

In [1]:

```
pip install wordcloud
```

Requirement already satisfied: wordcloud in c:\users\pmyls\anaconda3\lib\site-packages (1.9.3)  
Requirement already satisfied: numpy>=1.6.1 in c:\users\pmyls\anaconda3\lib\site-packages (from wordcloud) (1.21.5)  
Requirement already satisfied: matplotlib in c:\users\pmyls\anaconda3\lib\site-packages (from wordcloud) (3.5.2)  
Requirement already satisfied: pillow in c:\users\pmyls\anaconda3\lib\site-packages (from wordcloud) (9.2.0)  
Requirement already satisfied: cycler>=0.10 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (0.11.0)  
Requirement already satisfied: fonttools>=4.22.0 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (4.25.0)  
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (1.4.2)  
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (3.0.9)  
Requirement already satisfied: packaging>=20.0 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (21.3)  
Requirement already satisfied: python-dateutil>=2.7 in c:\users\pmyls\anaconda3\lib\site-packages (from matplotlib->wordcloud) (2.8.2)  
Requirement already satisfied: six>=1.5 in c:\users\pmyls\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib->wordcloud) (1.16.0)  
Note: you may need to restart the kernel to use updated packages.

In [2]:

```
import pandas as pd  
import plotly.express as px  
import plotly.graph_objects as go  
import plotly.io as pio  
from wordcloud import WordCloud  
pio.templates.default = "plotly_white"
```

In [3]:

```
inst = pd.read_csv('Instagram_data.csv',encoding='latin-1')  
inst.head()
```

Out[3]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows	Caption	Hashtags
0	3920	2586	1028	619	56	98	9	5	162	35	2	Here are some of the most important data visua...	#finance #money #business #investing #investme...
1	5394	2727	1838	1174	78	194	7	14	224	48	10	Here are some of the best data science project...	#healthcare #health #covid #data #datascience ...
2	4021	2085	1188	0	533	41	11	1	131	62	12	Learn how to train a machine learning model an...	#data #datascience #dataanalysis #dataanalytic...
3	4528	2700	621	932	73	172	10	7	213	23	8	HereDs how you can write a Python program to d...	#python #pythonprogramming #pythonprojects #py...
4	2518	1704	255	279	37	96	5	4	123	8	0	Plotting annotations while visualizing your da...	#datavisualization #datascience #data #dataana...

In [4]:

```
# these are the columns names  
inst.columns
```

Out[4]:

```
Index(['Impressions', 'From Home', 'From Hashtags', 'From Explore',  
      'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits',  
      'Follows', 'Caption', 'Hashtags'],  
      dtype='object')
```

In [5]:

```
# info of dataset  
inst.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 119 entries, 0 to 118  
Data columns (total 13 columns):  
#   Column                Non-Null Count  Dtype  
---  ---  
0   Impressions            119 non-null    int64  
1   From Home              119 non-null    int64  
2   From Hashtags          119 non-null    int64  
3   From Explore           119 non-null    int64  
4   From Other             119 non-null    int64  
5   Saves                  119 non-null    int64  
6   Comments               119 non-null    int64  
7   Shares                 119 non-null    int64  
8   Likes                  119 non-null    int64  
9   Profile Visits         119 non-null    int64  
10  Follows                119 non-null    int64  
11  Caption                119 non-null    object  
12  Hashtags               119 non-null    object  
dtypes: int64(11), object(2)  
memory usage: 12.2+ KB
```

In [6]:

```
# applying descriptive statistics  
descriptive_stats = inst.describe()  
descriptive_stats
```

Out[6]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
count	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000
mean	5703.991597	2475.789916	1887.512605	1078.100840	171.092437	153.310924	6.663866	9.361345	173.781513	50.621849	20.756303
std	4843.780105	1489.386348	1884.361443	2613.026132	289.431031	156.317731	3.544576	10.089205	82.378947	87.088402	40.921580
min	1941.000000	1133.000000	116.000000	0.000000	9.000000	22.000000	0.000000	0.000000	72.000000	4.000000	0.000000
25%	3467.000000	1945.000000	726.000000	157.500000	38.000000	65.000000	4.000000	3.000000	121.500000	15.000000	4.000000
50%	4289.000000	2207.000000	1278.000000	326.000000	74.000000	109.000000	6.000000	6.000000	151.000000	23.000000	8.000000
75%	6138.000000	2602.500000	2363.500000	689.500000	196.000000	169.000000	8.000000	13.500000	204.000000	42.000000	18.000000
max	36919.000000	13473.000000	11817.000000	17414.000000	2547.000000	1095.000000	19.000000	75.000000	549.000000	611.000000	260.000000

In [7]:

```
# now I Check is any columns contains any missing values  
  
inst.isnull().sum() # this show their is no null values
```

Out[7]:

```
Impressions      0  
From Home        0  
From Hashtags    0  
From Explore     0  
From Other       0  
Saves            0  
Comments         0  
Shares           0  
Likes            0  
Profile Visits   0  
Follows          0  
Caption          0  
Hashtags         0  
dtype: int64
```

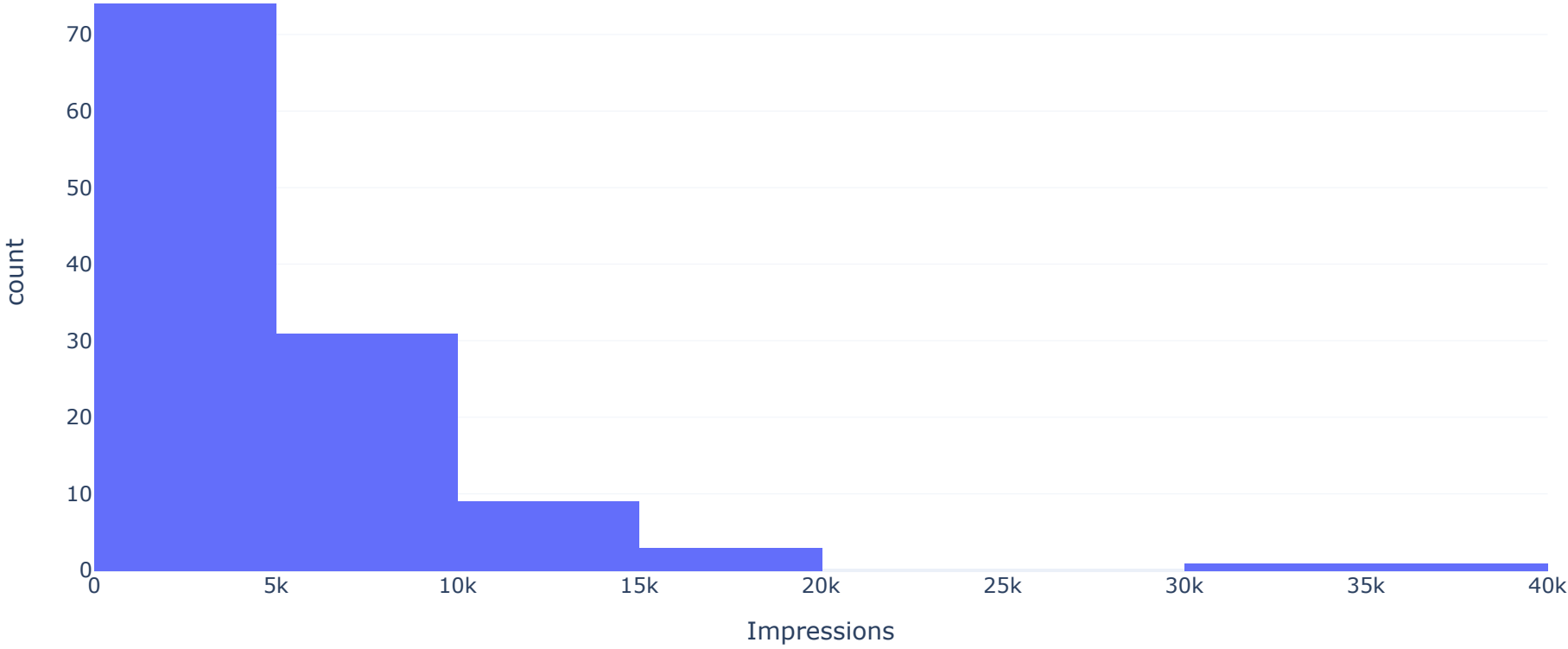
In [8]:

```
# now i will histogram of 'Distribution of Impressions'
```

In [9]:

```
fig = px.histogram(inst,x='Impressions',nbins=8,title='Distribution of Impressions')  
fig
```

Distribution of Impressions



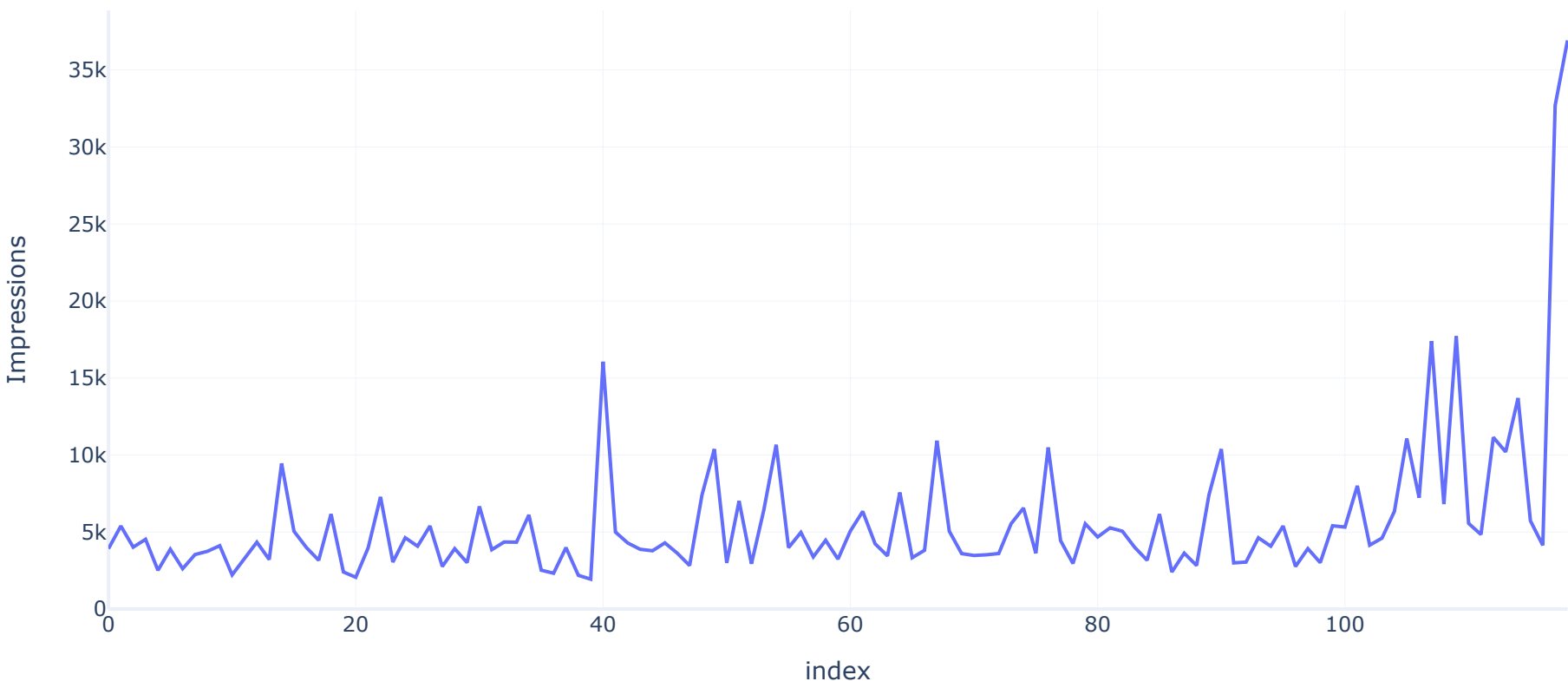
In [10]: `inst.head(3)`

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows	Caption	Hashtags
0	3920	2586	1028	619	56	98	9	5	162	35	2	Here are some of the most important data visua...	#finance #money #business #investing #investme...
1	5394	2727	1838	1174	78	194	7	14	224	48	10	Here are some of the best data science project...	#healthcare #health #covid #data #datascience ...
2	4021	2085	1188	0	533	41	11	1	131	62	12	Learn how to train a machine learning model an...	#data #datascience #dataanalysis #dataanalytic...

In [11]: `# graph for impressions on each post over time`

In [12]: `fig = go.Figure(go.Scatter(y=inst['Impressions']))  
fig.update_layout(  
 title='Impressions Over Time',  
 xaxis_title='index',  
 yaxis_title='Impressions'  
)`

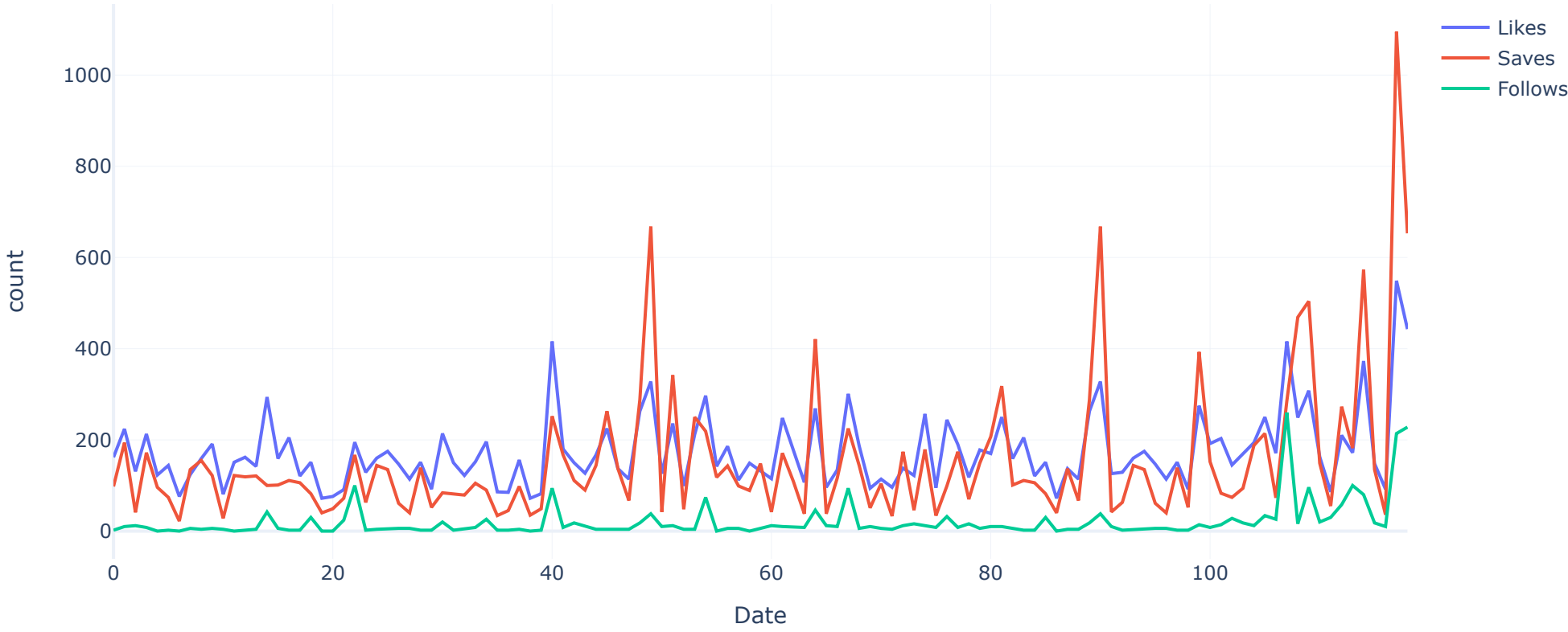
Impressions Over Time



In [13]: `# Graph for Likes, Saves, and Follows from each post over time`

In [14]: `fig = go.Figure(  
 [  
 go.Scatter(y=inst['Likes'], mode='lines', name='Likes'),  
 go.Scatter(y=inst['Saves'], mode='lines', name='Saves'),  
 go.Scatter(y=inst['Follows'], mode='lines', name='Follows')  
 ]  
)  
  
# Update layout with plot title and axis labels  
fig.update_layout(  
 title='Metrics Over Time',  
 xaxis_title='Date',  
 yaxis_title='count'  
)  
  
# Show the plot  
fig.show()`

Metrics Over Time



In [15]: `inst.head()`

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows	Caption	Hashtags
0	3920	2586	1028	619	56	98	9	5	162	35	2	Here are some of the most important data visua...	#finance #money #business #investing #investme...
1	5394	2727	1838	1174	78	194	7	14	224	48	10	Here are some of the best data science project...	#healthcare #health #covid #data #datascience ...
2	4021	2085	1188	0	533	41	11	1	131	62	12	Learn how to train a machine learning model an...	#data #datascience #dataanalysis #dataanalytic...
3	4528	2700	621	932	73	172	10	7	213	23	8	Here's how you can write a Python program to d...	#python #pythonprogramming #pythonprojects #py...
4	2518	1704	255	279	37	96	5	4	123	8	0	Plotting annotations while visualizing your da...	#datavisualization #datascience #data #dataana...

In [16]: `# Distribution of reach from different sources`

In [17]: `impression_sum = inst.sum()`  
`impression_sum`

Out[17]:	Impressions	678775
	From Home	294619
	From Hashtags	224614
	From Explore	128294
	From Other	20360
	Saves	18244
	Comments	793
	Shares	1114
	Likes	20680
	Profile Visits	6024
	Follows	2470
	Caption	Here are some of the most important data visua...
	Hashtags	#finance #money #business #investing #investme...
	dtype:	object

In [18]: `# now we required some of the columns only for this`  
`dff = inst[['Impressions', 'From Home', 'From Hashtags', 'From Explore', 'From Other']]`

In [19]: `dff`

	Impressions	From Home	From Hashtags	From Explore	From Other
0	3920	2586	1028	619	56
1	5394	2727	1838	1174	78
2	4021	2085	1188	0	533
3	4528	2700	621	932	73
4	2518	1704	255	279	37
...	...	...	...	...	...
114	13700	5185	3041	5352	77
115	5731	1923	1368	2266	65
116	4139	1133	1538	1367	33
117	32695	11815	3147	17414	170
118	36919	13473	4176	16444	2547

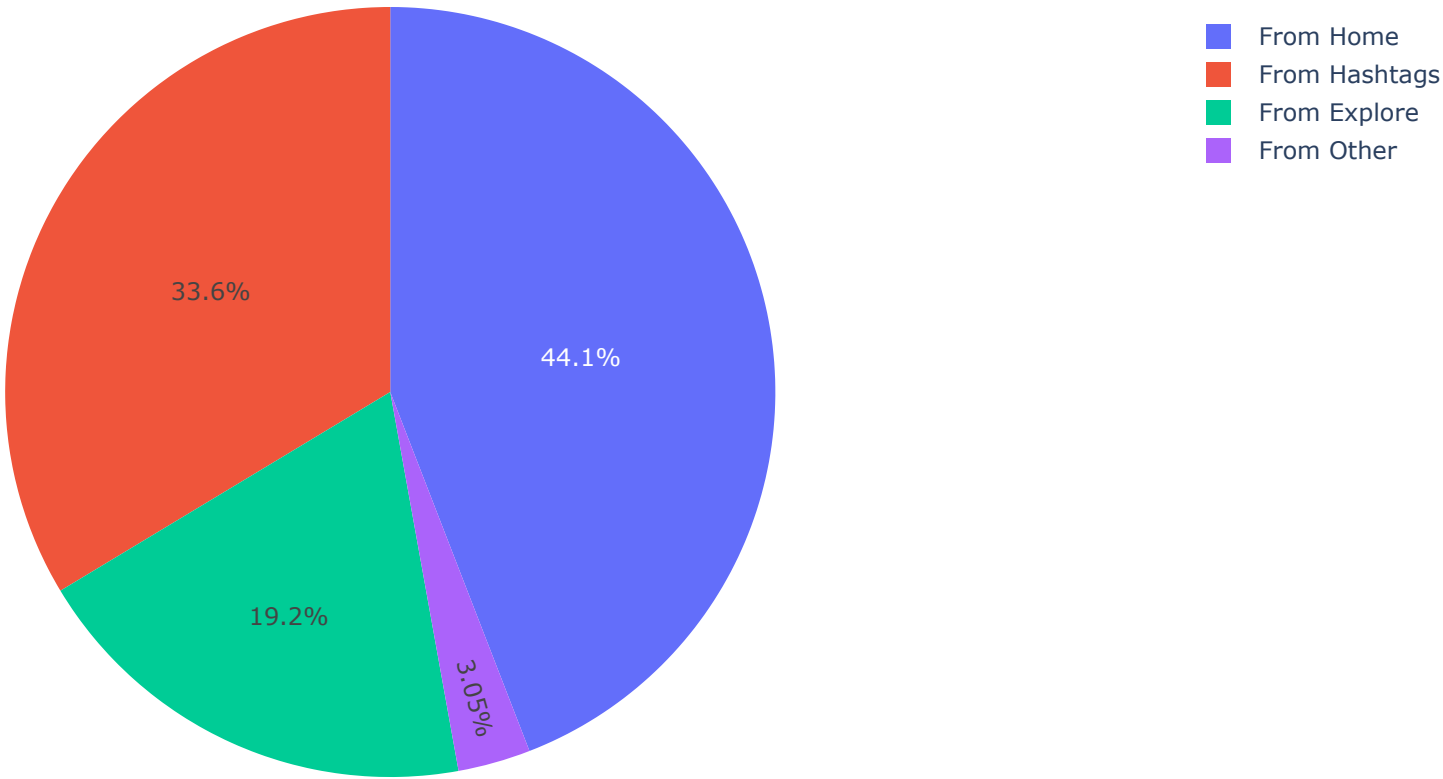
119 rows × 5 columns

In [20]: `# now i find the sum of this dff`  
`imp = dff.sum()`  
`imp`

Out[20]:	Impressions	678775
	From Home	294619
	From Hashtags	224614
	From Explore	128294
	From Other	20360
	dtype:	int64

In [21]: `fig = px.pie(names = imp.index[1:], values = imp.values[1:],)`

In [22]: `fig`



```
In [23]: # to find the distribution of engagement sources

In [24]: eng_sources_df = inst[['Impressions','Likes','Saves','Shares','Comments']]

In [25]: eng_sources_df

Out[25]:
   Impressions  Likes  Saves  Shares  Comments
0          3920    162    98      5          9
1          5394    224   194     14          7
2          4021    131    41      1         11
3          4528    213   172      7         10
4          2518    123    96      4          5
...         ...    ...    ...    ...        ...
114        13700    373   573     38          2
115         5731    148   135      1          4
116         4139     92    36      1          0
117        32695   549  1095     75          2
118        36919   443   653     26          5

119 rows x 5 columns

In [26]: eng_sources_df = eng_sources_df.sum()

In [27]: eng_sources_df

Out[27]:
Impressions    678775
Likes          20680
Saves         18244
Shares         1114
Comments         793
dtype: int64

In [28]: names = eng_sources_df.index[1:,]
names

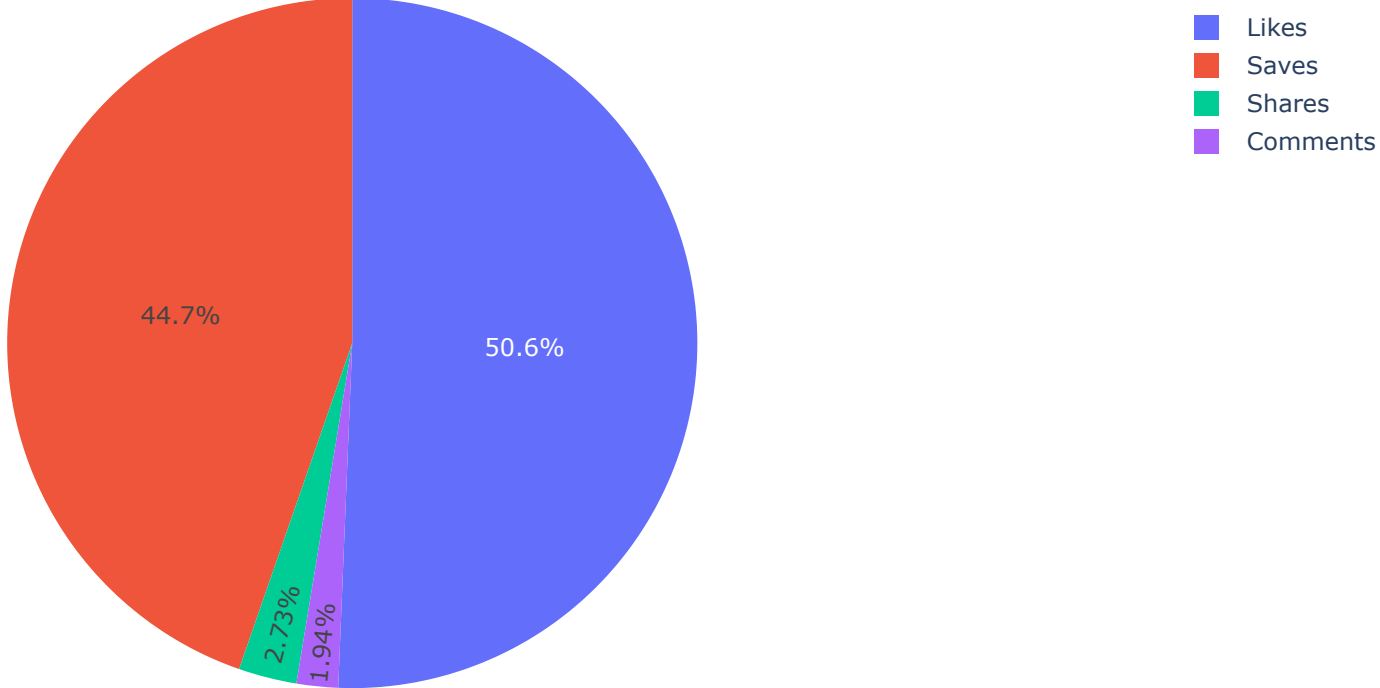
Out[28]:
Index(['Likes', 'Saves', 'Shares', 'Comments'], dtype='object')

In [29]: values = eng_sources_df.values[1:]
values

Out[29]:
array([20680, 18244, 1114, 793], dtype=int64)

In [30]: fig = px.pie(names = names,values = values,title='Engagement Sources')
fig.show()
```

Engagement Sources



```
In [31]: # Now i will find the number of Profile Visits and follows

fig = px.scatter(
    inst, x='Profile Visits', y='Follows', opacity=0.65,
    trendline='ols',
)
fig.show()
```

```
In [33]: # Now i will generate a worldcloud
```

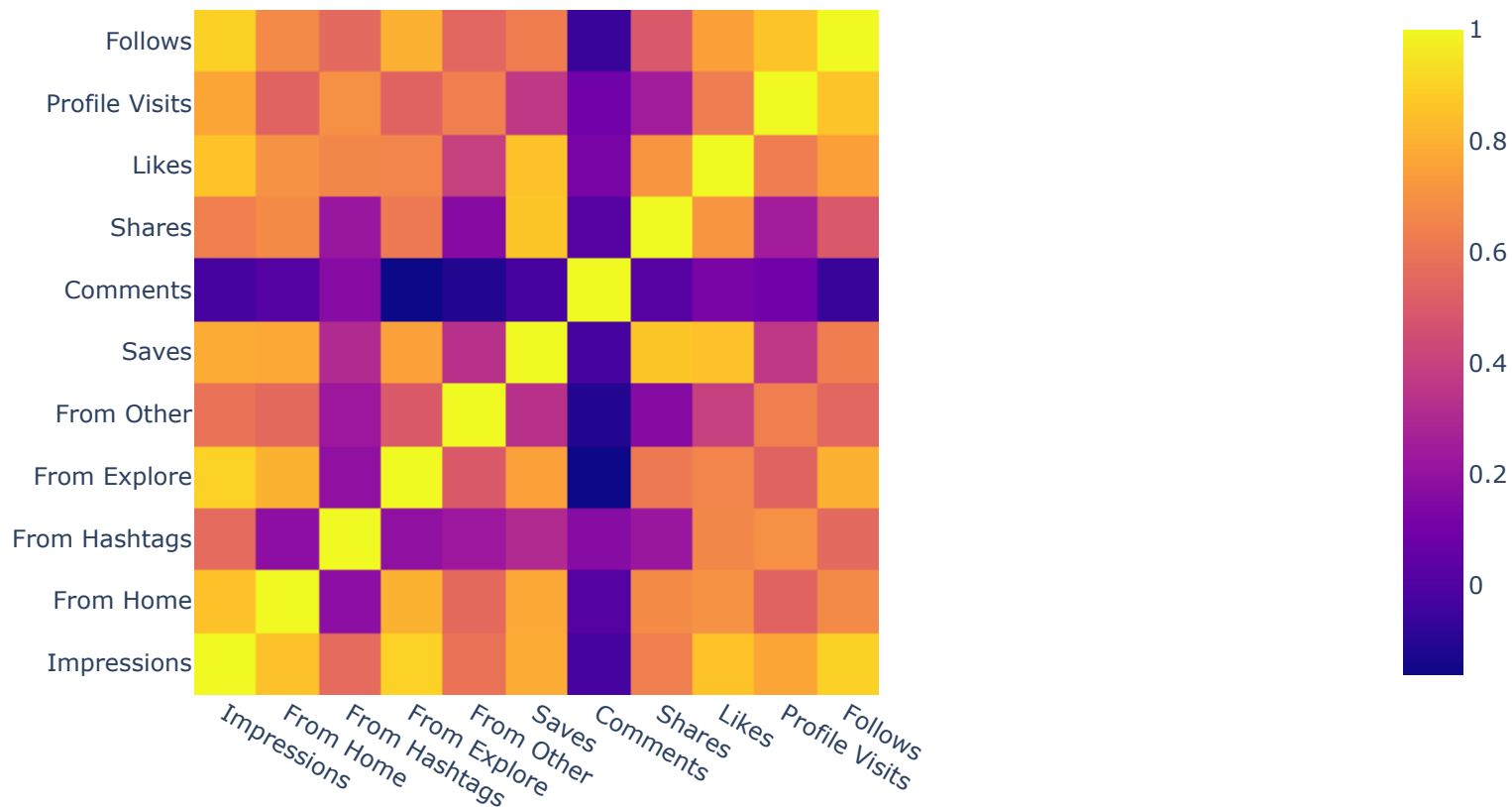
```
In [36]: inst.sample()
```

analytic...

Out[37]:	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
Impressions	1.000000	0.844698	0.560760	0.893607	0.592960	0.779231	-0.028524	0.634675	0.849835	0.760981	0.889363
From Home	0.844698	1.000000	0.177516	0.800573	0.555666	0.768817	0.012716	0.674985	0.698330	0.531076	0.672675
From Hashtags	0.560760	0.177516	1.000000	0.190453	0.229623	0.305929	0.161439	0.219511	0.662124	0.691345	0.555485
From Explore	0.893607	0.800573	0.190453	1.000000	0.495685	0.747803	-0.158565	0.615731	0.653699	0.531850	0.796019
From Other	0.592960	0.555666	0.229623	0.495685	1.000000	0.331907	-0.108703	0.156834	0.393510	0.633080	0.546737
Saves	0.779231	0.768817	0.305929	0.747803	0.331907	1.000000	-0.026912	0.860324	0.845643	0.360628	0.628461
Comments	-0.028524	0.012716	0.161439	-0.158565	-0.108703	-0.026912	1.000000	0.016933	0.123586	0.096714	-0.060631
Shares	0.634675	0.674985	0.219511	0.615731	0.156834	0.860324	0.016933	1.000000	0.707794	0.245361	0.493070
Likes	0.849835	0.698330	0.662124	0.653699	0.393510	0.845643	0.123586	0.707794	1.000000	0.626107	0.746333
Profile Visits	0.760981	0.531076	0.691345	0.531850	0.633080	0.360628	0.096714	0.245361	0.626107	1.000000	0.853152
Follows	0.889363	0.672675	0.555485	0.796019	0.546737	0.628461	-0.060631	0.493070	0.746333	0.853152	1.000000



## Features VS Features



```
In [39]: # distribution of hashtags to see which hashtag is used the most
```

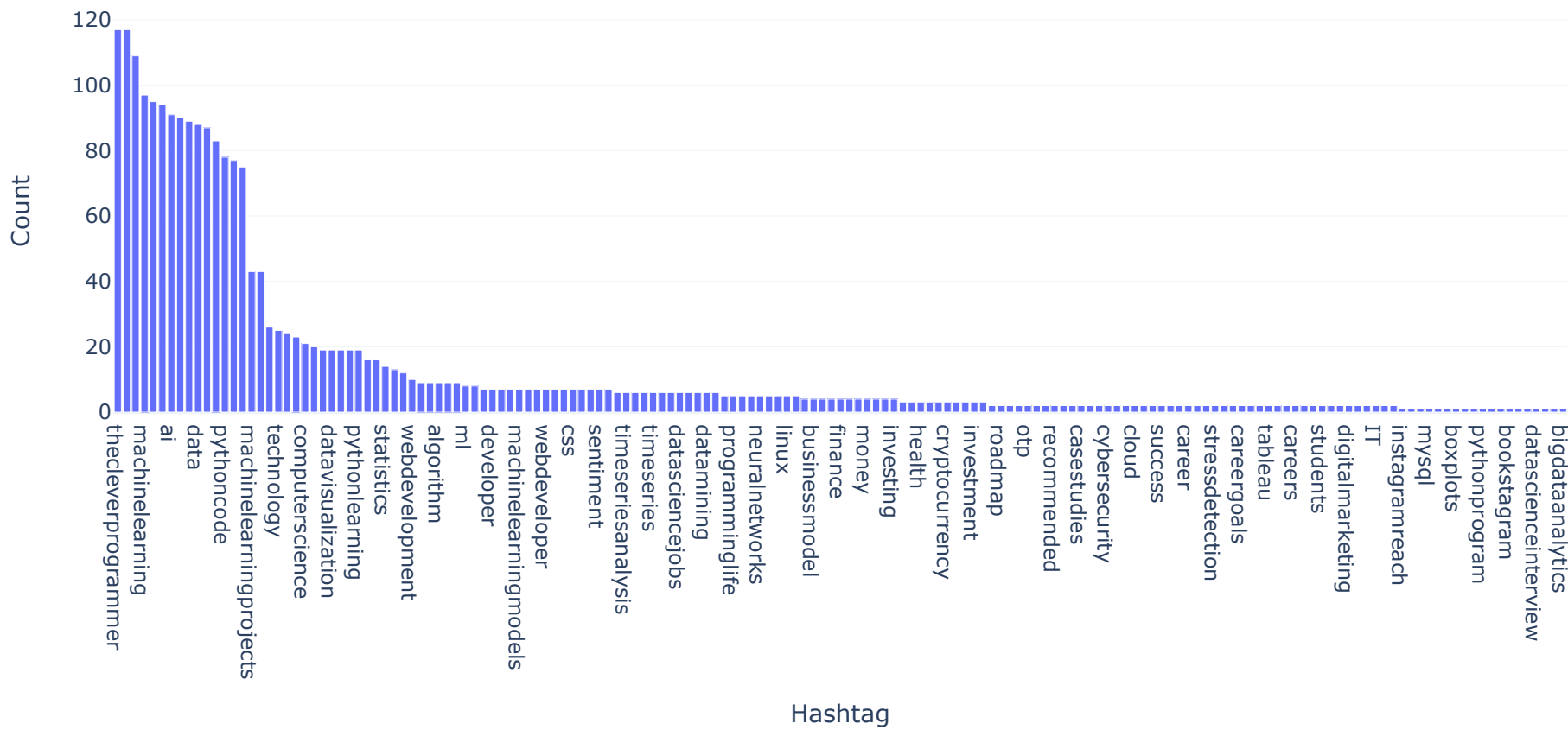
```
hashtags = inst['Hashtags'].str.findall(r'#\w+').explode()

hashtag_counts = hashtag_counts.value_counts()

hashtag_df = pd.DataFrame(hashtag_counts).reset_index()
hashtag_df.columns = ['Hashtag', 'Count']

fig = px.bar(hashtag_df, x='Hashtag', y='Count', title='Hashtag Distribution')
fig.show()
```

## Hashtag Distribution



```
In [40]: # distribution of likes and impressions received from the presence of each hashtag on the post
```

```
hashtags = inst['Hashtags'].str.findall(r'#\w+').explode()

hashtag_counts = hashtag_counts.value_counts()

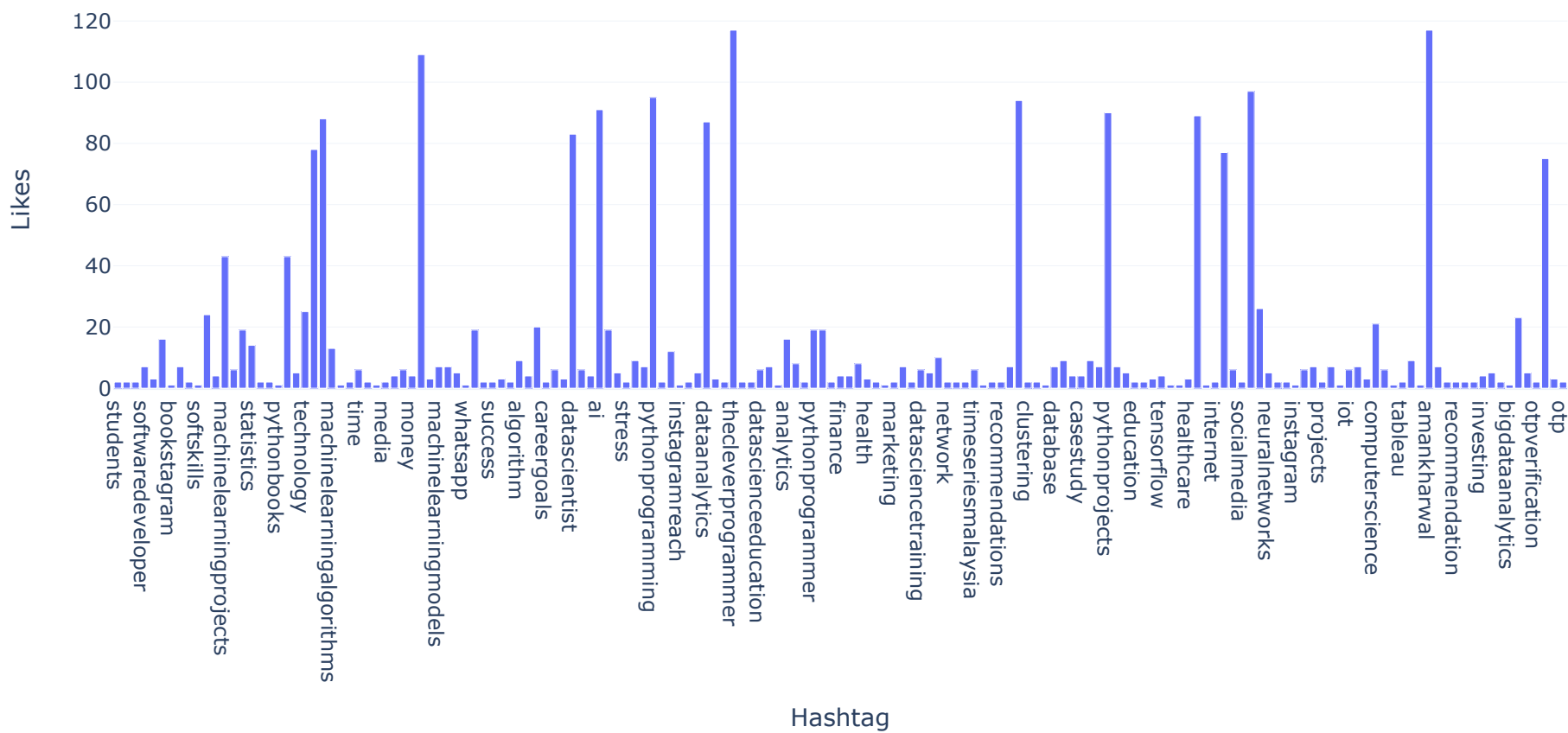
hashtag_df = pd.DataFrame(hashtag_counts).reset_index()
hashtag_df.columns = ['Hashtag', 'Likes']

hashtag_df = hashtag_df.sample(frac=1, random_state=42)

fig = px.bar(hashtag_df, x='Hashtag', y='Likes', title='Likes Distribution for each Hashtags')

fig.show()
```

### Likes Distribution for each Hashtags



```
In [42]: hashtags = inst['Hashtags'].str.findall(r'#(\w+)').explode()

        hashtag_counts = hashtags.value_counts()

        hashtag_df = pd.DataFrame(hashtag_counts).reset_index()
        hashtag_df.columns = ['Hashtag', 'Impressions']

        fig = px.bar(hashtag_df, x='Hashtag', y='Impressions', title='Hashtag Distribution')
        fig.show()
```

