Import libraries

In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
data = pd.read_csv(r"C:\Users\user\Downloads\6_Salesworkload1.csv")
data
```

Out[2]:

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLea
0	10.2016	1.0	United Kingdom	88253.0	London (I)	1.0	Dry	3184.764	
1	10.2016	1.0	United Kingdom	88253.0	London (I)	2.0	Frozen	1582.941	
2	10.2016	1.0	United Kingdom	88253.0	London (I)	3.0	other	47.205	
3	10.2016	1.0	United Kingdom	88253.0	London (I)	4.0	Fish	1623.852	
4	10.2016	1.0	United Kingdom	88253.0	London (I)	5.0	Fruits & Vegetables	1759.173	
...	...	...	...	...	...	...	...	...	
7653	06.2017	9.0	Sweden	29650.0	Gothenburg	12.0	Checkout	6322.323	
7654	06.2017	9.0	Sweden	29650.0	Gothenburg	16.0	Customer Services	4270.479	
7655	06.2017	9.0	Sweden	29650.0	Gothenburg	11.0	Delivery	0	
7656	06.2017	9.0	Sweden	29650.0	Gothenburg	17.0	others	2224.929	
7657	06.2017	9.0	Sweden	29650.0	Gothenburg	18.0	all	39652.2	

7658 rows × 14 columns

In [3]:
```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7658 entries, 0 to 7657
Data columns (total 14 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   MonthYear      7658 non-null   object 
 1   Time index     7650 non-null   float64
 2   Country        7650 non-null   object 
 3   StoreID        7650 non-null   float64
 4   City           7650 non-null   object 
 5   Dept_ID        7650 non-null   float64
 6   Dept. Name     7650 non-null   object 
 7   HoursOwn       7650 non-null   object 
 8   HoursLease     7650 non-null   float64
 9   Sales units    7650 non-null   float64
 10  Turnover       7650 non-null   float64
 11  Customer       0 non-null      float64
 12  Area (m2)      7650 non-null   object 
 13  Opening hours  7650 non-null   object 
dtypes: float64(7), object(7)
memory usage: 837.7+ KB
None
```

In [4]:
```
print(data.describe())
```

```
        Time index       StoreID      Dept_ID   HoursLease   Sales units  \
count  7650.000000   7650.000000  7650.000000  7650.000000  7.650000e+03   
mean      5.000000  61995.220000     9.470588    22.036078  1.076471e+06   
std       2.582158  29924.581631     5.337429   133.299513  1.728113e+06   
min       1.000000  12227.000000     1.000000     0.000000  0.000000e+00   
25%       3.000000  29650.000000     5.000000     0.000000  5.457125e+04   
50%       5.000000  75400.500000     9.000000     0.000000  2.932300e+05   
75%       7.000000  87703.000000    14.000000     0.000000  9.175075e+05   
max       9.000000  98422.000000    18.000000  3984.000000  1.124296e+07   

           Turnover  Customer  
count  7.650000e+03       0.0  
mean   3.721393e+06       NaN  
std    6.003380e+06       NaN  
min    0.000000e+00       NaN  
25%    2.726798e+05       NaN  
50%    9.319575e+05       NaN  
75%    3.264432e+06       NaN  
max    4.271739e+07       NaN  
```

In [5]:
```
data.size
```

Out[5]: 107212

In [6]:
```
data.shape
```

Out[6]: (7658, 14)

In [7]:
```
data1 = data.drop(["Country","City","Dept. Name","Customer","Opening hours"],a
data1
```

Out[7]:

	MonthYear	Time index	StoreID	Dept_ID	HoursOwn	HoursLease	Sales units	Turnover	Are (m
0	10.2016	1.0	88253.0	1.0	3184.764	0.0	398560.0	1226244.0	953.(
1	10.2016	1.0	88253.0	2.0	1582.941	0.0	82725.0	387810.0	720.4
2	10.2016	1.0	88253.0	3.0	47.205	0.0	438400.0	654657.0	966.7
3	10.2016	1.0	88253.0	4.0	1623.852	0.0	309425.0	499434.0	1053.3
4	10.2016	1.0	88253.0	5.0	1759.173	0.0	165515.0	329397.0	1053.3
...	...	...	...	...	...	...	...	...	
7653	06.2017	9.0	29650.0	12.0	6322.323	0.0	3886530.0	14538825.0	#N
7654	06.2017	9.0	29650.0	16.0	4270.479	0.0	245.0	0.0	#N
7655	06.2017	9.0	29650.0	11.0	0	0.0	0.0	0.0	#N
7656	06.2017	9.0	29650.0	17.0	2224.929	0.0	245.0	0.0	#N
7657	06.2017	9.0	29650.0	18.0	39652.2	0.0	3886530.0	15056214.0	#N

7658 rows × 9 columns

In [8]:
```
print(data1.mean())
```

```
Time index     5.000000e+00
StoreID        6.199522e+04
Dept_ID        9.470588e+00
HoursLease     2.203608e+01
Sales units    1.076471e+06
Turnover       3.721393e+06
dtype: float64
```

In [9]:
```
print(data.median())
```

```
Time index          5.0
StoreID         75400.5
Dept_ID             9.0
HoursLease          0.0
Sales units    293230.0
Turnover       931957.5
Customer            NaN
dtype: float64
```

In [12]:
```
print(data1.mode())
```

```
     Happiness Rank  Happiness Score  Standard Error  \
0              82.0            5.192         0.03751
1               NaN              NaN         0.03780
2               NaN              NaN         0.04394
3               NaN              NaN         0.04934
4               NaN              NaN         0.05051
..              ...              ...             ...
153             NaN              NaN             NaN
154             NaN              NaN             NaN
155             NaN              NaN             NaN
156             NaN              NaN             NaN
157             NaN              NaN             NaN

     Economy (GDP per Capita)   Family  Health (Life Expectancy)  Freedom  \
0                     0.00000  0.00000                   0.92356  0.00000
1                     0.01530  0.13995                       NaN  0.07699
2                     0.01604  0.30285                       NaN  0.09245
3                     0.06940  0.35386                       NaN  0.10081
4                     0.07120  0.38174                       NaN  0.10384
..                        ...      ...                       ...      ...
153                   1.45900  1.34043                       NaN  0.65821
154                   1.52186  1.34951                       NaN  0.65980
155                   1.55422  1.36058                       NaN  0.66246
156                   1.56391  1.36948                       NaN  0.66557
157                   1.69042  1.40223                       NaN  0.66973

     Trust (Government Corruption)  Generosity  Dystopia Residual
0                          0.32524     0.00000            0.32858
1                              NaN     0.00199            0.65429
2                              NaN     0.02641            0.67042
3                              NaN     0.05444            0.67108
4                              NaN     0.05547            0.89991
..                             ...         ...                ...
153                            NaN     0.51535            3.10712
154                            NaN     0.51752            3.17728
155                            NaN     0.51912            3.19131
156                            NaN     0.57630            3.26001
157                            NaN     0.79588            3.60214

[158 rows x 10 columns]
```

In [10]:
```
print(data1.isna().sum())
```

```
MonthYear      0
Time index     8
StoreID        8
Dept_ID        8
HoursOwn       8
HoursLease     8
Sales units    8
Turnover       8
Area (m2)      8
dtype: int64
```

In [12]:
```
data1.fillna(value="50")
```

Out[12]:

	MonthYear	Time index	StoreID	Dept_ID	HoursOwn	HoursLease	Sales units	Turnover	Are (m
0	10.2016	1.0	88253.0	1.0	3184.764	0.0	398560.0	1226244.0	953.(
1	10.2016	1.0	88253.0	2.0	1582.941	0.0	82725.0	387810.0	720.4
2	10.2016	1.0	88253.0	3.0	47.205	0.0	438400.0	654657.0	966.7
3	10.2016	1.0	88253.0	4.0	1623.852	0.0	309425.0	499434.0	1053.3
4	10.2016	1.0	88253.0	5.0	1759.173	0.0	165515.0	329397.0	1053.3
...	...	...	...	...	...	...	...	...	
7653	06.2017	9.0	29650.0	12.0	6322.323	0.0	3886530.0	14538825.0	#N
7654	06.2017	9.0	29650.0	16.0	4270.479	0.0	245.0	0.0	#N
7655	06.2017	9.0	29650.0	11.0	0	0.0	0.0	0.0	#N
7656	06.2017	9.0	29650.0	17.0	2224.929	0.0	245.0	0.0	#N
7657	06.2017	9.0	29650.0	18.0	39652.2	0.0	3886530.0	15056214.0	#N

7658 rows × 9 columns

In [13]:
```
print(data1.isna().sum())
```

```
MonthYear      0
Time index     8
StoreID        8
Dept_ID        8
HoursOwn       8
HoursLease     8
Sales units    8
Turnover       8
Area (m2)      8
dtype: int64
```

In [14]:
```
print(data1.sum())
```

```
MonthYear      10.201610.201610.201610.201610.201610.201610.2...
Time index                                              38250.0
StoreID                                            474263433.0
Dept_ID                                                72450.0
HoursLease                                            168576.0
Sales units                                        8235000965.0
Turnover                                          28468656015.0
dtype: object
```

In [15]:
```
data2 = data1[["MonthYear","Sales units"]]
data2
```

Out[15]:

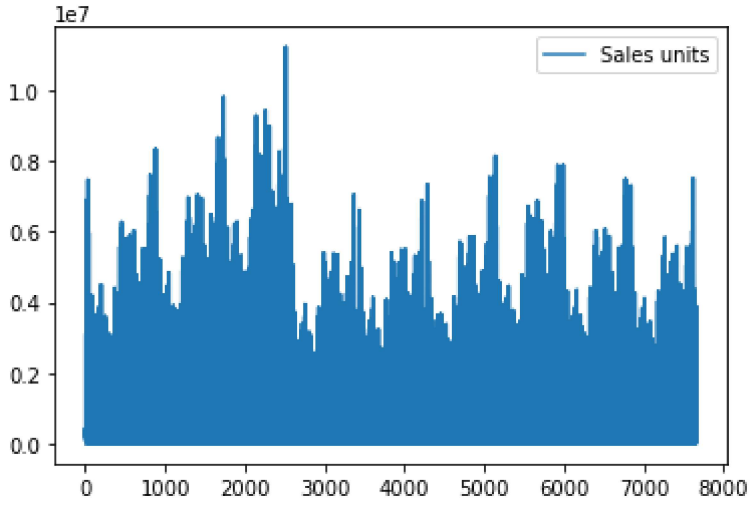
	MonthYear	Sales units
0	10.2016	398560.0
1	10.2016	82725.0
2	10.2016	438400.0
3	10.2016	309425.0
4	10.2016	165515.0
...	...	...
7653	06.2017	3886530.0
7654	06.2017	245.0
7655	06.2017	0.0
7656	06.2017	245.0
7657	06.2017	3886530.0

7658 rows × 2 columns

In [16]:
```
import matplotlib.pyplot as plot
```

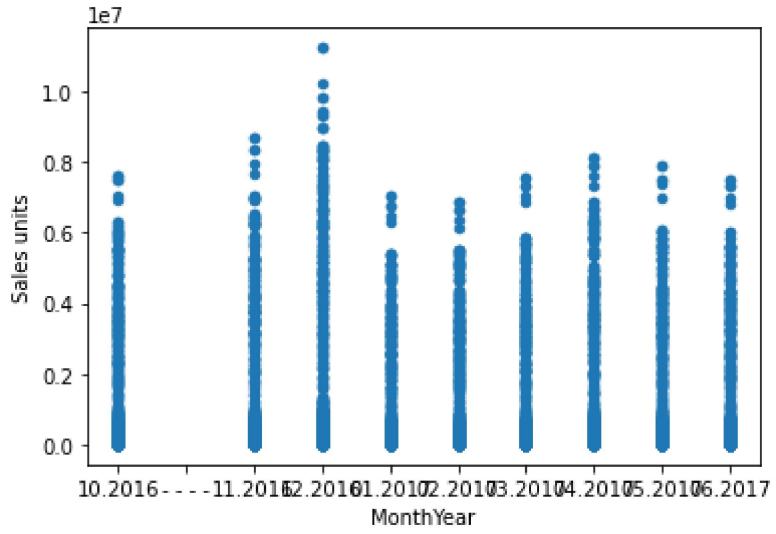
In [17]:
```
data2.plot.line()
```

Out[17]: <AxesSubplot:>



In [18]:
```
data2.plot.scatter("MonthYear","Sales units")
```

Out[18]:
```
<AxesSubplot:xlabel='MonthYear', ylabel='Sales units'>
```

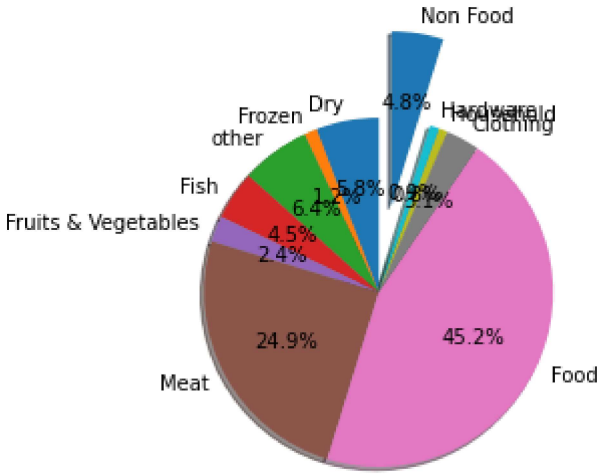


In [21]:
```
data3 = data[["Dept. Name","Sales units"]][0:11]
data3
```

Out[21]:

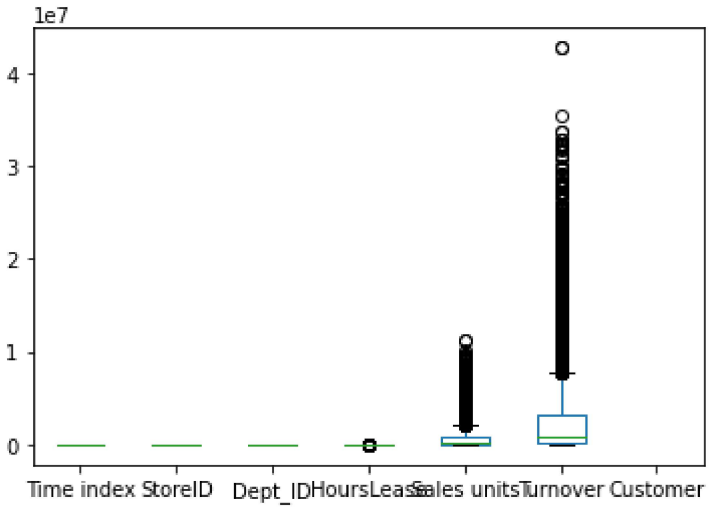
	Dept. Name	Sales units
0	Dry	398560.0
1	Frozen	82725.0
2	other	438400.0
3	Fish	309425.0
4	Fruits & Vegetables	165515.0
5	Meat	1713310.0
6	Food	3107935.0
7	Clothing	213680.0
8	Household	54915.0
9	Hardware	59260.0
10	Non Food	327855.0

In [28]:
```
slice = data3["Sales units"]
country = data3["Dept. Name"]
#col = ["r","g","b","orange","y","black","white","violet","brown"]
plot.pie(slice,labels=country,startangle=90,shadow=True,explode=(0,0,0,0,0,0,0,
plot.show()
```



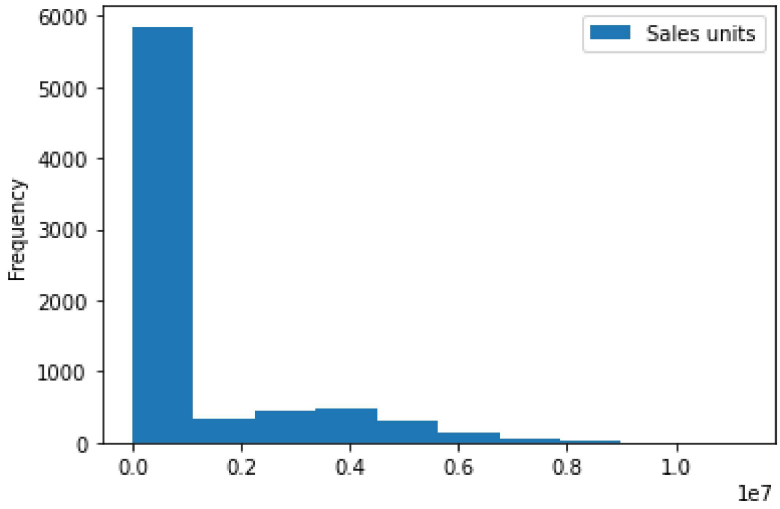
In [29]:
```
data.plot.box()
```

Out[29]: <AxesSubplot:>



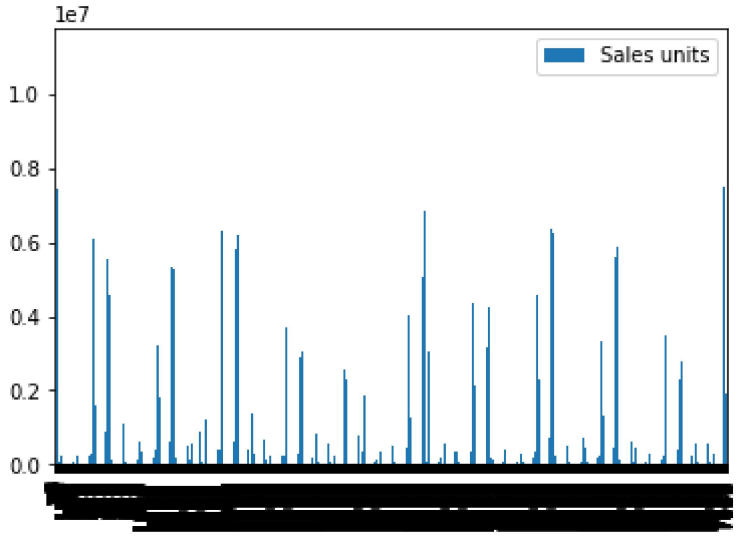
In [30]:
```
data2.plot.hist()
```

Out[30]: <AxesSubplot:ylabel='Frequency'>



In [31]:
```
data2.plot.bar()
```

Out[31]: <AxesSubplot:>



In [ ]: