Digital Dilemma: How Cashless Transactions are Impacting Spending Habits in India

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

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [3]:
```
df = pd.read_csv('upi_data.csv')
```

In [4]:
```
df
```

Out[4]:

	Month	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
0	Mar'24	572	13440.00	1978353.23
1	Feb'24	560	12102.67	1827869.35
2	Jan'24	550	12203.02	1841083.97
3	Dec'23	522	12020.23	1822949.42
4	Nov'23	516	11235.29	1739740.61
...	...	...	...	...
94	May-16	21	0.00	0.00
95	Apr-16	0	0.00	0.00
96	NaN	0	0.00	0.00
97	NaN	0	0.00	0.00
98	NaN	0	0.00	0.00

99 rows × 4 columns

In [5]:
```
df = df.dropna(subset=['Month'])
df = df[df['No. of Banks live on UPI'] != 0]
```

In [6]:
```
df
```

Out[6]:

	Month	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
0	Mar'24	572	13440.00	1978353.23
1	Feb'24	560	12102.67	1827869.35
2	Jan'24	550	12203.02	1841083.97
3	Dec'23	522	12020.23	1822949.42
4	Nov'23	516	11235.29	1739740.61
...	...	...	...	...
90	Sep-16	25	0.09	32.64
91	Aug-16	21	0.09	3.09
92	Jul-16	21	0.09	0.38
93	Jun-16	21	0.09	0.00
94	May-16	21	0.00	0.00

95 rows × 4 columns

In [7]:
```
df.shape
```

Out[7]:
```
(95, 4)
```

In [8]:
```
df.columns
```

Out[8]:
```
Index(['Month', 'No. of Banks live on UPI', 'Volume (in Mn)',
       'Value (in Cr.)'],
      dtype='object')
```

In [9]:
```
df.duplicated().sum()
```

Out[9]:
```
0
```

In [10]:
```
df.isnull().sum()
```

Out[10]:
```
Month                      0
No. of Banks live on UPI   0
Volume (in Mn)             0
Value (in Cr.)             0
dtype: int64
```

In [11]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 95 entries, 0 to 94
Data columns (total 4 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Month                     95 non-null     object 
 1   No. of Banks live on UPI  95 non-null     int64  
 2   Volume (in Mn)            95 non-null     float64
 3   Value (in Cr.)            95 non-null     float64
dtypes: float64(2), int64(1), object(1)
memory usage: 3.7+ KB
```

In [12]:
```
df.describe()
```

Out[12]:

	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
count	95.000000	95.000000	9.500000e+01
mean	209.463158	3179.550842	5.216219e+05
std	150.746710	3769.049211	5.859732e+05
min	21.000000	0.000000	0.000000e+00
25%	99.000000	189.780000	3.015518e+04
50%	153.000000	1308.400000	2.183916e+05
75%	315.000000	5494.350000	9.719420e+05
max	572.000000	13440.000000	1.978353e+06

In [13]:
```
df.nunique()
```

Out[13]:
```
Month                       95
No. of Banks live on UPI    80
Volume (in Mn)              92
Value (in Cr.)              94
dtype: int64
```

In [14]:
```
import matplotlib.pyplot as plt
```

In [15]:
```
df['Month']
```

Out[15]:
```
0     Mar'24
1     Feb'24
2     Jan'24
3     Dec'23
4     Nov'23
       ...
90    Sep-16
91    Aug-16
92    Jul-16
93    Jun-16
94    May-16
Name: Month, Length: 95, dtype: object
```

In [16]:
```
month_map = {
    'Jan': 'January',
    'Feb': 'February',
    'Mar': 'March',
    'Apr': 'April',
    'May': 'May',
    'Jun': 'June',
    'Jul': 'July',
    'Aug': 'August',
    'Sep': 'September',
    'Oct': 'October',
    'Nov': 'November',
    'Dec': 'December'
}
```

In [17]:
```
df['Month'] = df['Month'].apply(lambda x: x.replace("'", "-"))
```

In [18]:
```
df
```

Out[18]:

	Month	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
0	Mar-24	572	13440.00	1978353.23
1	Feb-24	560	12102.67	1827869.35
2	Jan-24	550	12203.02	1841083.97
3	Dec-23	522	12020.23	1822949.42
4	Nov-23	516	11235.29	1739740.61
...	...	...	...	...
90	Sep-16	25	0.09	32.64
91	Aug-16	21	0.09	3.09
92	Jul-16	21	0.09	0.38
93	Jun-16	21	0.09	0.00
94	May-16	21	0.00	0.00

95 rows × 4 columns

In [19]:
```
df['Month']
```

Out[19]:
```
0     Mar-24
1     Feb-24
2     Jan-24
3     Dec-23
4     Nov-23
       ...
90    Sep-16
91    Aug-16
92    Jul-16
93    Jun-16
94    May-16
Name: Month, Length: 95, dtype: object
```

In [20]:
```
df['Month'] = df['Month'].str.replace('-', ' ')
```

In [21]:
```
df
```

Out[21]:

	Month	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
0	Mar 24	572	13440.00	1978353.23
1	Feb 24	560	12102.67	1827869.35
2	Jan 24	550	12203.02	1841083.97
3	Dec 23	522	12020.23	1822949.42
4	Nov 23	516	11235.29	1739740.61
...	...	...	...	...
90	Sep 16	25	0.09	32.64
91	Aug 16	21	0.09	3.09
92	Jul 16	21	0.09	0.38
93	Jun 16	21	0.09	0.00
94	May 16	21	0.00	0.00

95 rows × 4 columns

In [22]:
```
df.set_index('Month', inplace=True)
```

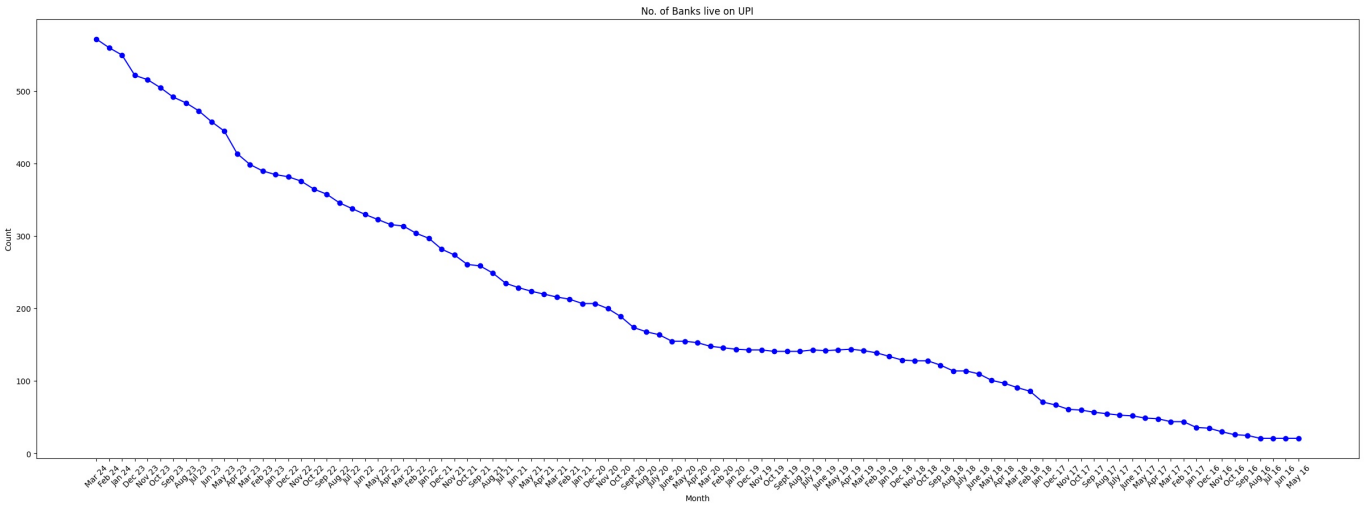
In [23]:
```
df
```

Out[23]:

Month	No. of Banks live on UPI	Volume (in Mn)	Value (in Cr.)
Mar 24	572	13440.00	1978353.23
Feb 24	560	12102.67	1827869.35
Jan 24	550	12203.02	1841083.97
Dec 23	522	12020.23	1822949.42
Nov 23	516	11235.29	1739740.61
...	...	...	...
Sep 16	25	0.09	32.64
Aug 16	21	0.09	3.09
Jul 16	21	0.09	0.38
Jun 16	21	0.09	0.00
May 16	21	0.00	0.00

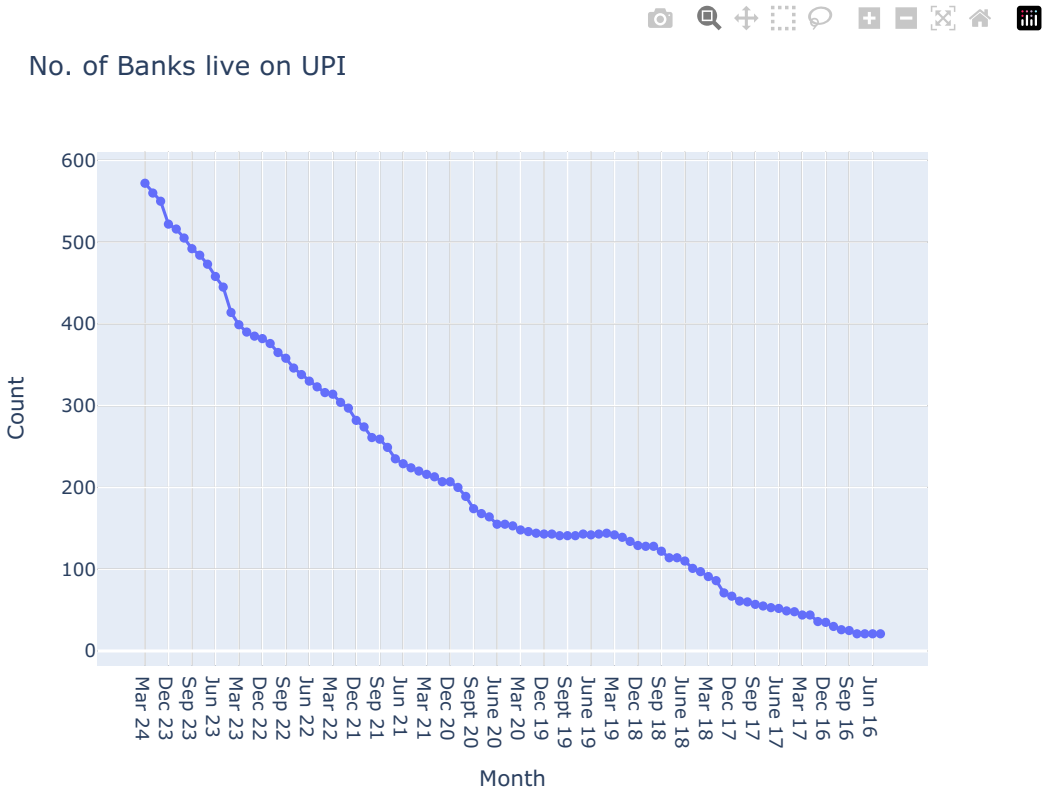
95 rows × 3 columns

In [24]:
```
plt.figure(figsize=(30, 10))
plt.plot(df.index, df['No. of Banks live on UPI'], marker='o', color='b')
plt.title('No. of Banks live on UPI')
plt.xticks(rotation = 45)
plt.xlabel('Month')
plt.ylabel('Count')
plt.show()
```

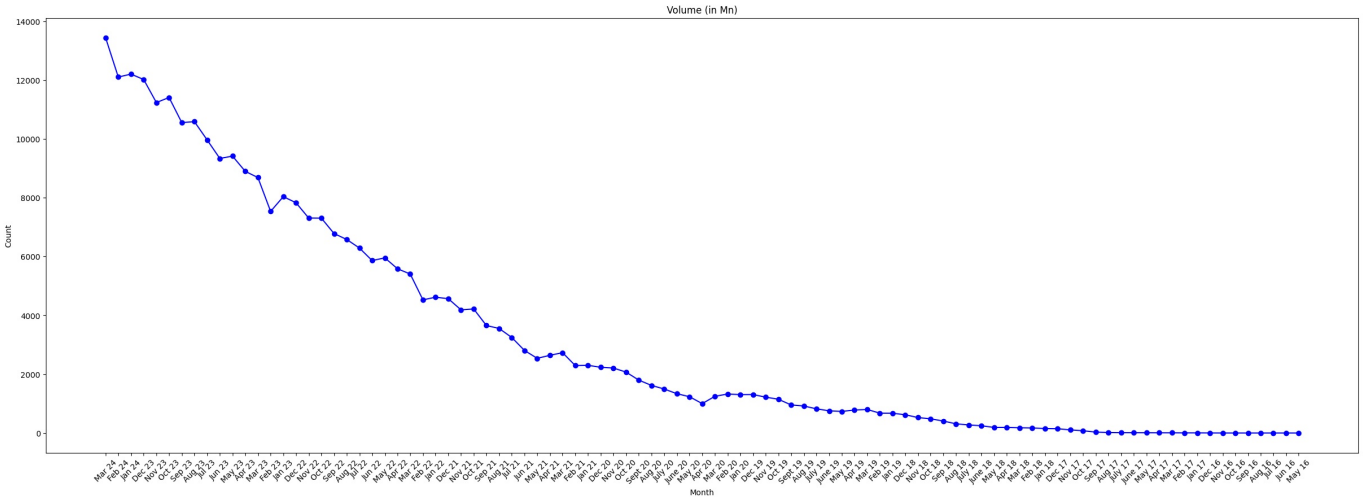


In [25]:
```
import plotly.graph_objects as go
```

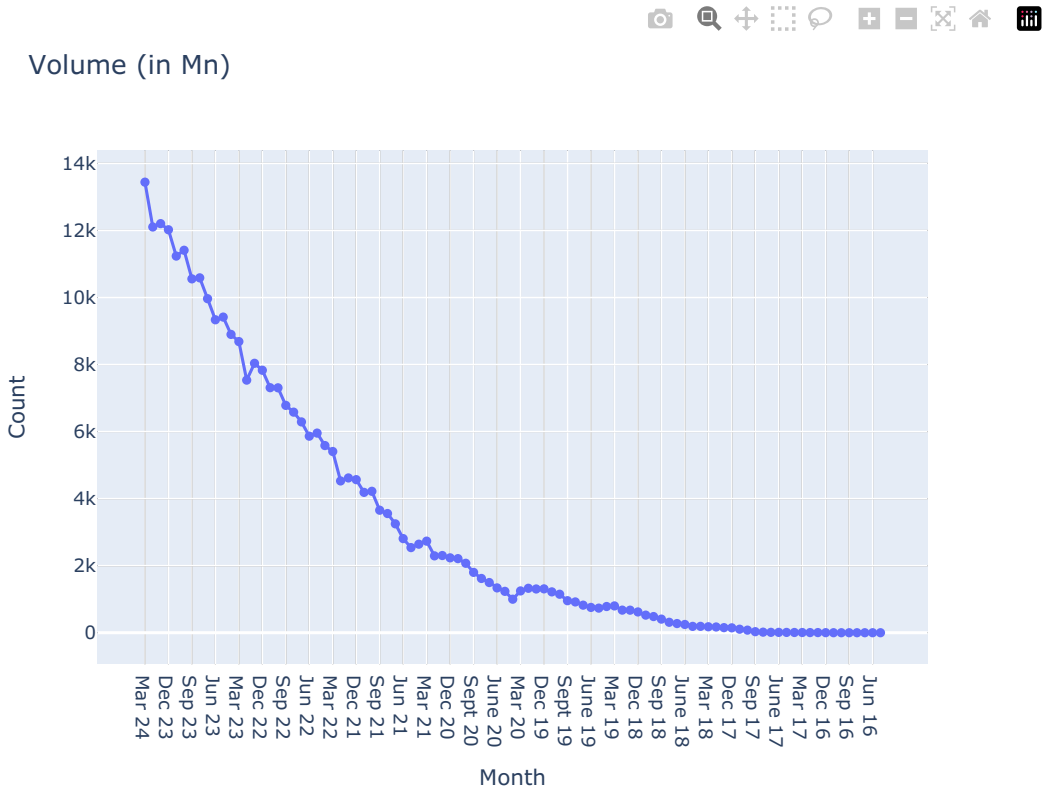
In [26]:
```
fig = go.Figure()
fig.add_trace(go.Scatter(x=df.index, y=df['No. of Banks live on UPI'], mode='lines+markers', name='No. of Banks
fig.update_layout(title='No. of Banks live on UPI', xaxis_title='Month', yaxis_title='Count')
fig.show()
```



In [27]:
```
plt.figure(figsize=(30, 10))
plt.plot(df.index, df['Volume (in Mn)'], marker='o', color='b')
plt.title('Volume (in Mn)')
plt.xticks(rotation = 45)
plt.xlabel('Month')
plt.ylabel('Count')
plt.show()
```

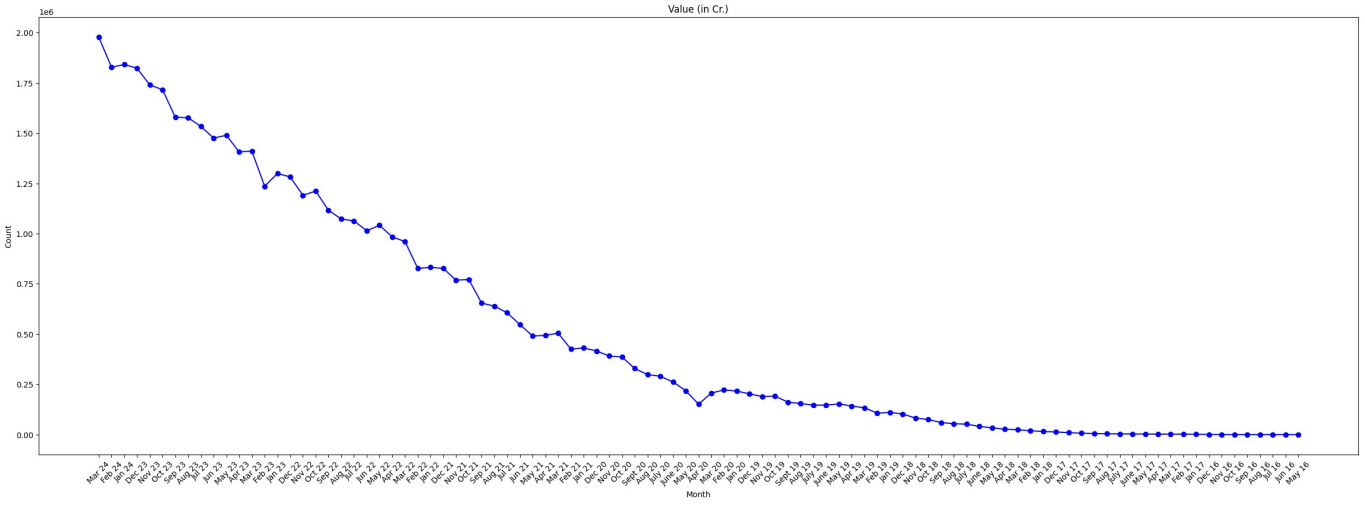


In [28]:
```
fig = go.Figure()
fig.add_trace(go.Scatter(x=df.index, y=df['Volume (in Mn)'], mode='lines+markers', name='Volume (in Mn)'))
fig.update_layout(title='Volume (in Mn)', xaxis_title='Month', yaxis_title='Count')
fig.show()
```



In [29]:

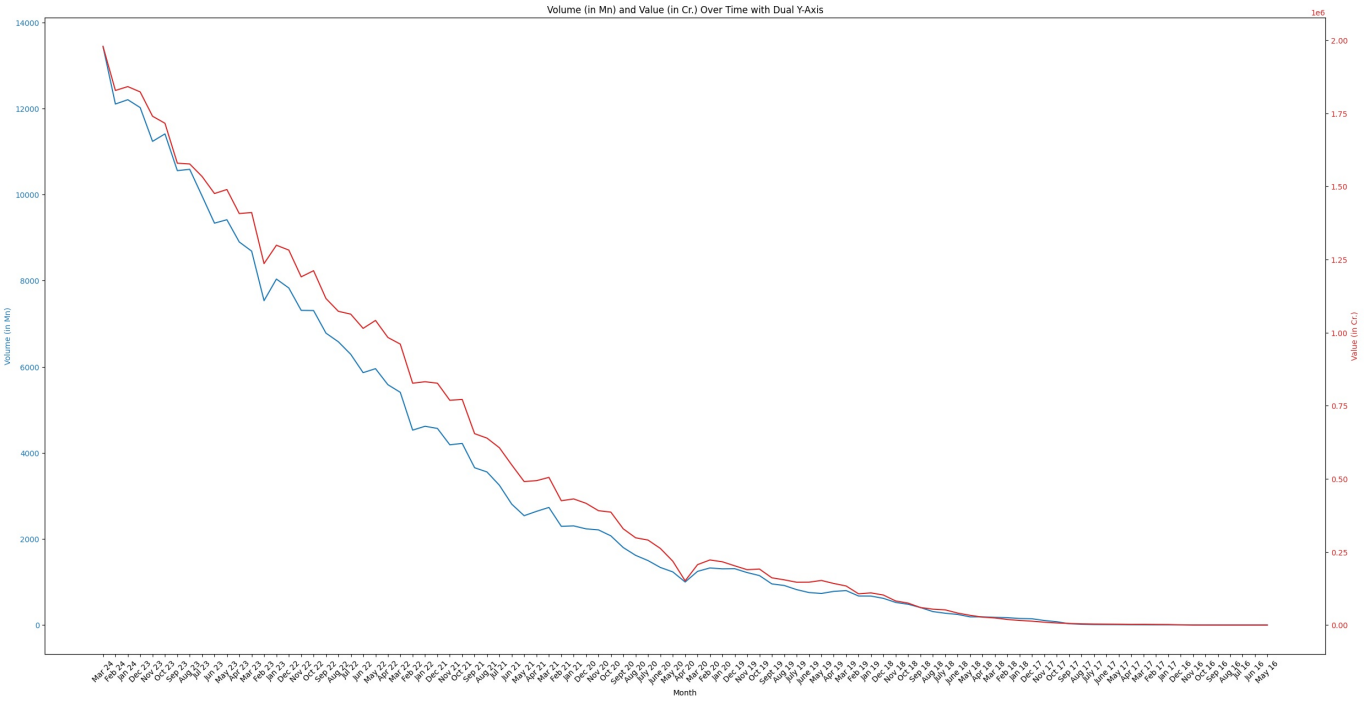
```
plt.figure(figsize=(30, 10))
plt.plot(df.index, df['Value (in Cr.)'], marker='o', color='b')
plt.title('Value (in Cr.)')
plt.xticks(rotation = 45)
plt.xlabel('Month')
plt.ylabel('Count')
plt.show()
```



In [30]:

```
fig = go.Figure()
fig.add_trace(go.Scatter(x=df.index, y=df['Value (in Cr.)'], mode='lines+markers', name='Value (in Cr.)'))
fig.update_layout(title='Value (in Cr.)', xaxis_title='Month', yaxis_title='Count')
fig.show()
```

In [31]:
```
fig, ax1 = plt.subplots(figsize=(30, 15))

color = 'tab:blue'
ax1.set_xlabel('Month')
ax1.set_ylabel('Volume (in Mn)', color=color)
ax1.plot(df.index, df['Volume (in Mn)'], color=color)
plt.xticks(rotation=45)
ax1.tick_params(axis='y', labelcolor=color)

ax2 = ax1.twinx()
color = 'tab:red'
ax2.set_ylabel('Value (in Cr.)', color=color)
ax2.plot(df.index, df['Value (in Cr.)'], color=color)
plt.xticks(rotation=45)
ax2.tick_params(axis='y', labelcolor=color)

plt.title('Volume (in Mn) and Value (in Cr.) Over Time with Dual Y-Axis')
plt.xticks(rotation=45)
plt.show()
```



In [32]:
```
fig = go.Figure()

fig.add_trace(go.Scatter(x=df.index, y=df['Volume (in Mn)'], mode='lines', name='Volume (in Mn)', line=dict(colo
```

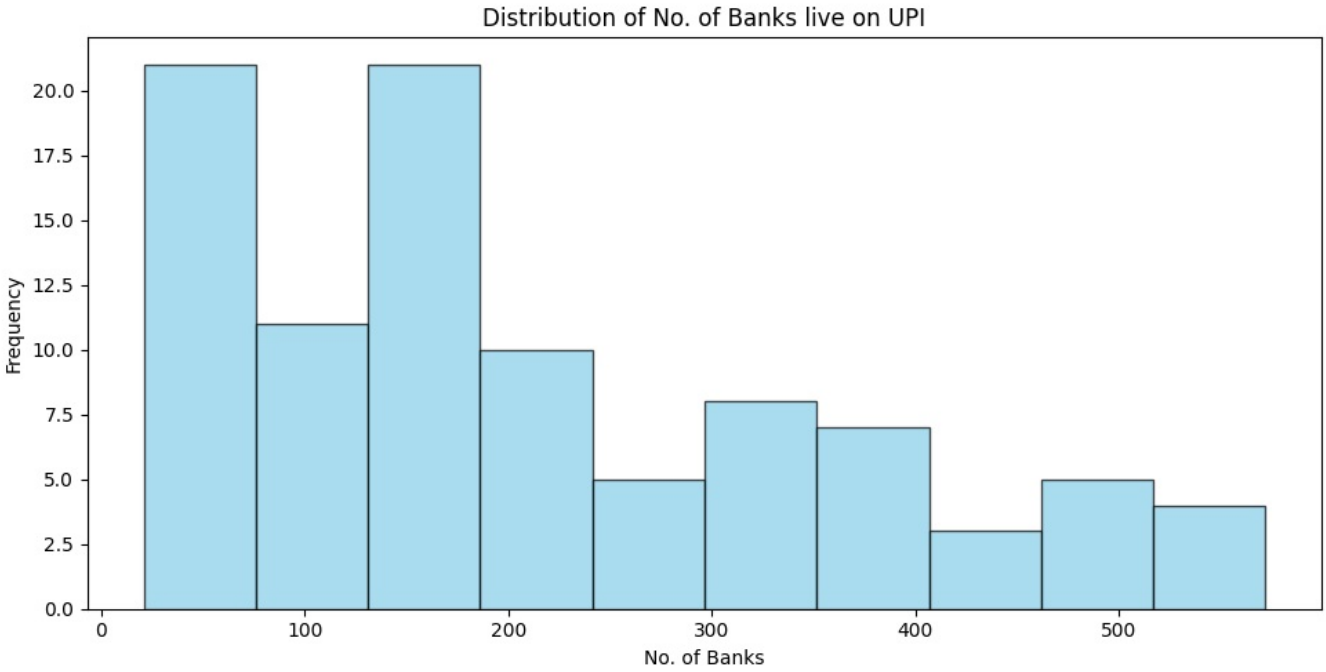
```
fig.add_trace(go.Scatter(x=df.index, y=df['Value (in Cr.)'], mode='lines', name='Value (in Cr.)', line=dict(col

fig.update_layout(
    title='Volume (in Mn) and Value (in Cr.) Over Time with Dual Y-Axis',
    xaxis=dict(title='Month', tickangle=45),
    yaxis=dict(title='Volume (in Mn)', color='blue'),
    yaxis2=dict(title='Value (in Cr.)', color='red', overlaying='y', side='right'),
    legend=dict(x=0, y=1.1, orientation='h'),
    margin=dict(l=50, r=50, t=100, b=50),
    height=600,
    width=1200
)

fig.show()
```

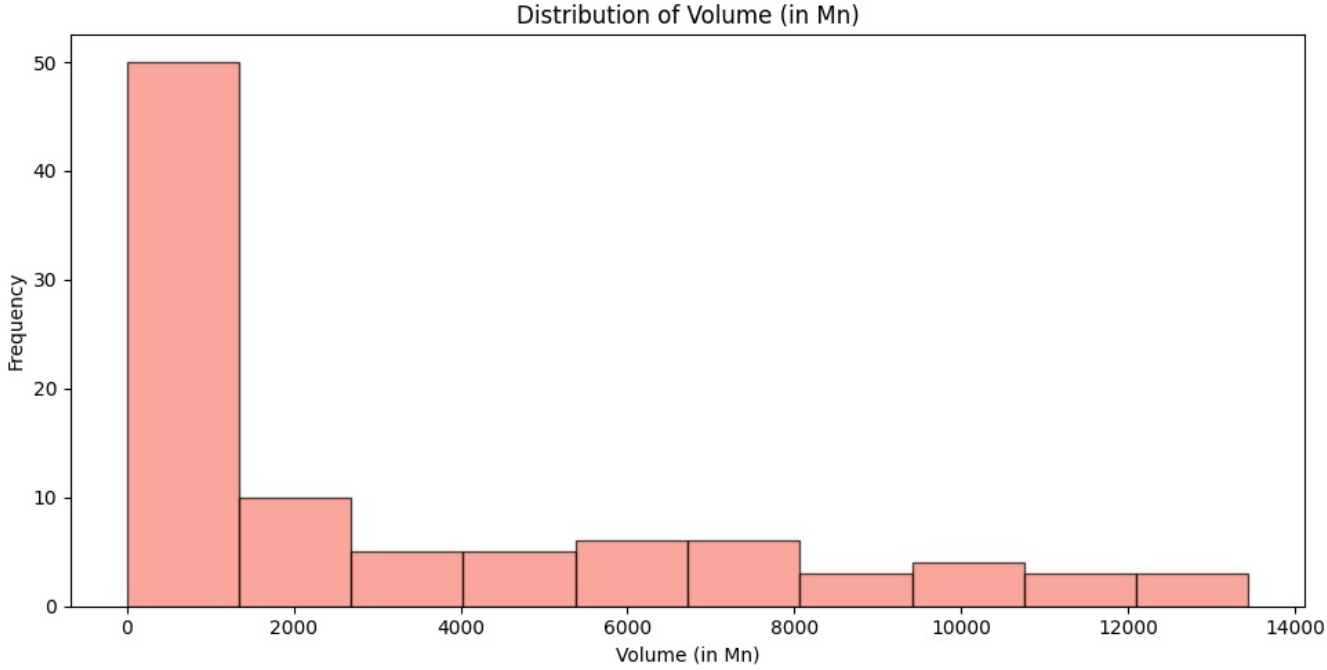
In [33]:
```
fig = go.Figure()

fig.add_trace(go.Bar(
    x=df.index,
    y=df['No. of Banks live on UPI'],
    name='No. of Banks live on UPI',
    marker_color='green'
))

fig.update_layout(
    title='Number of Banks Live on UPI Over Time',
    xaxis=dict(title='Month', tickangle=45),
    yaxis=dict(title='Count'),
    legend=dict(x=0, y=1.1, orientation='h'),
    margin=dict(l=50, r=50, t=100, b=50),
    height=600,
    width=1200
)

fig.show()
```

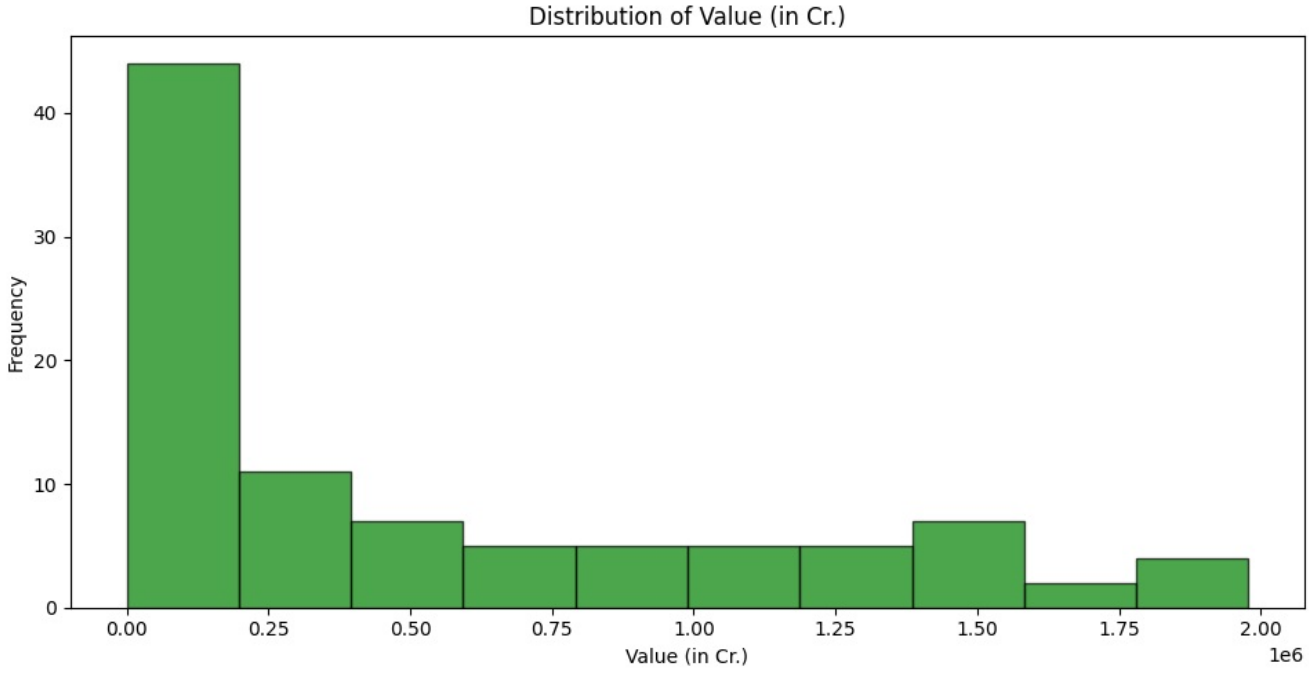
In [34]:
```
df.columns
```

Out[34]:
```
Index(['No. of Banks live on UPI', 'Volume (in Mn)', 'Value (in Cr.)'], dtype='object')
```

In [35]:
```
fig, axes = plt.subplots(nrows=3, ncols=1, figsize=(10, 15))

axes[0].hist(df['No. of Banks live on UPI'], bins=10, color='skyblue', edgecolor='black', alpha=0.7)
axes[0].set_title('Distribution of No. of Banks live on UPI')
axes[0].set_xlabel('No. of Banks')
axes[0].set_ylabel('Frequency')

axes[1].hist(df['Volume (in Mn)'], bins=10, color='salmon', edgecolor='black', alpha=0.7)
axes[1].set_title('Distribution of Volume (in Mn)')
axes[1].set_xlabel('Volume (in Mn)')
axes[1].set_ylabel('Frequency')

axes[2].hist(df['Value (in Cr.)'], bins=10, color='green', edgecolor='black', alpha=0.7)
axes[2].set_title('Distribution of Value (in Cr.)')
axes[2].set_xlabel('Value (in Cr.)')
axes[2].set_ylabel('Frequency')

plt.tight_layout()
plt.show()
```


Distribution of No. of Banks live on UPI
Frequency
No. of Banks
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
100
200
300
400
500


Distribution of Volume (in Mn)
Frequency
Volume (in Mn)
0
10
20
30
40
50
0
2000
4000
6000
8000
10000
12000
14000


Distribution of Value (in Cr.)
Frequency
Value (in Cr.)
1e6
0
10
20
30
40
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

Thanks !!!