

In [8]: %matplotlib inline

```
from numpy import arange
from matplotlib import pyplot as plt
from scipy.stats import norm
import pandas as pd
from pandas_profiling import ProfileReport

plt.rcParams['figure.figsize'] = [16, 7]
```

In [10]: columns = ['Date of trading', 'Price from the first transaction of a trading day', '']

df = pd.read\_csv('Daily-DOGE-USD.csv')
df.columns = columns
df.head()

Out[10]:

	Date of trading	Price from the first transaction of a trading day	Maximum price in a trading day	Minimum price in a trading day	Price from the last transaction of a trading day	Closing price adjusted to reflect the value after accounting for any corporate actions	Number of units traded in a day
0	2017-08-24	0.001731	0.001845	0.001722	0.001816	0.001816	4796720.0
1	2017-08-25	0.001808	0.001830	0.001762	0.001790	0.001790	2654040.0
2	2017-08-26	0.001787	0.001797	0.001735	0.001778	0.001778	1935350.0
3	2017-08-27	0.001779	0.001784	0.001745	0.001760	0.001760	2241620.0
4	2017-08-28	0.001764	0.001825	0.001751	0.001825	0.001825	3210440.0

In [5]: df.dtypes

Out[5]: Date of trading  
object  
Price from the first transaction of a trading day  
float64  
Maximum price in a trading day  
float64  
Minimum price in a trading day  
float64  
Price from the last transaction of a trading day  
float64  
Closing price adjusted to reflect the value after accounting for any corporate actions  
float64  
Number of units traded in a day  
float64  
dtype: object

In [6]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1462 entries, 0 to 1461
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype  
 ---  --          --          --      
 0   Date of trading    1462 non-null   object 

```

```

1462 non-null    object
1   Price from the first transaction of a trading day
1458 non-null    float64
2   Maximum price in a trading day
1458 non-null    float64
3   Minimum price in a trading day
1458 non-null    float64
4   Price from the last transaction of a trading day
1458 non-null    float64
5   Closing price adjusted to reflect the value after accounting for any corporate
actions 1458 non-null    float64
6   Number of units traded in a day
1458 non-null    float64
dtypes: float64(6), object(1)
memory usage: 80.1+ KB

```

In [7]: `df.memory_usage()`

Out[7]:

```

Index
128
Date of trading
11696
Price from the first transaction of a trading day
11696
Maximum price in a trading day
11696
Minimum price in a trading day
11696
Price from the last transaction of a trading day
11696
Closing price adjusted to reflect the value after accounting for any corporate actions
11696
Number of units traded in a day
11696
dtype: int64

```

In [8]: `df.memory_usage().sum()`

Out[8]: 82000

In [9]: `df.describe()`

Out[9]:

	Price from the first transaction of a trading day	Maximum price in a trading day	Minimum price in a trading day	Price from the last transaction of a trading day	Closing price adjusted to reflect the value after accounting for any corporate actions	Number of units traded in a day
<b>count</b>	1458.000000	1458.000000	1458.000000	1458.000000	1458.000000	1.458000e+03
<b>mean</b>	0.033272	0.036116	0.030603	0.033487	0.033487	9.026038e+08
<b>std</b>	0.092222	0.101371	0.083382	0.092597	0.092597	3.842058e+09
<b>min</b>	0.000762	0.000783	0.000626	0.000755	0.000755	1.072410e+06
<b>25%</b>	0.002373	0.002441	0.002312	0.002374	0.002374	1.672411e+07
<b>50%</b>	0.002827	0.002929	0.002769	0.002836	0.002836	4.818314e+07
<b>75%</b>	0.005095	0.005403	0.004744	0.005131	0.005131	1.610538e+08
<b>max</b>	0.687801	0.737567	0.608168	0.684777	0.684777	6.941068e+10

In [10]: `df.mean()`

Out[10]: Price from the first transaction of a trading day

```
3.327249e-02
Maximum price in a trading day
3.611581e-02
Minimum price in a trading day
3.060283e-02
Price from the last transaction of a trading day
3.348666e-02
Closing price adjusted to reflect the value after accounting for any corporate actions 3.348666e-02
Number of units traded in a day
9.026038e+08
dtype: float64
```

```
In [12]: df['Price from the first transaction of a trading day'].mean()
```

```
Out[12]: 0.033272485596707826
```

```
In [13]: df['Price from the last transaction of a trading day'].mean()
```

```
Out[13]: 0.03348666255144034
```

```
In [14]: df.var()
```

```
Price from the first transaction of a trading day
8.504959e-03
Maximum price in a trading day
1.027603e-02
Minimum price in a trading day
6.952502e-03
Price from the last transaction of a trading day
8.574280e-03
Closing price adjusted to reflect the value after accounting for any corporate actions 8.574280e-03
Number of units traded in a day
1.476141e+19
dtype: float64
```

```
In [15]: df.skew()
```

```
Price from the first transaction of a trading day
3.533527
Maximum price in a trading day
3.653203
Minimum price in a trading day
3.387153
Price from the last transaction of a trading day
3.515325
Closing price adjusted to reflect the value after accounting for any corporate actions 3.515325
Number of units traded in a day
9.037273
dtype: float64
```

```
In [16]: df.kurtosis()
```

```
Price from the first transaction of a trading day
13.062617
Maximum price in a trading day
14.324251
Minimum price in a trading day
11.537820
Price from the last transaction of a trading day
12.907410
Closing price adjusted to reflect the value after accounting for any corporate actions 12.907410
Number of units traded in a day
```

```
110.795222
dtype: float64
```

In [17]: `df.min()`

```
Date of trading
2017-08-24
Price from the first transaction of a trading day
0.000762
Maximum price in a trading day
0.000783
Minimum price in a trading day
0.000626
Price from the last transaction of a trading day
0.000755
Closing price adjusted to reflect the value after accounting for any corporate actions
0.000755
Number of units traded in a day
1.07241e+06
dtype: object
```

In [18]: `df.max()`

```
Date of trading
2021-08-24
Price from the first transaction of a trading day
0.687801
Maximum price in a trading day
0.737567
Minimum price in a trading day
0.608168
Price from the last transaction of a trading day
0.684777
Closing price adjusted to reflect the value after accounting for any corporate actions
0.684777
Number of units traded in a day
6.94107e+10
dtype: object
```

In [19]: `df.median()`

```
Price from the first transaction of a trading day
2.827500e-03
Maximum price in a trading day
2.929000e-03
Minimum price in a trading day
2.769000e-03
Price from the last transaction of a trading day
2.836000e-03
Closing price adjusted to reflect the value after accounting for any corporate actions
2.836000e-03
Number of units traded in a day
4.818314e+07
dtype: float64
```

In [20]: `df.corr()`

Out[20]:

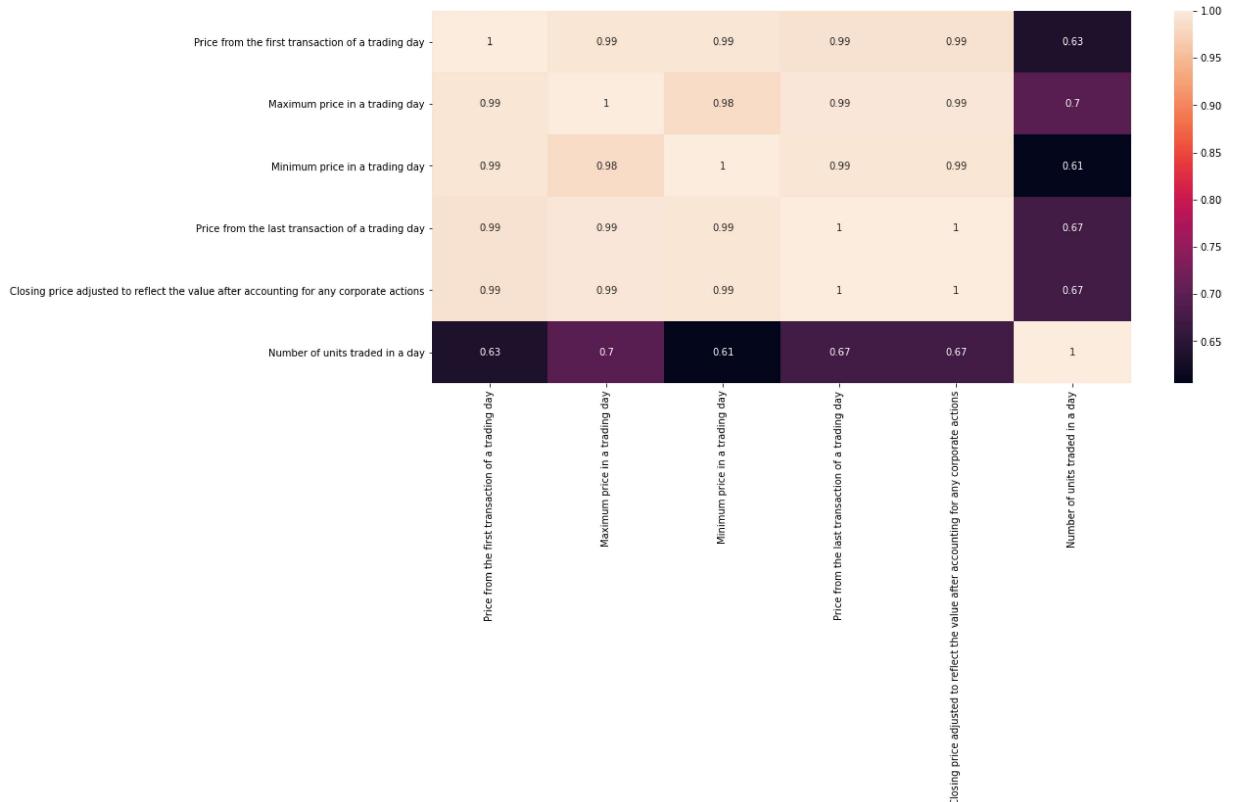
Price from the first transaction of a trading day	Maximum price in a trading day	Minimum price in a trading day	Price from the last transaction of a trading day	Closing price adjusted to reflect the value after accounting for any corporate actions	Number of units traded in a day
Price from the first transaction of a trading day	1.000000	0.992809	0.992453	0.990302	0.634959

	Price from the first transaction of a trading day	Maximum price in a trading day	Minimum price in a trading day	Price from the last transaction of a trading day	Closing price adjusted to reflect the value after accounting for any corporate actions	Number of units traded in a day
<b>Maximum price in a trading day</b>	0.992809	1.000000	0.984556	0.993993	0.993993	0.696660
<b>Minimum price in a trading day</b>	0.992453	0.984556	1.000000	0.993659	0.993659	0.605290
<b>Price from the last transaction of a trading day</b>	0.990302	0.993993	0.993659	1.000000	1.000000	0.671712
<b>Closing price adjusted to reflect the value after accounting for any corporate actions</b>	0.990302	0.993993	0.993659	1.000000	1.000000	0.671712
<b>Number of units traded in a day</b>	0.634959	0.696660	0.605290	0.671712	0.671712	1.000000

In [21]: `import seaborn as sns`

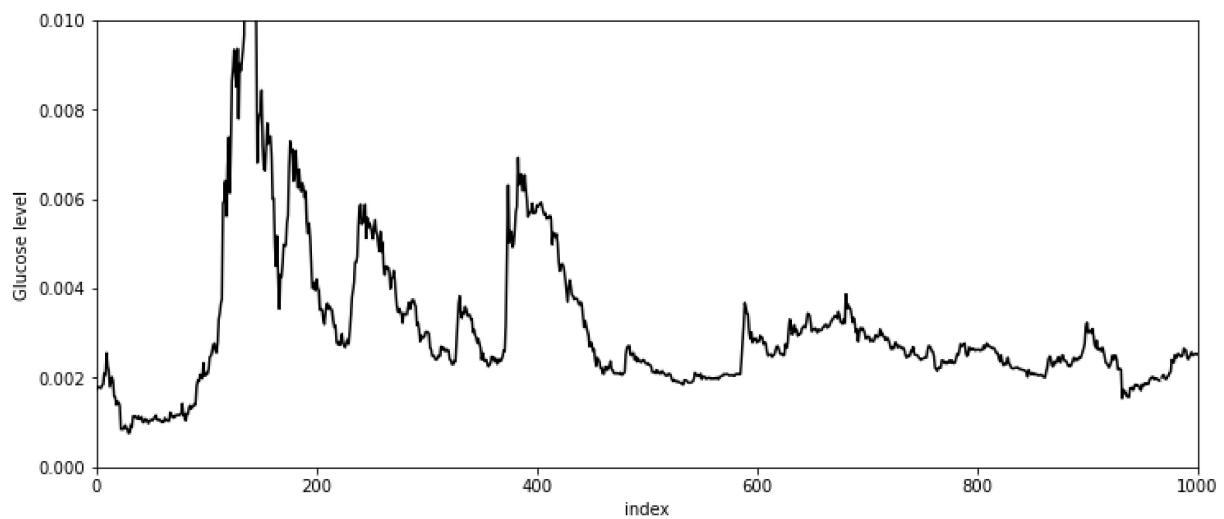
In [22]: `sns.heatmap(df.corr(), annot=True)`

Out[22]: <AxesSubplot:>



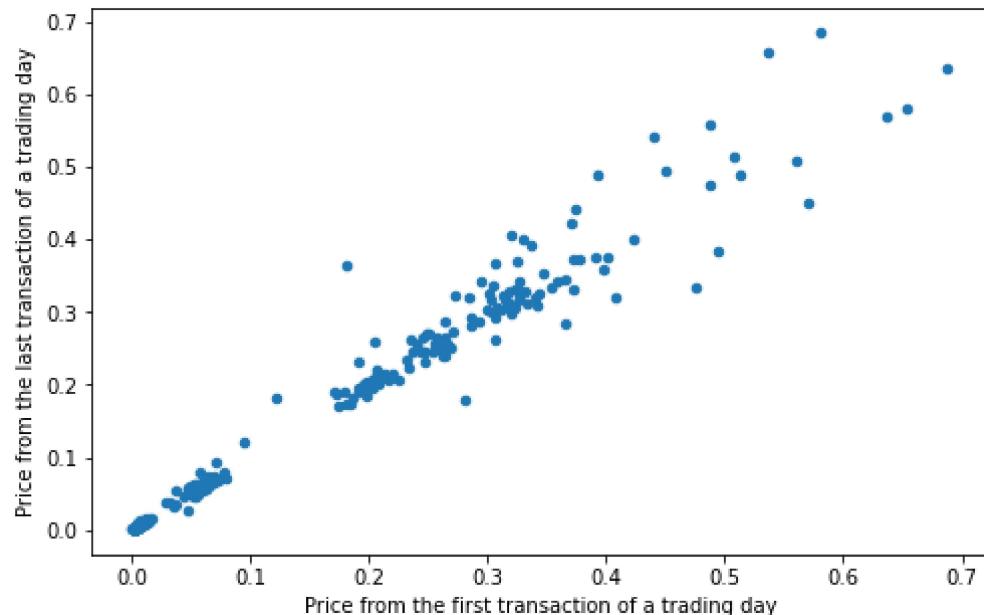
In [53]: `# Plotting with index along the x-axis`  
`df['Price from the first transaction of a trading day'].plot(figsize=(12, 5), color=`  
`plt.xlim(0,1000) # range for x-axis`  
`plt.ylim(0, 0.01) # range for y-axis`

```
plt.xlabel('index')
plt.ylabel('Price from the first transaction of a trading day'); # ";" prevents obje
```

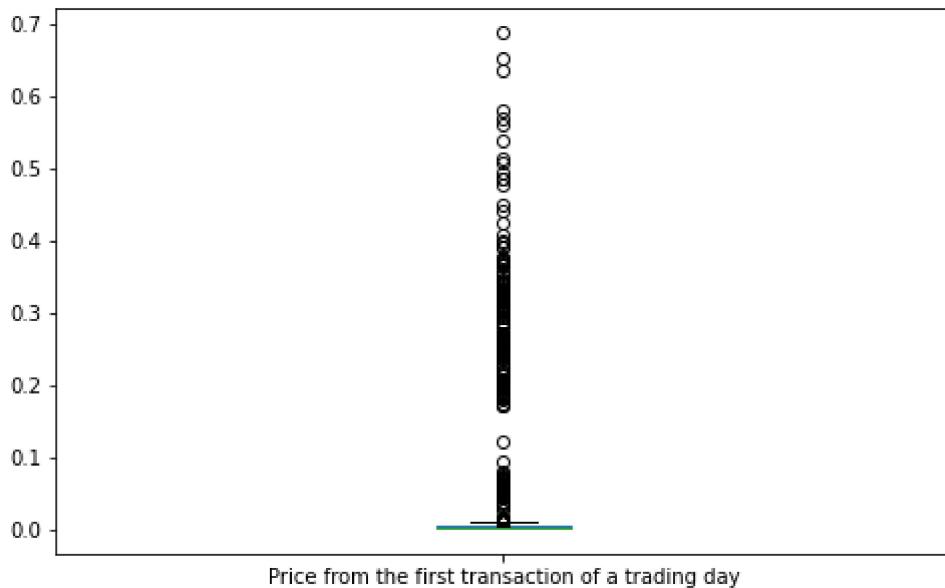


```
In [38]: df.plot.scatter('Price from the first transaction of a trading day', 'Price from the last transaction of a trading day')
```

```
Out[38]: <AxesSubplot:xlabel='Price from the first transaction of a trading day', ylabel='Price from the last transaction of a trading day'>
```

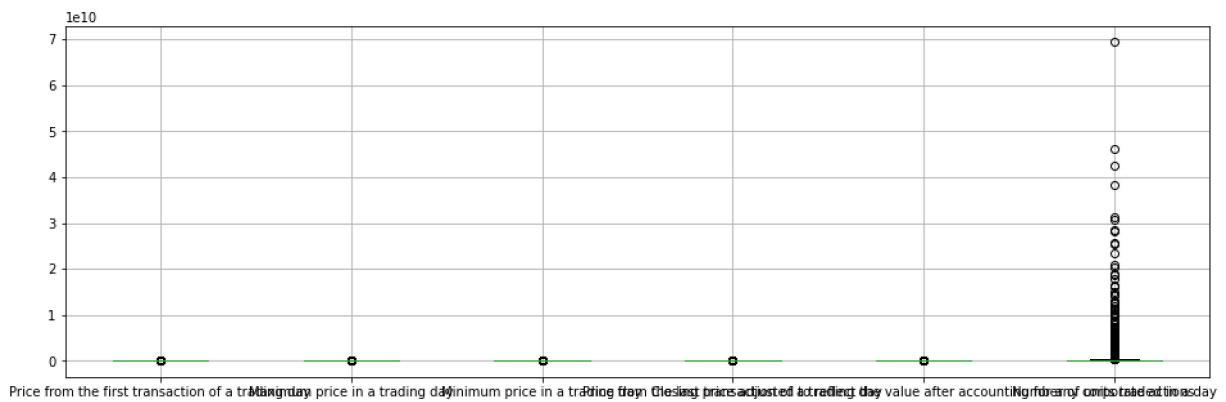


```
In [54]: df['Price from the first transaction of a trading day'].plot.box(figsize=(8, 5));
```

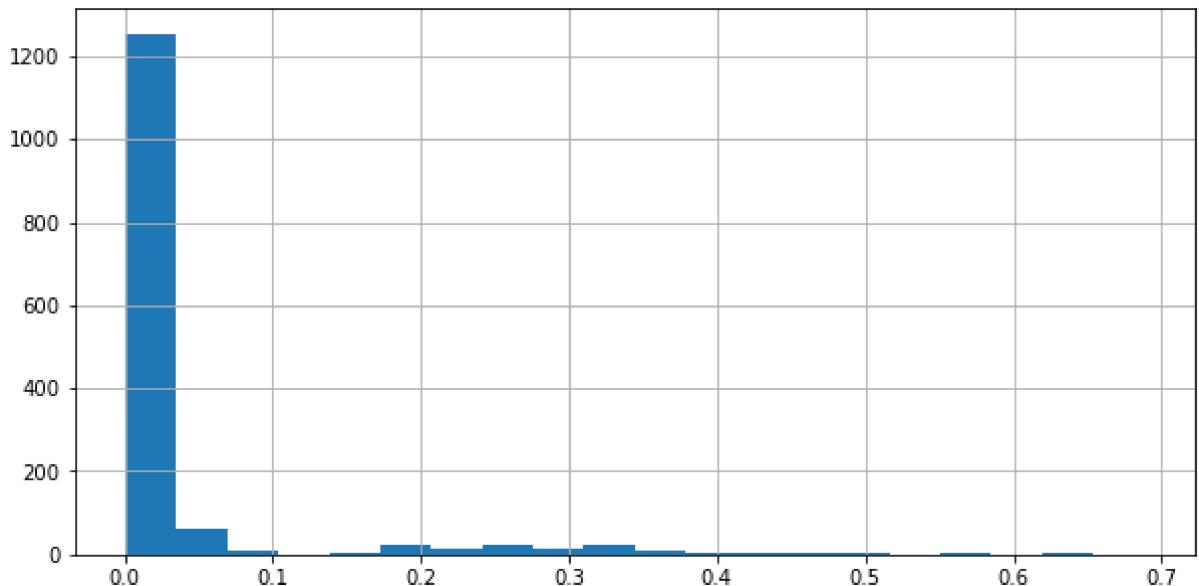


```
In [82]: df.boxplot(figsize=(16, 5)) # or df.plot.box()
```

```
Out[82]: <AxesSubplot:>
```

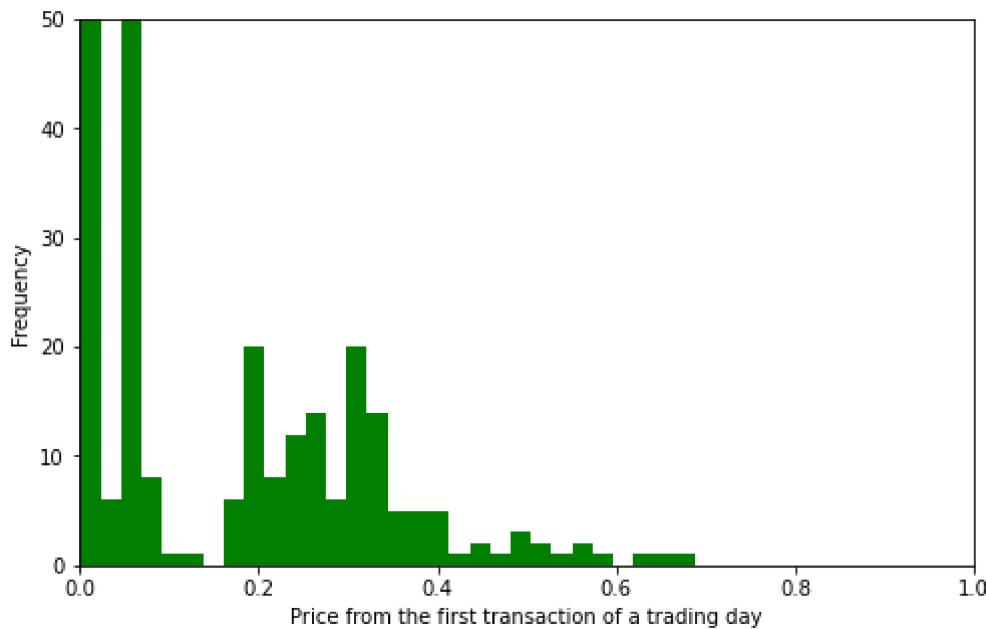


```
In [65]: df['Price from the first transaction of a trading day'].hist(bins=20, figsize=(10, 5))
```

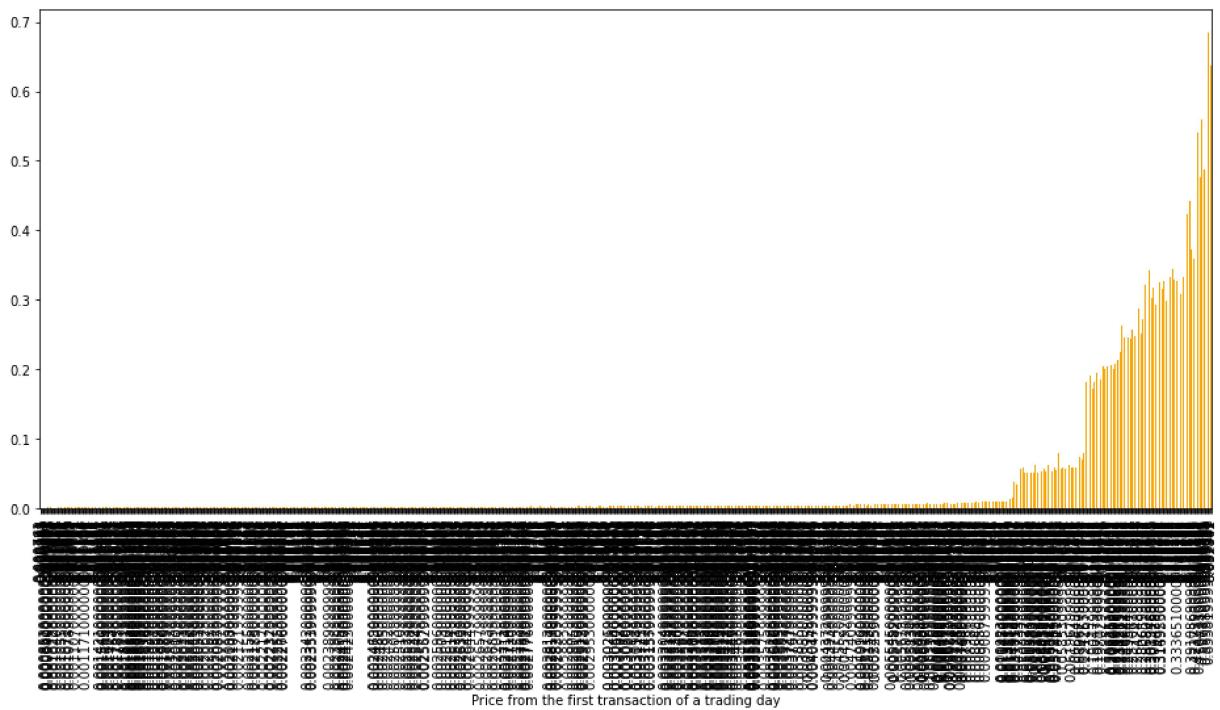


```
In [70]: ax = df['Price from the first transaction of a trading day'].hist(bins=30, grid=False)
ax.set_xlabel('Price from the first transaction of a trading day')
ax.set_ylabel('Frequency')
```

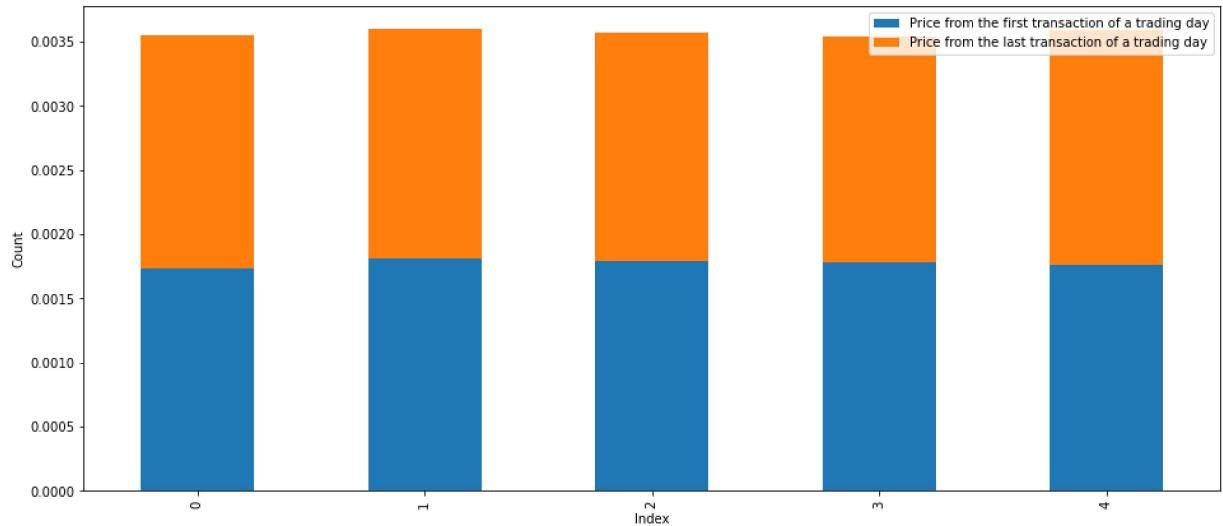
```
ax.set_xlim(0, 1) # Limiting display range to 0-70 for the x-axis  
ax.set_ylim(0, 50); # Limiting display range to 0-120 for the y-axis
```



```
In [79]: df_avg_BP = df.groupby('Price from the first transaction of a trading day')['Price f  
df_avg_BP[:5].plot.bar(color='orange');
```



```
In [72]: ax = df[['Price from the first transaction of a trading day', 'Price from the last t  
ax.set_xlabel("Index")  
ax.set_ylabel("Count");
```



```
In [78]: fig, axes = plt.subplots(2, 2, figsize=(12, 8))

df['Price from the first transaction of a trading day'].plot(ax = axes[0][0], style='k')
df['Price from the last transaction of a trading day'].plot(ax = axes[0][1], style='k')

df['Maximum price in a trading day'].plot.hist(bins=30, ax = axes[1][0], color='black')
df['Minimum price in a trading day'].plot.hist(bins=20, ax = axes[1][1], color='gray')

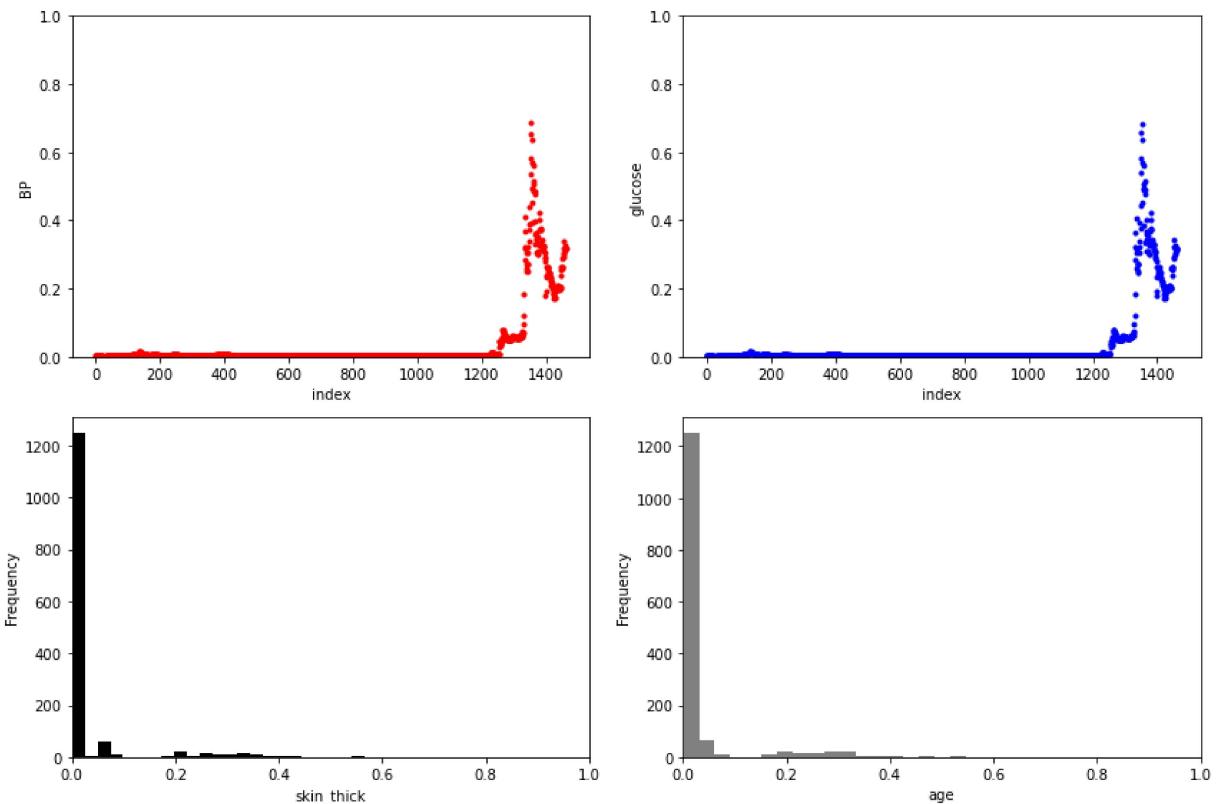
axes[0][0].set_xlabel('index')
axes[0][1].set_xlabel('index')
axes[1][0].set_xlabel('Maximum price in a trading day')
axes[1][1].set_xlabel('Minimum price in a trading day')

axes[0][0].set_ylabel('Price from the first transaction of a trading day')
axes[0][1].set_ylabel('Price from the last transaction of a trading day')

axes[0][0].set_ylim(0, 1)
axes[0][1].set_ylim(0, 1)

axes[1][0].set_xlim(0, 1)
axes[1][1].set_xlim(0, 1)

fig.tight_layout()
```



```
In [11]: profile = ProfileReport(df,title='Pandas profiling Report',explorative=True)
profile.to_widgets()
```

```
In [ ]:
```