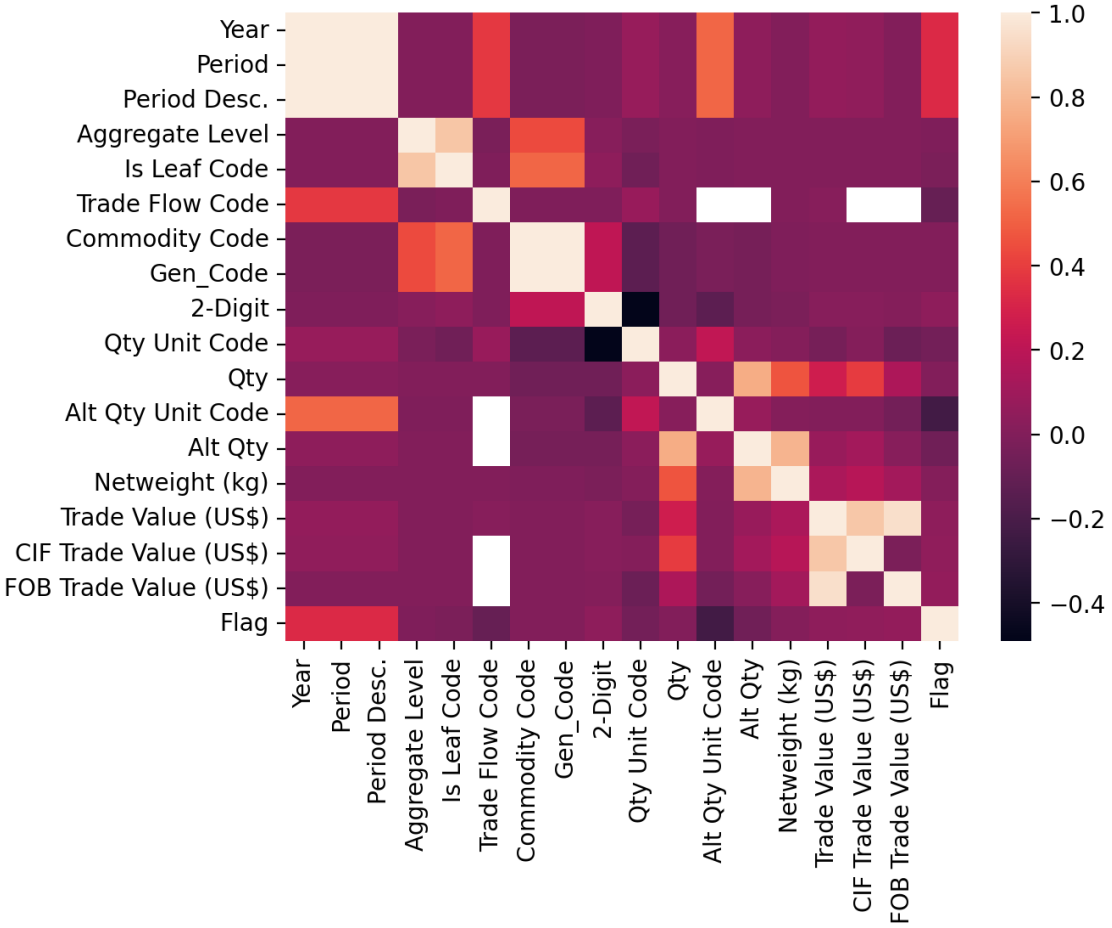
In [70]:
```
df_corr1 = df.corr(method = "pearson").round(decimals = 2)
df_corr1
```

Out[70]:

	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Commodity Code	Gen_Code	2-Digit	C
Year	1.00	1.00	1.00	0.00	-0.00	0.38	-0.02	-0.02	-0.01	
Period	1.00	1.00	1.00	0.00	-0.00	0.38	-0.02	-0.02	-0.01	
Period Desc.	1.00	1.00	1.00	0.00	-0.00	0.38	-0.02	-0.02	-0.01	
Aggregate Level	0.00	0.00	0.00	1.00	0.85	-0.03	0.44	0.44	0.02	-
Is Leaf Code	-0.00	-0.00	-0.00	0.85	1.00	-0.01	0.52	0.52	0.04	-
Trade Flow Code	0.38	0.38	0.38	-0.03	-0.01	1.00	-0.01	-0.01	-0.01	
Commodity Code	-0.02	-0.02	-0.02	0.44	0.52	-0.01	1.00	1.00	0.21	-
Gen_Code	-0.02	-0.02	-0.02	0.44	0.52	-0.01	1.00	1.00	0.21	-
2-Digit	-0.01	-0.01	-0.01	0.02	0.04	-0.01	0.21	0.21	1.00	-
Qty Unit Code	0.07	0.07	0.07	-0.03	-0.06	0.08	-0.13	-0.13	-0.49	
Qty	0.02	0.02	0.02	-0.00	-0.00	0.00	-0.06	-0.06	-0.06	
Alt Qty Unit Code	0.52	0.52	0.52	-0.01	-0.01	NaN	-0.03	-0.03	-0.13	
Alt Qty	0.04	0.04	0.04	-0.00	-0.00	NaN	-0.04	-0.04	-0.04	
Netweight (kg)	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.01	-0.01	-0.02	
Trade Value (US$)	0.06	0.06	0.06	-0.00	-0.00	0.02	0.00	0.00	0.02	-
CIF Trade Value (US$)	0.05	0.05	0.05	-0.00	-0.00	NaN	0.00	0.00	0.02	
FOB Trade Value (US$)	0.00	0.00	0.00	-0.00	-0.00	NaN	0.00	0.00	0.01	-
Flag	0.33	0.33	0.33	-0.01	-0.02	-0.10	-0.00	-0.00	0.04	-

In [71]:

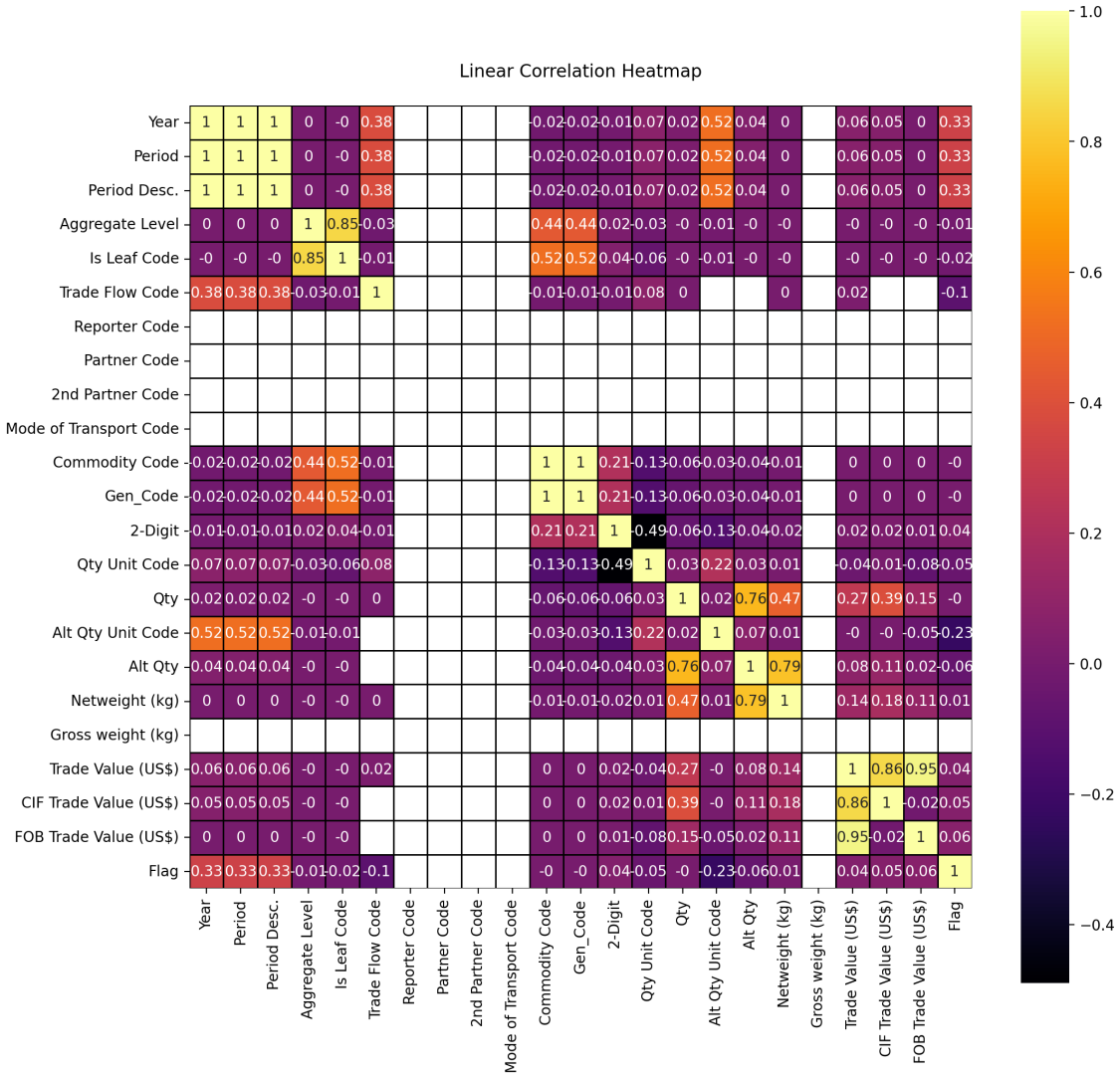
```
# Instantiating a heatmap
sns.heatmap(df_corr1)

# Displaying the plot
plt.show()
```



In [50]:

```
# Specifying plot size (making it bigger)
fig, ax = plt.subplots(figsize=(12,12))


# Developing a spicy heatmap
sns.heatmap(data       = df_corr, # the correlation matrix
            cmap       = 'inferno',      # changing to SPICY colors
            square     = True,           # tightening the layout
            annot      = True,           # should there be numbers in the he
            linecolor  = 'black',        # lines between boxes
            linewidths = 0.5)          # how thick should the lines be?


# Title and displaying the plot
plt.title("""
Linear Correlation Heatmap
""")
```

Out[50]: Text(0.5, 1.0, '\nLinear Correlation Heatmap\n')



In [63]:

```
df.columns
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18630 entries, 0 to 18629
Data columns (total 32 columns):
 #   Column                 Non-Null Count  Dtype  
---  ------                 --------------  -----  
 0   Classification         18630 non-null  object 
 1   Year                   18630 non-null  int64  
 2   Period                 18630 non-null  int64  
 3   Period Desc.           18630 non-null  int64  
 4   Aggregate Level        18630 non-null  int64  
 5   Is Leaf Code           18630 non-null  int64  
 6   Trade Flow Code        18630 non-null  int64  
 7   Trade Flow             18630 non-null  object 
 8   Reporter               18630 non-null  object 
 9   Reporter ISO           18630 non-null  object 
 10  Partner                18630 non-null  object 
 11  Partner ISO            18630 non-null  object 
 12  2nd Partner            16504 non-null  object 
 13  2nd Partner ISO        16504 non-null  object 
 14  Customs Proc. Code     16504 non-null  object 
 15  Customs                16504 non-null  object 
 16  Mode of Transport      16504 non-null  object 
 17  Commodity Code         18630 non-null  int64  
 18  Gen_Code               18630 non-null  int64  
 19  2-Digit                18630 non-null  int64  
 20  Commodity              18630 non-null  object 
 21  Qty Unit Code          18630 non-null  int64  
 22  Qty Unit               14947 non-null  object 
 23  Qty                    16268 non-null  float64
 24  Alt Qty Unit Code      16504 non-null  float64
 25  Alt Qty Unit           2097 non-null   object 
 26  Alt Qty                6491 non-null   float64
 27  Netweight (kg)         17652 non-null  float64
 28  Trade Value (US$)      18630 non-null  int64  
 29  CIF Trade Value (US$)  11687 non-null  float64
 30  FOB Trade Value (US$)  6389 non-null   float64
 31  Flag                   18630 non-null  int64  
dtypes: float64(6), int64(12), object(14)
memory usage: 4.5+ MB
```

In [80]:
```
df.drop(['Trade Flow Code','Period', 'Period Desc.', 'Aggregate Level', 'Is
```

```
---------------------------------------------------------------------------
--
KeyError                                  Traceback (most recent call las
t)
~\AppData\Local\Temp\ipykernel_12592\3489206375.py in <module>
----> 1 df.drop(['Trade Flow Code','Period', 'Period Desc.', 'Aggregate L
evel', 'Is Leaf Code'],axis=1, inplace=True)

C:\ProgramData\Anaconda3\lib\site-packages\pandas\util\_decorators.py in
wrapper(*args, **kwargs)
    309                     stacklevel=stacklevel,
    310                 )
--> 311             return func(*args, **kwargs)
    312
    313         return wrapper

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\frame.py in drop(s
elf, labels, axis, index, columns, level, inplace, errors)
   4955                 weight  1.0     0.8
   4956         """
-> 4957         return super().drop(
   4958             labels=labels,
   4959             axis=axis,

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in drop
(self, labels, axis, index, columns, level, inplace, errors)
   4265         for axis, labels in axes.items():
   4266             if labels is not None:
-> 4267                 obj = obj._drop_axis(labels, axis, level=level, e
rrors=errors)
   4268
   4269         if inplace:

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in _dro
p_axis(self, labels, axis, level, errors, consolidate, only_slice)
   4309                 new_axis = axis.drop(labels, level=level, errors=
errors)
   4310             else:
-> 4311                 new_axis = axis.drop(labels, errors=errors)
   4312             indexer = axis.get_indexer(new_axis)
   4313

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexes\base.py in
drop(self, labels, errors)
   6659         if mask.any():
   6660             if errors != "ignore":
-> 6661                 raise KeyError(f"{list(labels[mask])} not found i
n axis")
   6662             indexer = indexer[~mask]
   6663         return self.delete(indexer)

KeyError: "['Trade Flow Code', 'Period', 'Period Desc.', 'Aggregate Leve
l', 'Is Leaf Code'] not found in axis"
```

In [81]:
```
df.columns
```

Out[81]:
```
Index(['Year', 'Trade Flow', 'Reporter', 'Reporter ISO', 'Partner',
       'Partner ISO', '2nd Partner', '2nd Partner ISO', 'Customs Proc. Co
de',
       'Customs', 'Mode of Transport', 'Commodity Code', 'Gen_Code', '2-D
igit',
       'Commodity', 'Qty Unit Code', 'Qty Unit', 'Qty', 'Alt Qty Unit Cod
e',
       'Alt Qty Unit', 'Alt Qty', 'Netweight (kg)', 'Trade Value (US$)',
       'CIF Trade Value (US$)', 'FOB Trade Value (US$)', 'Flag'],
      dtype='object')
```

In [82]:
```
df
```

Out[82]:

	Year	Trade Flow	Reporter	Reporter ISO	Partner	Partner ISO	2nd Partner	2nd Partner ISO	Customs Proc. Code	Custo
0	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
1	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
2	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
3	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
4	2010	Export	Cambodia	KHM	World	W00	World	W00	C00	TO
...	...	...	...	...	...	...	...	...	...	
18625	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18626	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18627	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18628	2021	Export	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18629	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	

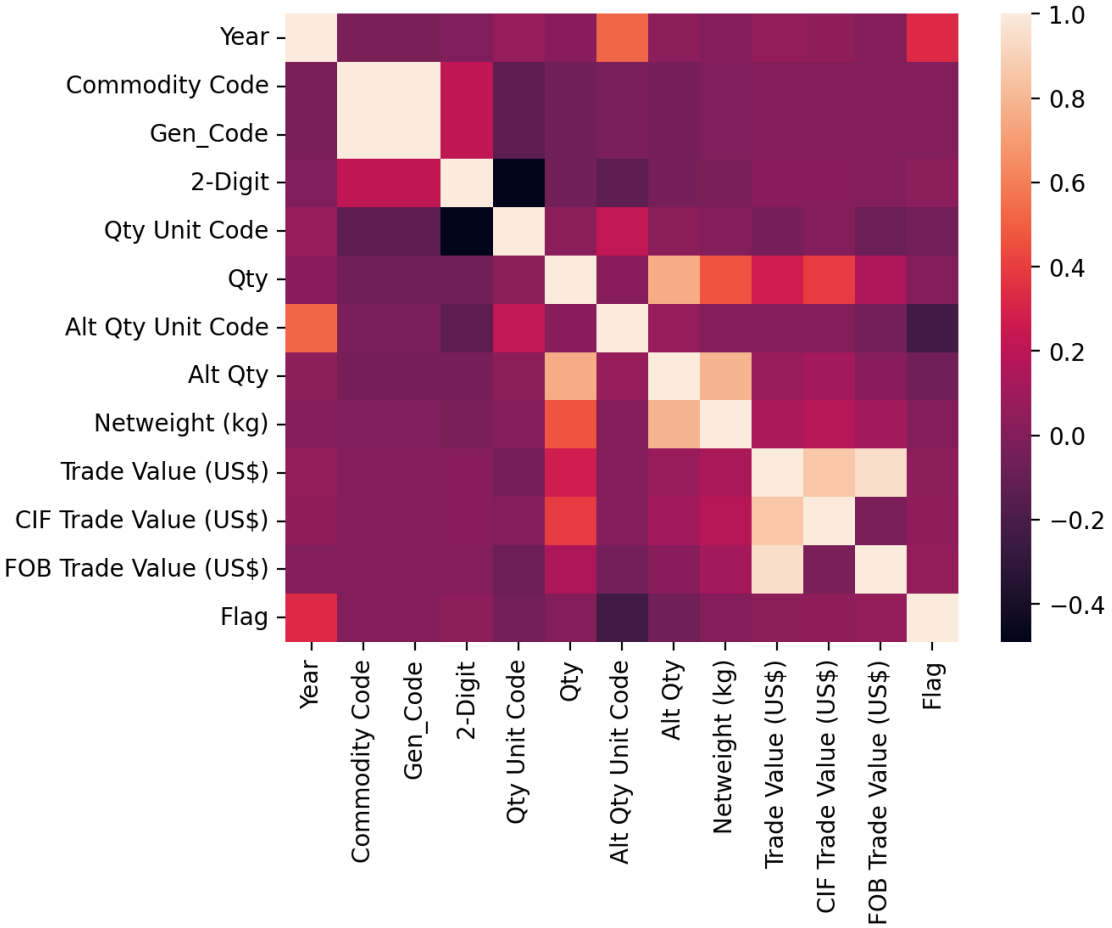
18630 rows × 26 columns

In [99]:
```
df.shape
```

Out[99]:
```
(18630, 16)
```

In [84]:

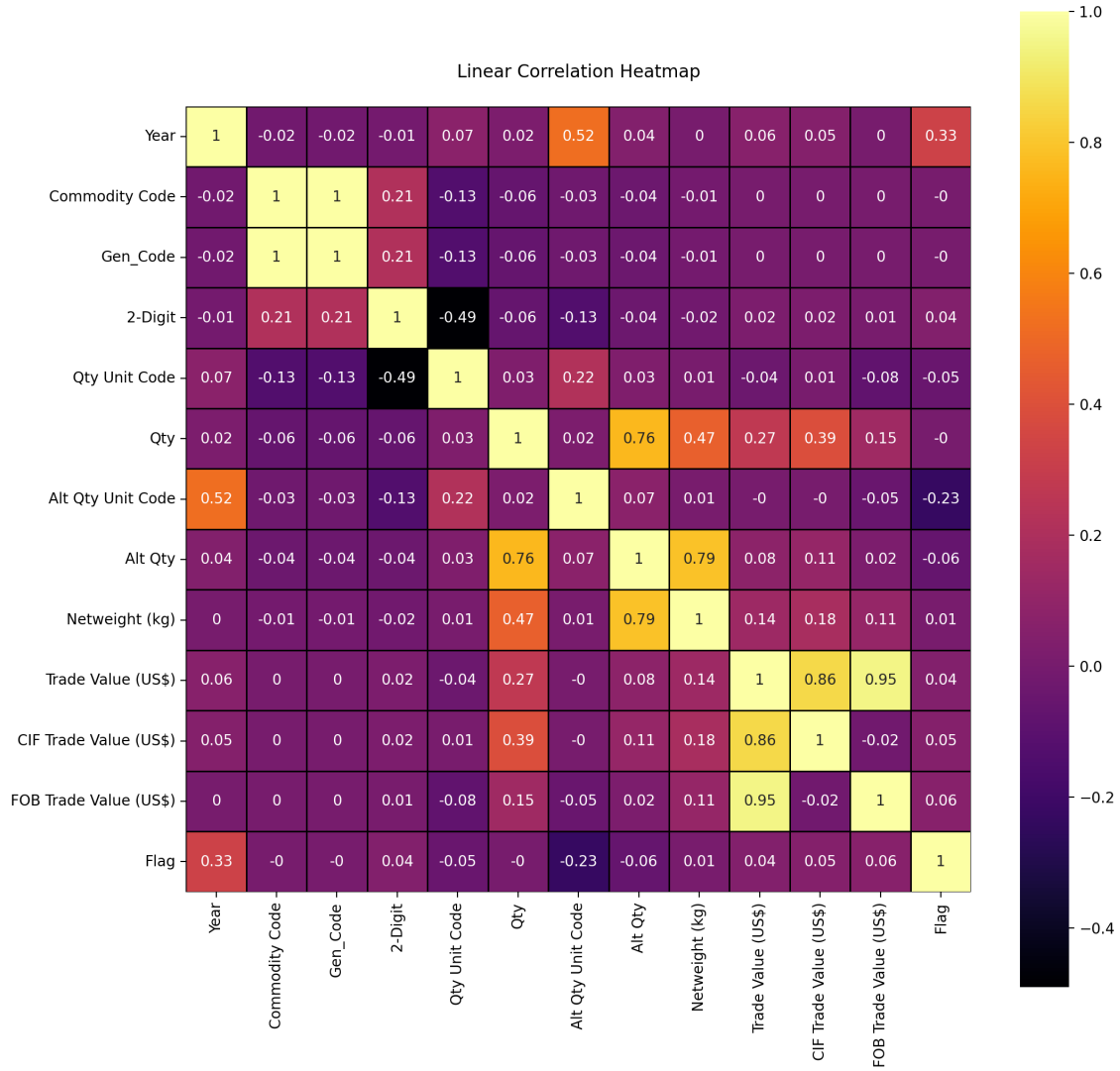
```
# Linear (Pearson) correlation
df_corr2 = df.corr(method = "pearson").round(decimals = 2)
df_corr2
# Instantiating a heatmap
sns.heatmap(df_corr2)

# Displaying the plot
plt.show()
```



In [85]:

```
# Specifying plot size (making it bigger)
fig, ax = plt.subplots(figsize=(12,12))


# Developing a spicy heatmap
sns.heatmap(data       = df_corr2, # the correlation matrix
            cmap       = 'inferno',     # changing to SPICY colors
            square     = True,          # tightening the layout
            annot      = True,          # should there be numbers in the he
            linecolor  = 'black',       # lines between boxes
            linewidths = 0.5)          # how thick should the lines be?


# Title and displaying the plot
plt.title("""
Linear Correlation Heatmap
""")
```

Out[85]: Text(0.5, 1.0, '\nLinear Correlation Heatmap\n')



In [87]:
```
df
df.columns
```

Out[87]:
```
Index(['Year', 'Trade Flow', 'Reporter', 'Reporter ISO', 'Partner',
       'Partner ISO', '2nd Partner', '2nd Partner ISO', 'Customs Proc. Co
de',
       'Customs', 'Mode of Transport', 'Commodity Code', 'Gen_Code', '2-D
igit',
       'Commodity', 'Qty Unit Code', 'Qty Unit', 'Qty', 'Alt Qty Unit Cod
e',
       'Alt Qty Unit', 'Alt Qty', 'Netweight (kg)', 'Trade Value (US$)',
       'CIF Trade Value (US$)', 'FOB Trade Value (US$)', 'Flag'],
      dtype='object')
```

In [89]:
```
df.drop(['Commodity Code', 'Gen_Code', 'Qty Unit Code', 'Qty Unit', 'Qty',
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_12592\870476761.py in <module>
----> 1 df.drop(['Commodity Code', 'Gen_Code', 'Qty Unit Code', 'Qty Unit', 'Qty', 'Alt Qty Unit Code'],axis=1, inplace=True)

C:\ProgramData\Anaconda3\lib\site-packages\pandas\util\_decorators.py in wrapper(*args, **kwargs)
    309                     stacklevel=stacklevel,
    310                 )
--> 311             return func(*args, **kwargs)
    312
    313         return wrapper

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\frame.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   4955                 weight  1.0     0.8
   4956         """
-> 4957         return super().drop(
   4958             labels=labels,
   4959             axis=axis,

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   4265         for axis, labels in axes.items():
   4266             if labels is not None:
-> 4267                 obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4268
   4269         if inplace:

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\generic.py in _drop_axis(self, labels, axis, level, errors, consolidate, only_slice)
   4309                 new_axis = axis.drop(labels, level=level, errors=errors)
   4310             else:
-> 4311                 new_axis = axis.drop(labels, errors=errors)
   4312             indexer = axis.get_indexer(new_axis)
   4313

C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\indexes\base.py in drop(self, labels, errors)
   6659         if mask.any():
   6660             if errors != "ignore":
-> 6661                 raise KeyError(f"{list(labels[mask])} not found in axis")
   6662             indexer = indexer[~mask]
   6663         return self.delete(indexer)

KeyError: "['Commodity Code', 'Gen_Code', 'Qty Unit Code', 'Qty Unit', 'Qty', 'Alt Qty Unit Code'] not found in axis"
```

In [90]:
```
df
```

Out[90]:

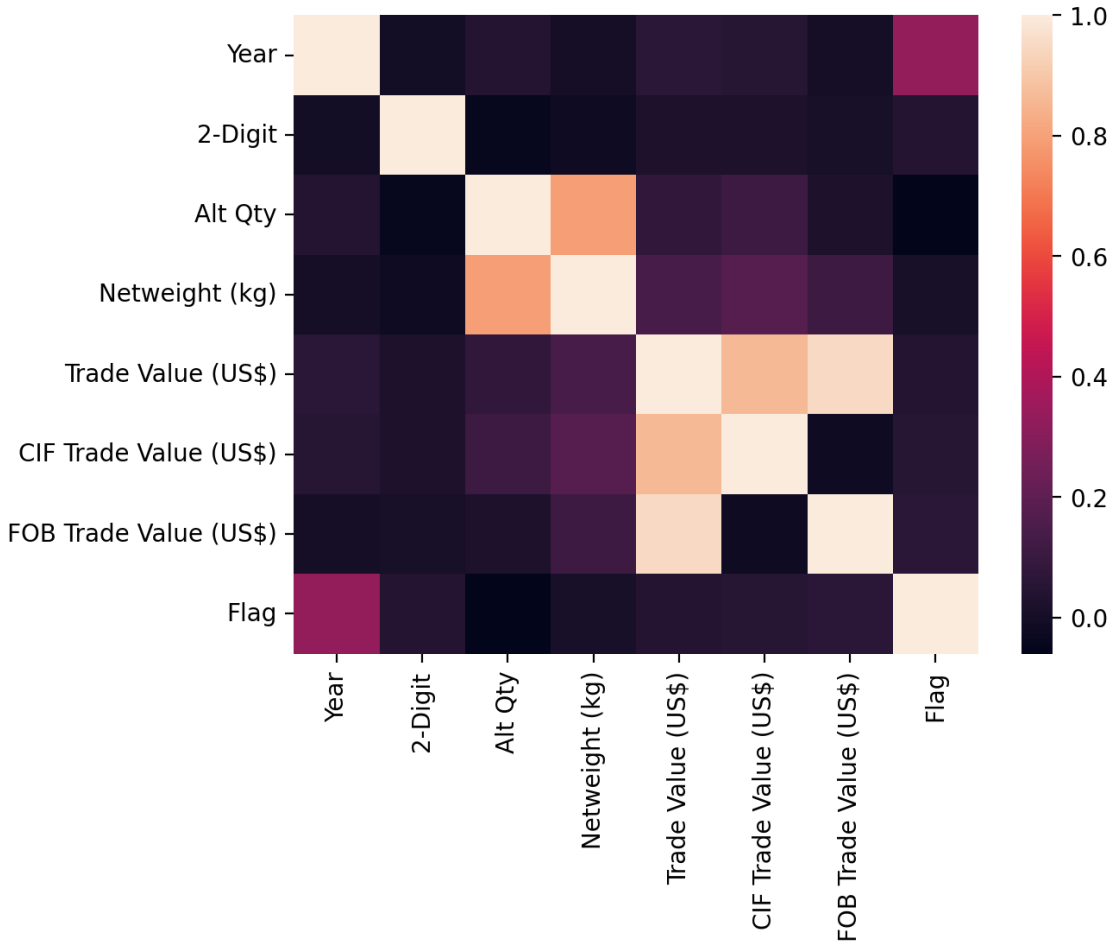
	Year	Trade Flow	Reporter	Reporter ISO	Partner	Partner ISO	2nd Partner	2nd Partner ISO	Customs Proc. Code	Custo
0	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
1	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
2	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
3	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
4	2010	Export	Cambodia	KHM	World	W00	World	W00	C00	TO
...	...	...	...	...	...	...	...	...	...	
18625	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18626	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18627	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18628	2021	Export	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18629	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	

18630 rows × 20 columns

In [93]:
```
df.columns
```

Out[93]:
```
Index(['Year', 'Trade Flow', 'Reporter', 'Reporter ISO', 'Partner',
       'Partner ISO', '2nd Partner', '2nd Partner ISO', 'Customs Proc. Co
de',
       'Customs', 'Mode of Transport', '2-Digit', 'Commodity', 'Alt Qty U
nit',
       'Alt Qty', 'Netweight (kg)', 'Trade Value (US$)',
       'CIF Trade Value (US$)', 'FOB Trade Value (US$)', 'Flag'],
      dtype='object')
```

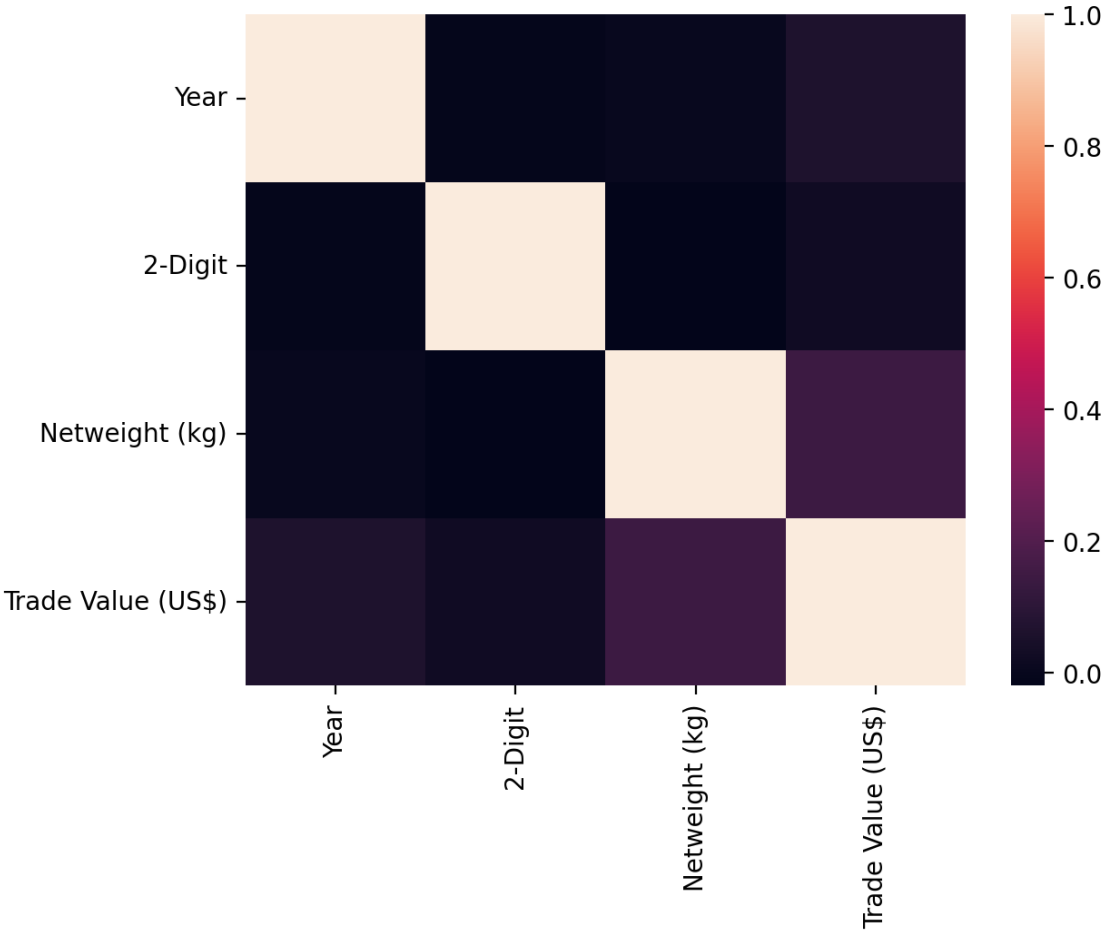
In [95]:
```
# Linear (Pearson) correlation
df_corr3 = df.corr(method = "pearson").round(decimals = 2)
df_corr3
# Instantiating a heatmap
sns.heatmap(df_corr3)

# Displaying the plot
plt.show()
```



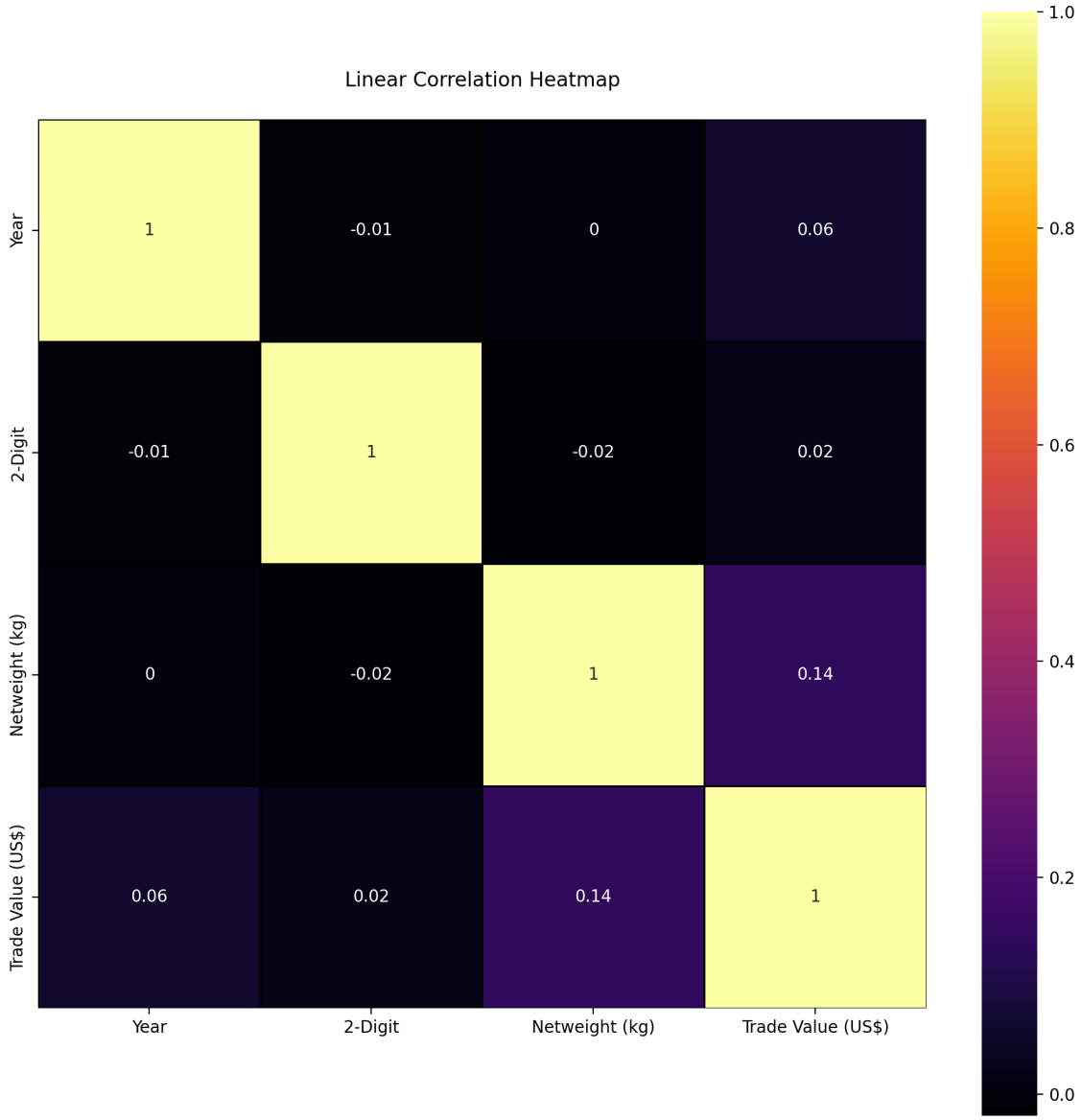
In [97]:
```
df.drop(['Flag', 'Alt Qty','CIF Trade Value (US$)', 'FOB Trade Value (US$)'
```

In [98]:

```
# Linear (Pearson) correlation
df_corr4 = df.corr(method = "pearson").round(decimals = 2)
df_corr4
# Instantiating a heatmap
sns.heatmap(df_corr4)

# Displaying the plot
plt.show()

# Specifying plot size (making it bigger)
fig, ax = plt.subplots(figsize=(12,12))


# Developing a spicy heatmap
sns.heatmap(data       = df_corr4, # the correlation matrix
            cmap       = 'inferno',     # changing to SPICY colors
            square     = True,          # tightening the layout
            annot      = True,          # should there be numbers in the he
            linecolor  = 'black',       # lines between boxes
            linewidths = 0.5)          # how thick should the lines be?


# Title and displaying the plot
plt.title("""
Linear Correlation Heatmap
""")
```



Out[98]: Text(0.5, 1.0, '\nLinear Correlation Heatmap\n')



In [100]:
```
df
```

Out[100]:

	Year	Trade Flow	Reporter	Reporter ISO	Partner	Partner ISO	2nd Partner	2nd Partner ISO	Customs Proc. Code	Custo
0	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
1	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
2	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
3	2010	Import	Cambodia	KHM	World	W00	World	W00	C00	TO
4	2010	Export	Cambodia	KHM	World	W00	World	W00	C00	TO
...	...	...	...	...	...	...	...	...	...	
18625	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18626	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18627	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18628	2021	Export	Cambodia	KHM	World	WLD	NaN	NaN	NaN	
18629	2021	Import	Cambodia	KHM	World	WLD	NaN	NaN	NaN	

18630 rows × 16 columns

In [101]:
```
df.drop(['Partner', 'Partner ISO', '2nd Partner'], axis=1, inplace=True)
```

In [102]:
```
df.head()
```

Out[102]:

	Year	Trade Flow	Reporter	Reporter ISO	2nd Partner ISO	Customs Proc. Code	Customs	Mode of Transport	2-Digit	Commod
0	2010	Import	Cambodia	KHM	W00	C00	TOTAL CPC	TOTAL MOT	1	Swine;
1	2010	Import	Cambodia	KHM	W00	C00	TOTAL CPC	TOTAL MOT	1	Sheep a goats;
2	2010	Import	Cambodia	KHM	W00	C00	TOTAL CPC	TOTAL MOT	1	Poultry; l fowls of spec Ga dor
3	2010	Import	Cambodia	KHM	W00	C00	TOTAL CPC	TOTAL MOT	1	Anim n.e.c chapter
4	2010	Export	Cambodia	KHM	W00	C00	TOTAL CPC	TOTAL MOT	1	Anim n.e.c chapter

In [103]:
```
df.drop(['Reporter ISO', '2nd Partner ISO', 'Customs Proc. Code'], axis=1,
df.head()
```

Out[103]:

	Year	Trade Flow	Reporter	Customs	Mode of Transport	2-Digit	Commodity	Alt Qty Unit	Netweight (kg)	Trade Value (US$)
0	2010	Import	Cambodia	TOTAL CPC	TOTAL MOT	1	Swine; live	NaN	3580060.0	193665(
1	2010	Import	Cambodia	TOTAL CPC	TOTAL MOT	1	Sheep and goats; live	NaN	30.0	142
2	2010	Import	Cambodia	TOTAL CPC	TOTAL MOT	1	Poultry; live, fowls of the species Gallus dom...	NaN	2168.0	12484(
3	2010	Import	Cambodia	TOTAL CPC	TOTAL MOT	1	Animals, n.e.c. in chapter 01; live	NaN	268.0	7831
4	2010	Export	Cambodia	TOTAL CPC	TOTAL MOT	1	Animals, n.e.c. in chapter 01; live	NaN	30196.0	97937

In [7]:
```
df.drop(['Mode of Transport', 'Alt Qty Unit'], axis=1, inplace=True)
df.head()
```

Out[7]:

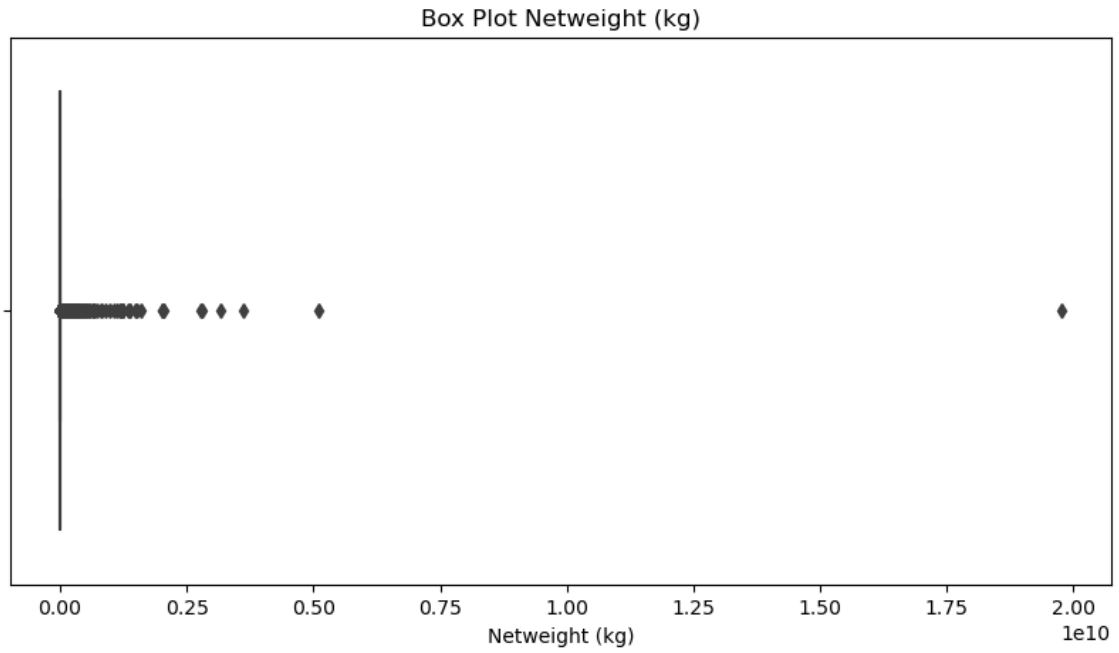
	Classification	Year	Period	Period Desc.	Aggregate Level	Is Leaf Code	Trade Flow Code	Trade Flow	Reporter Code	Reporter
0	H3	2010	2010	2010	4	0	0	Import	116	Cambodia
1	H3	2010	2010	2010	4	0	0	Import	116	Cambodia
2	H3	2010	2010	2010	4	0	0	Import	116	Cambodia
3	H3	2010	2010	2010	4	0	0	Import	116	Cambodia
4	H3	2010	2010	2010	4	0	0	Export	116	Cambodia

5 rows × 35 columns

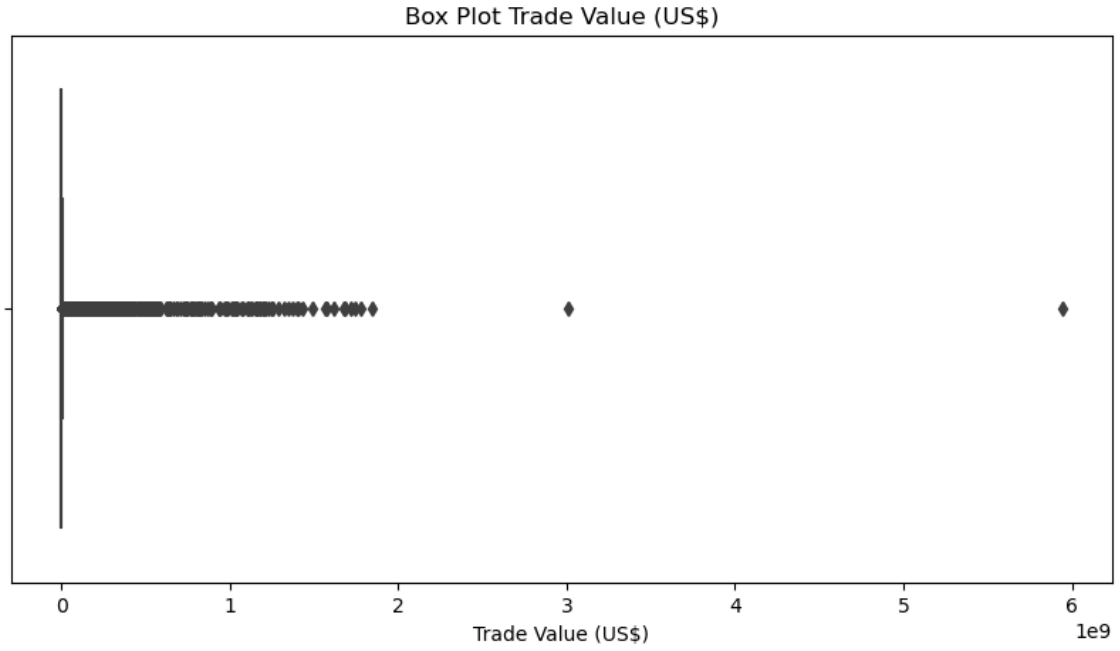
In [12]:
```
df['Netweight (kg)'].describe()
```

Out[12]:
```
count    1.765200e+04
mean     8.394154e+06
std      1.687688e+08
min      0.000000e+00
25%      5.180750e+03
50%      8.152800e+04
75%      1.054520e+06
max      1.975356e+10
Name: Netweight (kg), dtype: float64
```

In [9]:
```
fig, ax = plt.subplots(figsize= (10,5))
ax.set_title("Box Plot Netweight (kg)")
sns.boxplot(x='Netweight (kg)',data = df);
```



In [8]:
```
fig, ax = plt.subplots(figsize= (10,5))
ax.set_title("Box Plot Trade Value (US$)")
sns.boxplot(x='Trade Value (US$)',data = df);
```



In [39]:
```
df_com = df['Netweight (kg)'].groupby(df['2-Digit']&df['Year']).sum()
df_com
```

Out[39]:
```
0     2.158951e+10
1     1.354939e+10
2     4.843203e+09
3     1.008371e+10
4     3.093057e+09
          ...
95    5.891005e+06
96    1.316583e+08
97    2.661970e+05
98    0.000000e+00
99    0.000000e+00
Name: Netweight (kg), Length: 74, dtype: float64
```

In [15]:
```
#Import and Export Classification
#Import of each commodity from 2010 - 2021
#Export of each commodity from 2010 - 2021
```

In [38]:
```
df_com
```

Out[38]:

	count	mean	std	min	25%	50%	75%	ma
0	1325.0	1.629397e+07	1.554897e+08	0.0	9039.00	127829.0	1306184.00	3.601432e+(
1	474.0	2.858522e+07	2.104567e+08	0.0	5397.00	104888.0	1463658.75	3.171442e+(
2	419.0	1.155896e+07	8.608702e+07	1.0	7411.00	100800.0	915863.00	1.494454e+(
3	199.0	5.067193e+07	4.118275e+08	1.0	21865.50	200000.0	2174931.00	5.092310e+(
4	539.0	5.738510e+06	3.017474e+07	0.0	8662.00	135526.0	1179786.50	3.627603e+(
...	...	...	...	...	...	...	...	
95	12.0	4.909171e+05	1.025192e+06	3292.0	23073.25	54033.5	649257.00	3.623424e+(
96	185.0	7.116665e+05	2.217390e+06	0.0	1205.00	14299.0	217890.00	1.518041e+(
97	17.0	1.565865e+04	3.176334e+04	0.0	0.00	786.0	8829.00	9.736700e+(
98	3.0	0.000000e+00	0.000000e+00	0.0	0.00	0.0	0.00	0.000000e+(
99	2.0	0.000000e+00	0.000000e+00	0.0	0.00	0.0	0.00	0.000000e+(

74 rows × 8 columns

In [ ]:
```
```