INSTAGRAM REACH ANALYSIS USING PYTHON

Import the necessary libraries

In [8]:

```
import pandas as pd
import numpy as np
import plotly.graph_objs as go
import plotly.express as px
import plotly.io as pio
pio.templates.default = "plotly_white"
```

Next, we read the data

In [2]:

```
data = pd.read_csv("Instagram-Reach.csv", encoding = 'latin-1')
print(data.head())
```

```
                  Date  Instagram reach
0  2022-04-01T00:00:00             7620
1  2022-04-02T00:00:00            12859
2  2022-04-03T00:00:00            16008
3  2022-04-04T00:00:00            24349
4  2022-04-05T00:00:00            20532
```

We then convert the dtype of the date column to datetime

In [3]:

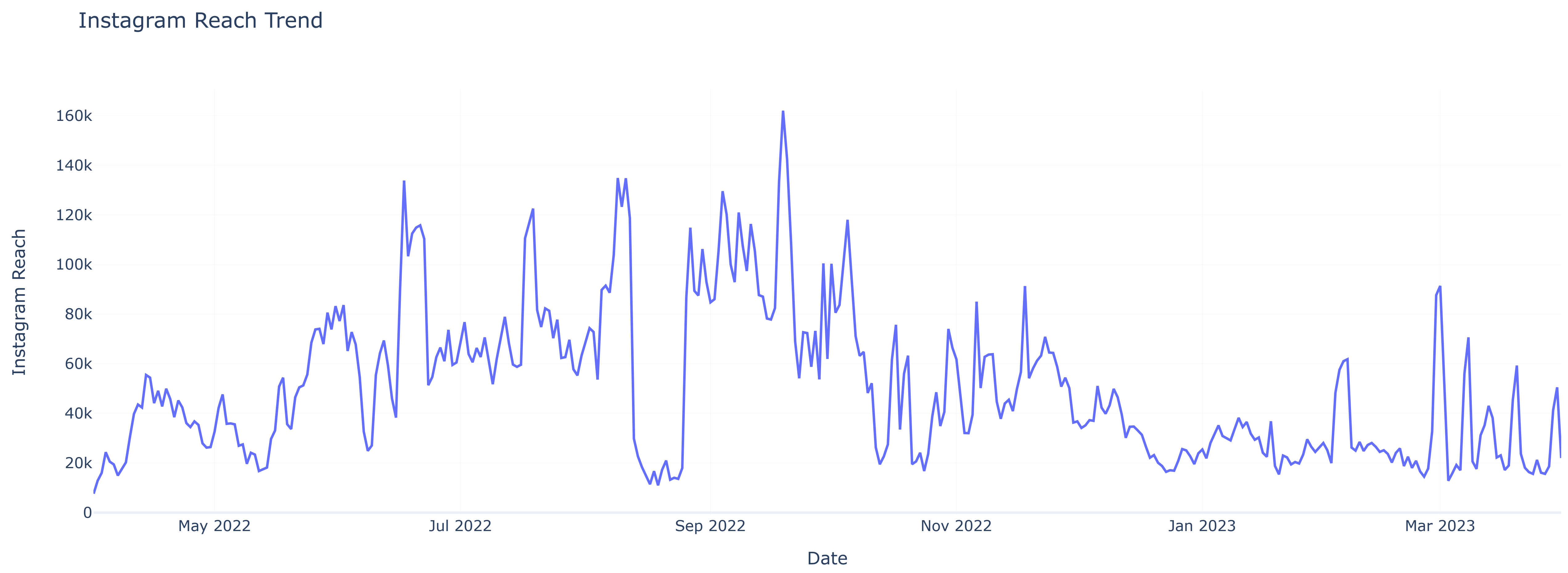
```
data['Date'] = pd.to_datetime(data['Date'])
print(data.head())
```

```
        Date  Instagram reach
0 2022-04-01             7620
1 2022-04-02            12859
2 2022-04-03            16008
3 2022-04-04            24349
4 2022-04-05            20532
```

We then analyse the trend of the instagam reach over time using a line chart

In [4]:

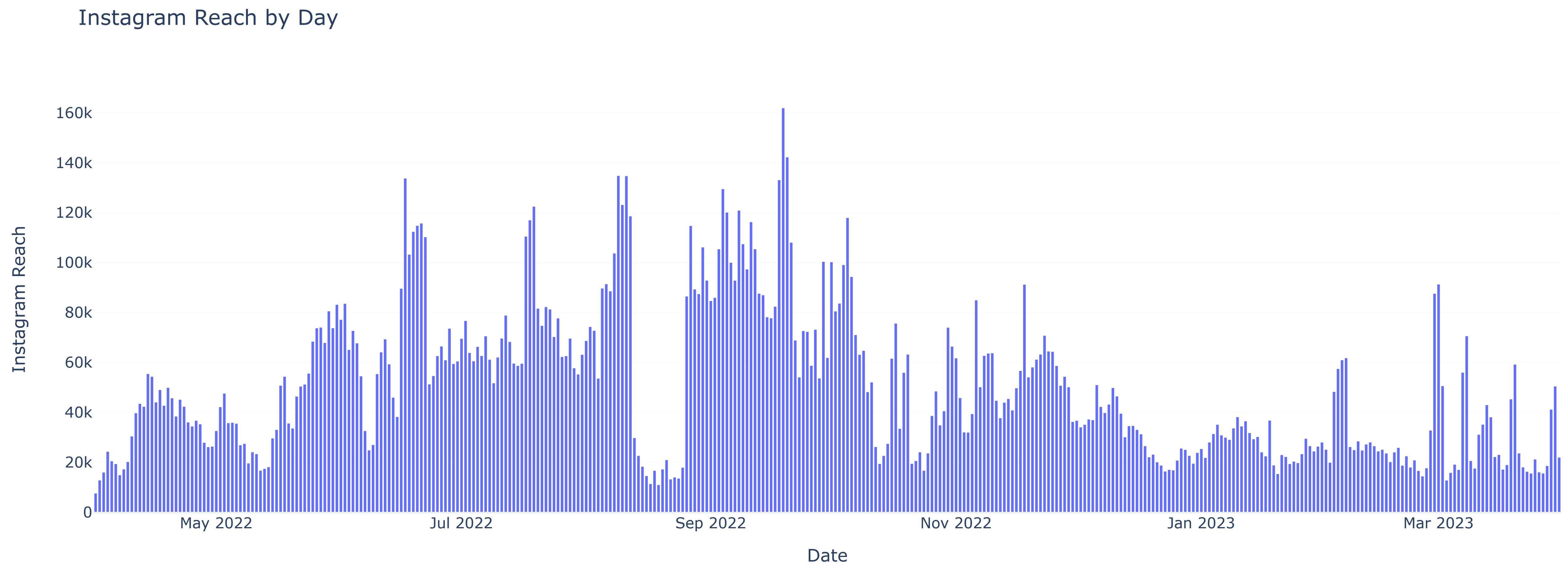
```
fig = go.Figure()
fig.add_trace(go.Scatter(x=data['Date'],
                         y=data['Instagram reach'],
                         mode='lines', name='Instagram reach'))
fig.update_layout(title='Instagram Reach Trend', xaxis_title='Date',
                  yaxis_title='Instagram Reach')
fig.show()
```



Next, we analyse the instagram reach for each day using a bar chart

In [5]:

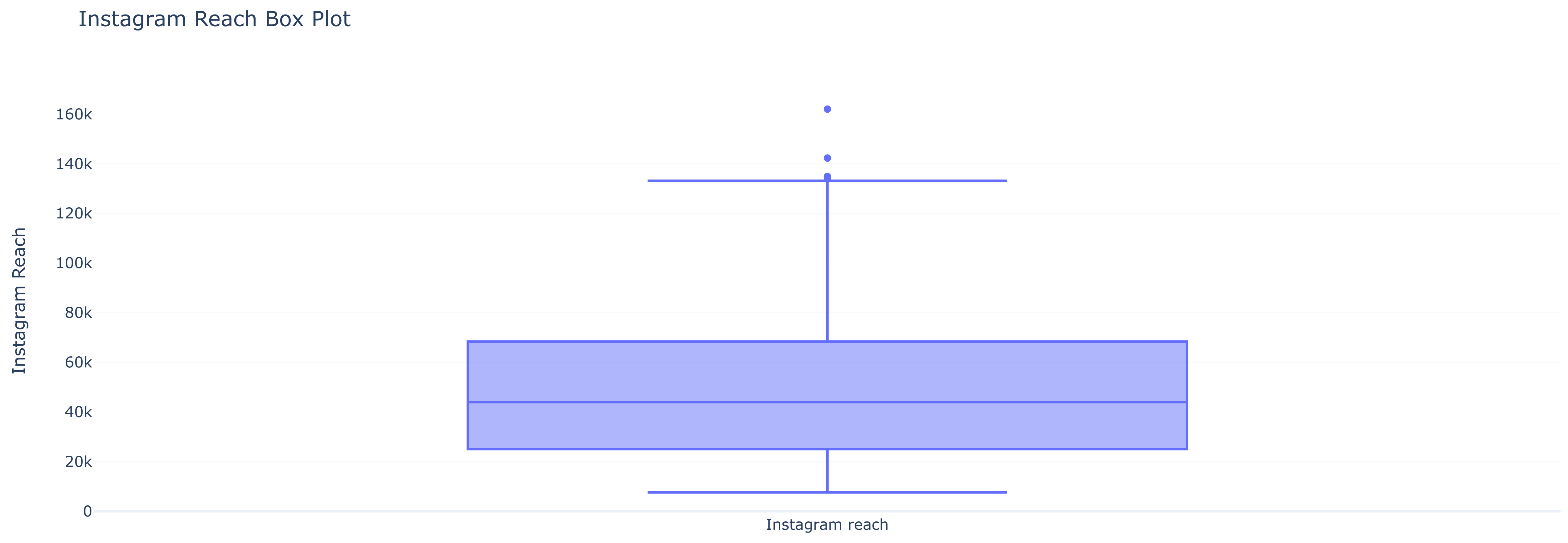
```
fig = go.Figure()
fig.add_trace(go.Bar(x=data['Date'],
                     y=data['Instagram reach'],
                     name='Instagram reach'))
fig.update_layout(title='Instagram Reach by Day',
                  xaxis_title='Date',
                  yaxis_title='Instagram Reach')
fig.show()
```



Using a boxplot, we analyse the distribution of the instagram reach

In [6]:

```
fig = go.Figure()
fig.add_trace(go.Box(y=data['Instagram reach'],
                     name='Instagram reach'))
fig.update_layout(title='Instagram Reach Box Plot',
                  yaxis_title='Instagram Reach')
fig.show()
```



Next we create a day column so we can analyse reach based on the day of the week. We do this by using dt.day_name() to extract the day of the week from the date column

In [7]:

```
 data['Day'] = data['Date'].dt.day_name()
print(data.head())
```

```
        Date  Instagram reach       Day
0 2022-04-01             7620    Friday
1 2022-04-02            12859  Saturday
2 2022-04-03            16008    Sunday
3 2022-04-04            24349    Monday
4 2022-04-05            20532   Tuesday
```

We move on to the analysis proper using days of the week. Using the groupby function via the day column we can calculate the mean, median and standard deviation of the instagram reach for each day.

In [9]:

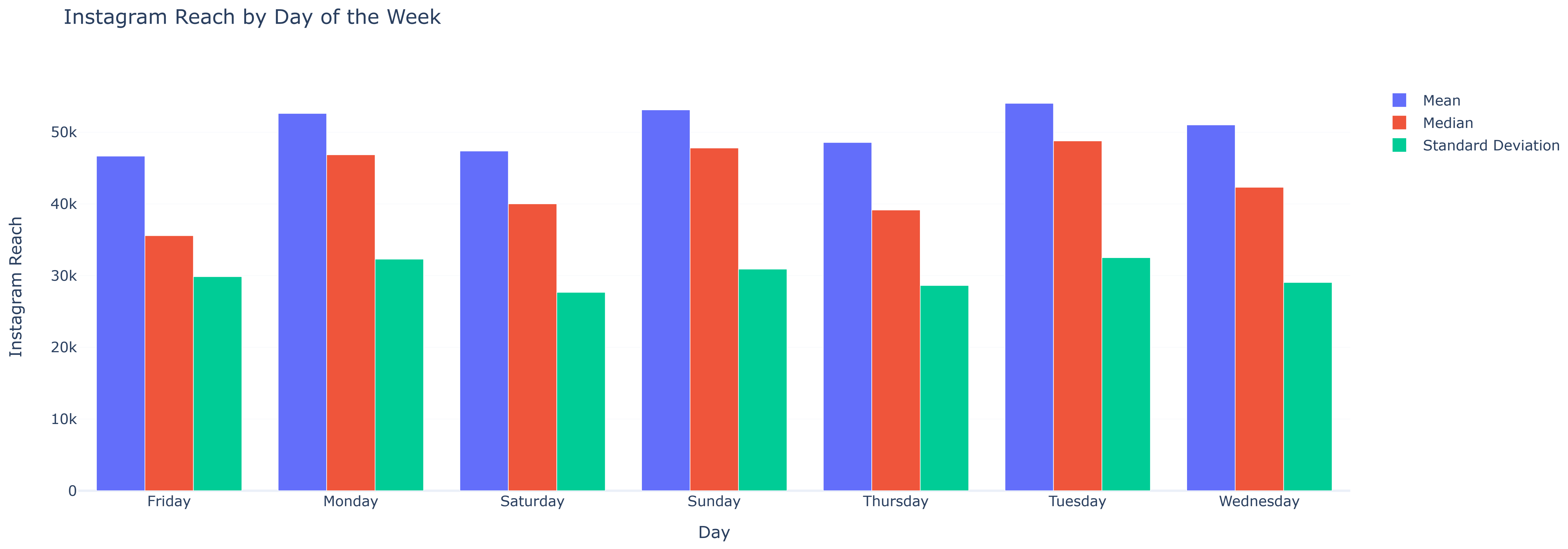
```
day_stats = data.groupby('Day')['Instagram reach'].agg(['mean', 'median', 'std']).reset_index()
print(day_stats)
```

```
         Day          mean   median           std
0     Friday  46666.849057  35574.0  29856.943036
1     Monday  52621.692308  46853.0  32296.071347
2   Saturday  47374.750000  40012.0  27667.043634
3     Sunday  53114.173077  47797.0  30906.162384
4   Thursday  48570.923077  39150.0  28623.220625
5    Tuesday  54030.557692  48786.0  32503.726482
6  Wednesday  51017.269231  42320.5  29047.869685
```

Lets visualize the above information using a bar chart (For each day of the week)

In [10]:

```
fig = go.Figure()
fig.add_trace(go.Bar(x=day_stats['Day'],
                     y=day_stats['mean'],
                     name='Mean'))
fig.add_trace(go.Bar(x=day_stats['Day'],
                     y=day_stats['median'],
                     name='Median'))
fig.add_trace(go.Bar(x=day_stats['Day'],
                     y=day_stats['std'],
                     name='Standard Deviation'))
fig.update_layout(title='Instagram Reach by Day of the Week',
                  xaxis_title='Day',
                  yaxis_title='Instagram Reach')
fig.show()
```



Conclusion

1. From the above we can ascertain that the best days to post content are on Tuesday, Sunday and Monday respectively according to the bar chart above.
2. The month of September is the month where content gets the highest reach as seen from the line chart above.
3. Therefore Tuesdays, Sundays and Mondays of september should be maximized to get enough reach that in turn will hopefully generate desired end results.

. . . .