In [2]:

```
#PROBLEM STATEMENTS
#•  What is the Highest and Lowest Sales units in the given dataset?
#•  What is the Highest and Lowest Turnover in the given dataset?
#•  What is the correlation between Turnover and sales unit?
#•  Remove empty column in the given data set.

import numpy as np
import pandas as pd
```

In [3]:
```
df=pd.read_csv(r"C:\Users\Jayadeep\Downloads\Salesworkload1 (1).csv")
df
```

Out[3]:

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLease	Sales units	Turnover	Customer	Area (m2)
0	10.2016	1.0	United Kingdom	88253.0	London (I)	1.0	Dry	3184.764	0.0	398560.0	1226244.0	NaN	953.04
1	10.2016	1.0	United Kingdom	88253.0	London (I)	2.0	Frozen	1582.941	0.0	82725.0	387810.0	NaN	720.48
2	10.2016	1.0	United Kingdom	88253.0	London (I)	3.0	other	47.205	0.0	438400.0	654657.0	NaN	966.72
3	10.2016	1.0	United Kingdom	88253.0	London (I)	4.0	Fish	1623.852	0.0	309425.0	499434.0	NaN	1053.36
4	10.2016	1.0	United Kingdom	88253.0	London (I)	5.0	Fruits & Vegetables	1759.173	0.0	165515.0	329397.0	NaN	1053.36
...	...	...	...	...	...	...	...	...	...	...	...	...	...
7653	06.2017	9.0	Sweden	29650.0	Gothenburg	12.0	Checkout	6322.323	0.0	3886530.0	14538825.0	NaN	#NV
7654	06.2017	9.0	Sweden	29650.0	Gothenburg	16.0	Customer Services	4270.479	0.0	245.0	0.0	NaN	#NV
7655	06.2017	9.0	Sweden	29650.0	Gothenburg	11.0	Delivery	0	0.0	0.0	0.0	NaN	#NV
7656	06.2017	9.0	Sweden	29650.0	Gothenburg	17.0	others	2224.929	0.0	245.0	0.0	NaN	#NV
7657	06.2017	9.0	Sweden	29650.0	Gothenburg	18.0	all	39652.2	0.0	3886530.0	15056214.0	NaN	#NV

7658 rows × 14 columns

In [7]:

```
df.head(20)
```

Out[7]:

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLease	Sales units	Turnover	Customer	Area (m2)	O
0	10.2016	1.0	United Kingdom	88253.0	London (I)	1.0	Dry	3184.764	0.0	398560.0	1226244.0	NaN	953.04	
1	10.2016	1.0	United Kingdom	88253.0	London (I)	2.0	Frozen	1582.941	0.0	82725.0	387810.0	NaN	720.48	
2	10.2016	1.0	United Kingdom	88253.0	London (I)	3.0	other	47.205	0.0	438400.0	654657.0	NaN	966.72	
3	10.2016	1.0	United Kingdom	88253.0	London (I)	4.0	Fish	1623.852	0.0	309425.0	499434.0	NaN	1053.36	
4	10.2016	1.0	United Kingdom	88253.0	London (I)	5.0	Fruits & Vegetables	1759.173	0.0	165515.0	329397.0	NaN	1053.36	
5	10.2016	1.0	United Kingdom	88253.0	London (I)	6.0	Meat	8270.316	0.0	1713310.0	5617137.0	NaN	11735.16	
6	10.2016	1.0	United Kingdom	88253.0	London (I)	13.0	Food	16468.251	0.0	3107935.0	8714679.0	NaN	19865.64	
7	10.2016	1.0	United Kingdom	88253.0	London (I)	7.0	Clothing	4698.471	0.0	213680.0	1615341.0	NaN	8513.52	
8	10.2016	1.0	United Kingdom	88253.0	London (I)	8.0	Household	1183.272	0.0	54915.0	290400.0	NaN	4842.72	
9	10.2016	1.0	United Kingdom	88253.0	London (I)	9.0	Hardware	2029.815	0.0	59260.0	450015.0	NaN	5608.8	
10	10.2016	1.0	United Kingdom	88253.0	London (I)	14.0	Non Food	7911.558	0.0	327855.0	2355756.0	NaN	19238.64	
11	10.2016	1.0	United Kingdom	88253.0	London (I)	15.0	Admin	4308.243	0.0	0.0	0.0	NaN	0	
12	10.2016	1.0	United Kingdom	88253.0	London (I)	12.0	Checkout	5825.097	0.0	3435790.0	11070435.0	NaN	39104.28	
13	10.2016	1.0	United Kingdom	88253.0	London (I)	16.0	Customer Services	3320.085	0.0	0.0	0.0	NaN	0	
14	10.2016	1.0	United Kingdom	88253.0	London (I)	11.0	Delivery	0	0.0	0.0	0.0	NaN	0	
15	10.2016	1.0	United Kingdom	88253.0	London (I)	17.0	others	2253.252	0.0	0.0	0.0	NaN	0	

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLease	Sales units	Turnover	Customer	Area (m2)	O
16	10.2016	1.0	United Kingdom	88253.0	London (I)	18.0	all	40086.486	0.0	3435790.0	11070435.0	NaN	39104.28	
17	10.2016	1.0	United Kingdom	38976.0	Manchester	1.0	Dry	2583.687	0.0	754600.0	2648175.0	NaN	1404.48	
18	10.2016	1.0	United Kingdom	38976.0	Manchester	2.0	Frozen	5145.345	0.0	216925.0	1291830.0	NaN	1057.92	
19	10.2016	1.0	United Kingdom	38976.0	Manchester	3.0	other	47.205	0.0	1111835.0	1822749.0	NaN	1235.76	

```
In [8]: df.tail(20)
```

Out[8]:

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLease	Sales units	Turnover	Customer	Area (m2)	Ope h
7638	06.2017	9.0	Sweden	90992.0	Malmö	11.0	Delivery	0	0.0	0.0	0.0	NaN	#NV	Ty
7639	06.2017	9.0	Sweden	90992.0	Malmö	17.0	others	1935.405	0.0	240.0	0.0	NaN	#NV	Ty
7640	06.2017	9.0	Sweden	90992.0	Malmö	18.0	all	40133.691	0.0	4363130.0	18666582.0	NaN	#NV	Ty
7641	06.2017	9.0	Sweden	29650.0	Gothenburg	1.0	Dry	1929.111	0.0	454145.0	1815402.0	NaN	#NV	Ty
7642	06.2017	9.0	Sweden	29650.0	Gothenburg	2.0	Frozen	2457.807	0.0	79500.0	466167.0	NaN	#NV	Ty
7643	06.2017	9.0	Sweden	29650.0	Gothenburg	3.0	other	47.205	0.0	511770.0	692709.0	NaN	#NV	Ty
7644	06.2017	9.0	Sweden	29650.0	Gothenburg	4.0	Fish	1689.939	0.0	363010.0	632217.0	NaN	#NV	Ty
7645	06.2017	9.0	Sweden	29650.0	Gothenburg	5.0	Fruits & Vegetables	2567.952	0.0	270165.0	514305.0	NaN	#NV	Ty
7646	06.2017	9.0	Sweden	29650.0	Gothenburg	6.0	Meat	8210.523	0.0	1895090.0	7750254.0	NaN	#NV	Ty
7647	06.2017	9.0	Sweden	29650.0	Gothenburg	13.0	Food	16902.537	0.0	3573680.0	12223593.0	NaN	#NV	Ty
7648	06.2017	9.0	Sweden	29650.0	Gothenburg	7.0	Clothing	3587.58	0.0	183700.0	1693797.0	NaN	#NV	Ty
7649	06.2017	9.0	Sweden	29650.0	Gothenburg	8.0	Household	1312.299	0.0	58045.0	575979.0	NaN	#NV	Ty
7650	06.2017	9.0	Sweden	29650.0	Gothenburg	9.0	Hardware	1598.676	0.0	71105.0	582849.0	NaN	#NV	Ty
7651	06.2017	9.0	Sweden	29650.0	Gothenburg	14.0	Non Food	6498.555	0.0	312850.0	2855895.0	NaN	#NV	Ty
7652	06.2017	9.0	Sweden	29650.0	Gothenburg	15.0	Admin	3433.377	0.0	245.0	0.0	NaN	#NV	Ty
7653	06.2017	9.0	Sweden	29650.0	Gothenburg	12.0	Checkout	6322.323	0.0	3886530.0	14538825.0	NaN	#NV	Ty
7654	06.2017	9.0	Sweden	29650.0	Gothenburg	16.0	Customer Services	4270.479	0.0	245.0	0.0	NaN	#NV	Ty
7655	06.2017	9.0	Sweden	29650.0	Gothenburg	11.0	Delivery	0	0.0	0.0	0.0	NaN	#NV	Ty
7656	06.2017	9.0	Sweden	29650.0	Gothenburg	17.0	others	2224.929	0.0	245.0	0.0	NaN	#NV	Ty
7657	06.2017	9.0	Sweden	29650.0	Gothenburg	18.0	all	39652.2	0.0	3886530.0	15056214.0	NaN	#NV	Ty

In [9]:
```
df.shape
```

Out[9]:
```
(7658, 14)
```

In [10]:

```
df.describe
```

Out[10]:

```
<bound method NDFrame.describe of      MonthYear  Time index         Country  StoreID         City  Dept_ID  \
0      10.2016         1.0  United Kingdom  88253.0  London (I)      1.0
1      10.2016         1.0  United Kingdom  88253.0  London (I)      2.0
2      10.2016         1.0  United Kingdom  88253.0  London (I)      3.0
3      10.2016         1.0  United Kingdom  88253.0  London (I)      4.0
4      10.2016         1.0  United Kingdom  88253.0  London (I)      5.0
...        ...         ...             ...      ...         ...      ...
7653   06.2017         9.0          Sweden  29650.0  Gothenburg     12.0
7654   06.2017         9.0          Sweden  29650.0  Gothenburg     16.0
7655   06.2017         9.0          Sweden  29650.0  Gothenburg     11.0
7656   06.2017         9.0          Sweden  29650.0  Gothenburg     17.0
7657   06.2017         9.0          Sweden  29650.0  Gothenburg     18.0

               Dept. Name  HoursOwn  HoursLease  Sales units    Turnover  \
0                     Dry  3184.764         0.0     398560.0   1226244.0
1                  Frozen  1582.941         0.0      82725.0    387810.0
2                   other    47.205         0.0     438400.0    654657.0
3                    Fish  1623.852         0.0     309425.0    499434.0
4     Fruits & Vegetables  1759.173         0.0     165515.0    329397.0
...                   ...       ...         ...          ...         ...
7653             Checkout  6322.323         0.0    3886530.0  14538825.0
7654    Customer Services  4270.479         0.0        245.0         0.0
7655             Delivery         0         0.0          0.0         0.0
7656               others  2224.929         0.0        245.0         0.0
7657                  all   39652.2         0.0    3886530.0  15056214.0

      Customer Area (m2) Opening hours
0          NaN    953.04        Type A
1          NaN    720.48        Type A
2          NaN    966.72        Type A
3          NaN   1053.36        Type A
4          NaN   1053.36        Type A
...        ...       ...           ...
7653       NaN       #NV        Type A
7654       NaN       #NV        Type A
7655       NaN       #NV        Type A
7656       NaN       #NV        Type A
7657       NaN       #NV        Type A

[7658 rows x 14 columns]>
```

In [11]:
```
df.isna().any()
```

Out[11]:
```
MonthYear        False
Time index        True
Country           True
StoreID           True
City              True
Dept_ID           True
Dept. Name        True
HoursOwn          True
HoursLease        True
Sales units       True
Turnover          True
Customer          True
Area (m2)         True
Opening hours     True
dtype: bool
```

In [12]:
```
df.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
```

In [13]:
```
print(df.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
Customer                                                     0.0
dtype: object

C:\Users\Jayadeep\AppData\Local\Temp\ipykernel_11736\760981365.py:1: FutureWarning: Dropping of nuisance columns in
DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  print(df.sum())
```

In [15]:
```
print(df.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
Customer                NaN
dtype: float64

C:\Users\Jayadeep\AppData\Local\Temp\ipykernel_11736\2807316344.py:1: FutureWarning: Dropping of nuisance columns in
DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  print(df.mean())
```

In [16]:
```
print(df.max())
```

```
MonthYear         12.2016
Time index            9.0
StoreID           98422.0
Dept_ID              18.0
HoursLease         3984.0
Sales units    11242955.0
Turnover       42717390.0
Customer              NaN
dtype: object

C:\Users\Jayadeep\AppData\Local\Temp\ipykernel_11736\2376050674.py:1: FutureWarning: Dropping of nuisance columns in
DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  print(df.max())
```

In [ ]:
```
#the highest sales units 11242955.0
#the highest  turnover rate 42717390.0
```

In [17]:

```
print(df.min())
```

```
MonthYear       - - - -
Time index          1.0
StoreID         12227.0
Dept_ID             1.0
HoursLease          0.0
Sales units         0.0
Turnover            0.0
Customer            NaN
dtype: object
```

```
C:\Users\Jayadeep\AppData\Local\Temp\ipykernel_11736\3863265735.py:1: FutureWarning: Dropping of nuisance columns in
DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError.  Selec
t only valid columns before calling the reduction.
  print(df.min())
```

In [ ]:

```
#the lowest sales unit 0.0
# the lowest turnover 0.0
```

In [18]:
```
print(df.mode())
```

```
   MonthYear  Time index         Country  StoreID              City  Dept_ID  \
0    01.2017         1.0          France  12227.0      Aalborg (I)      1.0
1    02.2017         2.0         Germany  15552.0     Aalborg (II)      2.0
2    03.2017         3.0  United Kingdom  16927.0        Amsterdam      3.0
3    04.2017         4.0             NaN  17647.0          Antwerp      4.0
4    05.2017         5.0             NaN  18808.0    Barcelona (I)      5.0
5    06.2017         6.0             NaN  19000.0   Barcelona (II)      6.0
6    10.2016         7.0             NaN  19340.0       Berlin (I)      7.0
7    11.2016         8.0             NaN  19769.0      Berlin (II)      8.0
8    12.2016         9.0             NaN  20166.0           Bilbao      9.0
9        NaN         NaN             NaN  20891.0       Birmingham     11.0
10       NaN         NaN             NaN  22117.0          Bologna     12.0
11       NaN         NaN             NaN  23623.0         Bordeaux     13.0
12       NaN         NaN             NaN  29650.0             Brno     14.0
13       NaN         NaN             NaN  32949.0    Brussels (I)     15.0
14       NaN         NaN             NaN  34378.0   Brussels (II)     16.0
15       NaN         NaN             NaN  38560.0          Cologne     17.0
16       NaN         NaN             NaN  38976.0  Copenhagen (I)     18.0
17       NaN         NaN             NaN  42367.0 Copenhagen (II)      NaN
```

In [19]:
```
df[['Turnover','Sales units']]
```

Out[19]:

	Turnover	Sales units
0	1226244.0	398560.0
1	387810.0	82725.0
2	654657.0	438400.0
3	499434.0	309425.0
4	329397.0	165515.0
...	...	...
7653	14538825.0	3886530.0
7654	0.0	245.0
7655	0.0	0.0
7656	0.0	245.0
7657	15056214.0	3886530.0

7658 rows × 2 columns

In [20]:
```
df2=df[['Turnover','Sales units']]
print(df2.corr())
```

```
             Turnover  Sales units
Turnover     1.000000     0.947374
Sales units  0.947374     1.000000
```

In [ ]:
```
#The correlation relation between Turnover and sales unit is 1.0  0.9
                                                             0.0  1.0
```

In [16]:
```
del df['Customer']
```

In [17]:

```
print(df)
```

```
     MonthYear  Time index         Country  StoreID        City  Dept_ID  \
0      10.2016         1.0  United Kingdom  88253.0  London (I)      1.0   
1      10.2016         1.0  United Kingdom  88253.0  London (I)      2.0   
2      10.2016         1.0  United Kingdom  88253.0  London (I)      3.0   
3      10.2016         1.0  United Kingdom  88253.0  London (I)      4.0   
4      10.2016         1.0  United Kingdom  88253.0  London (I)      5.0   
...        ...         ...             ...      ...         ...      ...   
7653   06.2017         9.0          Sweden  29650.0  Gothenburg     12.0   
7654   06.2017         9.0          Sweden  29650.0  Gothenburg     16.0   
7655   06.2017         9.0          Sweden  29650.0  Gothenburg     11.0   
7656   06.2017         9.0          Sweden  29650.0  Gothenburg     17.0   
7657   06.2017         9.0          Sweden  29650.0  Gothenburg     18.0   

               Dept. Name  HoursOwn  HoursLease  Sales units    Turnover  \
0                     Dry  3184.764         0.0     398560.0   1226244.0   
1                  Frozen  1582.941         0.0      82725.0    387810.0   
2                   other    47.205         0.0     438400.0    654657.0   
3                    Fish  1623.852         0.0     309425.0    499434.0   
4     Fruits & Vegetables  1759.173         0.0     165515.0    329397.0   
...                   ...       ...         ...          ...         ...   
7653             Checkout  6322.323         0.0    3886530.0  14538825.0   
7654    Customer Services  4270.479         0.0        245.0         0.0   
7655             Delivery         0         0.0          0.0         0.0   
7656               others  2224.929         0.0        245.0         0.0   
7657                  all   39652.2         0.0    3886530.0  15056214.0   

     Area (m2) Opening hours  
0       953.04        Type A  
1       720.48        Type A  
2       966.72        Type A  
3      1053.36        Type A  
4      1053.36        Type A  
...        ...           ...  
7653       #NV        Type A  
7654       #NV        Type A  
7655       #NV        Type A  
7656       #NV        Type A  
7657       #NV        Type A  

[7658 rows x 13 columns]
```

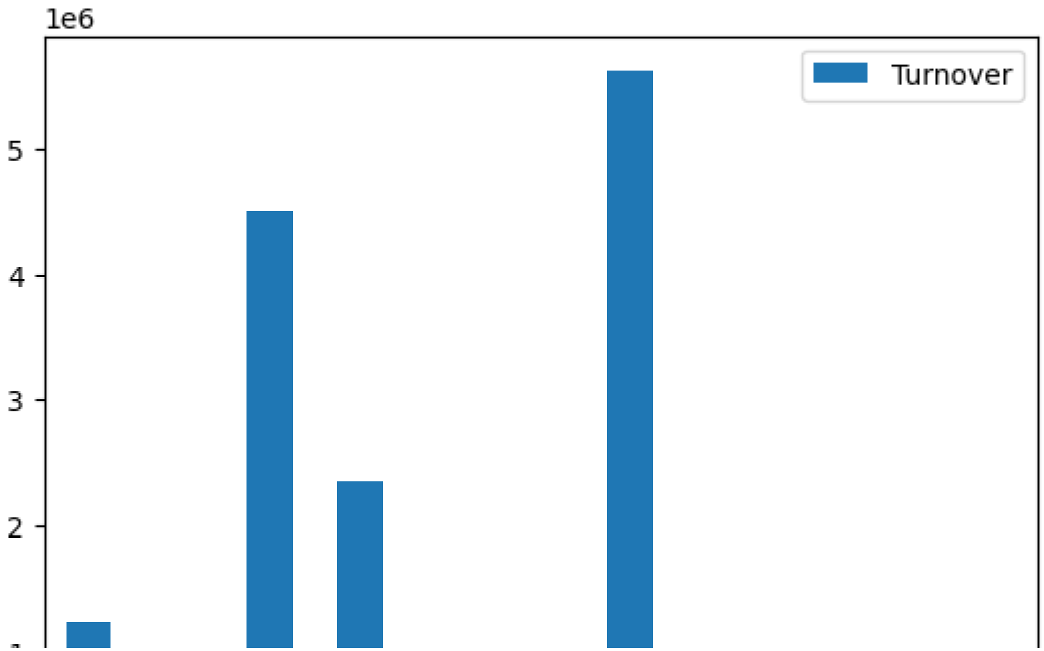
```
In [ ]: #Deleting empty column in the data set
```

```
In [18]: df.shape
```

```
Out[18]: (7658, 13)
```

```
In [6]: import pandas as pd
        import matplotlib.pyplot as plt
```

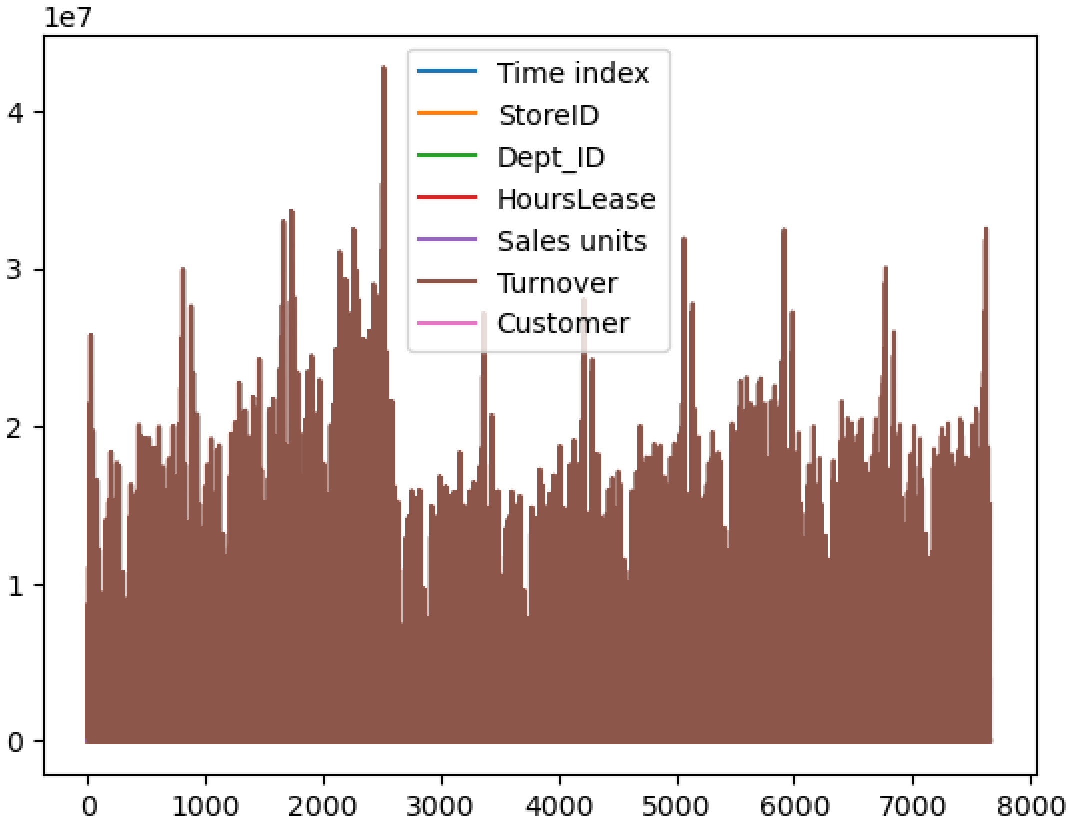
```
In [7]: df=pd.DataFrame({"Countries":['united kingdom','poland','the nether lands','czech republic','den mark','spain','italy'
                          "Turnover":[1226244,499439,4500150,2355756,329397,654657,5617137,290400,0.0,0.0,34523]})
        df.plot(x="Countries",y="Turnover",kind="bar")
        plt.show()
```



```
In [4]: df.plot()
        plt.show()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_16612\2546976976.py in <module>
      1 df.plot()
----> 2 plt.show()

NameError: name 'plt' is not defined
```



```
Answers:
1.The highest sales unit in the given data set is 11242955.0,and the lowest sales unit is 0.0
```

2.The highest turnover rate 42717390.0 and the lowest sales unit is 0.0
3.The correlation relation between Turnover and sales unit is

```
             Turnover    Sales units
Turnover     1.000000     0.947374
Sales units  0.947374     1.000000
```