In [14]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import re
from collections import Counter
n=0
dates_f=[]
time_f=[]
msg=[]
with open('chat6.txt', encoding="utf-8") as fp:
    while n<15000:
        text=fp.readline()
        date=re.compile('\d{2}/\d{2}/\d{2}.\s\d{2}:\d{2}\s-\s')
        dates=date.findall(text)
        n=n+1
        for x in dates:
            dates_f2=x[:-10]

            d4= dates_f2.split('/')
            d5=d4[::-1]
            d6='-'.join(d5)


            dates_f.append(d6)
            time_f.append(x[-8:-3])
            zz=re.split(date,text)
            msg.append(zz[-1])


print('anas')
```

```
anas
```

In [15]:

```
name_f=[]
msg_f=[]
media=[]
for x in msg:
    end=re.search(':|removed|added|changed|left',x).end()
    start=re.search(':|removed|added|changed|left',x).start()
    name_f.append(x[:start].strip())
    msg_f.append(x[end:-1].strip())
    if '<Media omitted>'in x:
        media.append(1)
    else:
        media.append(0)

df=pd.DataFrame({
    'Date':dates_f,
    'Time': time_f,
    'Name':name_f,
    'Message':msg_f,
    'Media':media
})
def Number_of_active_days_out_of_87(df):
        name=list(df["Name"].unique())
        ind=df.groupby(["Name","Date"]).count()
        indw=df.groupby(["Name","Date"]).count().reset_index(level=[0,1])
        fig=plt.figure(figsize=(20,5))
        plt.bar(indw.groupby("Name").count().index,indw.groupby("Name").count()["Date"])
        plt.xticks(rotation=90)
        plt.xlabel("Name",fontsize=24)
        plt.ylabel("No. of active days out of 86")
        print(indw.groupby("Name").count()["Date"])

Time=pd.DataFrame({
    'time':df['Time'],
})
def msg_counts(df):
    xx=df.groupby('Name').count()
    xx.unstack()
    yy=xx['Message'].sort_values(ascending=False)
    print(yy)
    fig=plt.figure(figsize=(20,5))
    plt.bar(yy.index,yy)
    plt.xticks(rotation=90)
    plt.xlabel("Name")
    plt.ylabel("Total No. of masseges sent")
Names=list(df['Name'].unique())
def date_wise_plot_by_eachone(data):
        for x in Names:
            xx=data.groupby(['Date','Name']).count().unstack()
            yy=xx.fillna(0)['Message']
            plt.figure(n)
            fig=plt.figure(figsize=(20,5))
            plt.bar(yy.index,yy[x])
            plt.xticks(rotation=90,fontsize=10)
            plt.ylabel('no. of messages each day')
            plt.title(x+'ऀ')
```

```
            oo=df.groupby(['Date','Name'])["Message"].count().unstack().fillna(0)
            pp=oo.loc[:].cumsum()
            plt.figure(figsize=(26,6))
            plt.plot(pp[x],color='red' ,label="no. of msg each day")
            plt.plot(oo[x] ,label="comulative no. of msg ")
            plt.xticks(rotation=90, )
            plt.title(x+'ह')
            plt.legend()
```

In [16]:

```
Names=list(df['Name'].unique())
def date_wise_plot_by_eachone(data):
        xx=data.groupby(['Date','Name']).count().unstack()
        yy=xx.fillna(0)['Message']
        for x in Names:
            fig=plt.figure(figsize=(22,5))
            plt.bar(yy.index,yy[x])
            plt.xticks(rotation=90,fontsize=12)
            plt.xlabel(x,fontsize=26)
            plt.ylabel('no. of messages')
```

In [17]:

```
def Date_wise_overall_msg(df):
        fig=plt.figure(figsize=(22,5))
        plt.bar(df.groupby('Date').count().index,df.groupby('Date').count()["Message"])
        plt.xticks(rotation=90)
        plt.xlabel('x')
        plt.ylabel('no. of messages')

        plt.figure(2)
        fig=plt.figure(figsize=(22,5))
        plt.plot(df.groupby('Date').count()["Message"].cumsum())
        plt.xticks(rotation=90)
        plt.xlabel('x')
        plt.ylabel('no. of messages')
        plt.title("No of messages sent in group")
```

In [18]:

```
def highest_msgsent_on_anyday_by_individuals(df):
        top=df.groupby(["Date","Name"]).count()["Message"].unstack().fillna(0)
        xx=top.loc[:].max().sort_values(ascending=False)
        xx.plot(kind="bar",figsize=(25,6))
        print(xx)


def media_count(df):
    return df[df["Media"]==1].sort()

def number_of_active_people_on_any_given_day(df):
        p=df.groupby(["Date","Name"]).count()
        q=p["Message"].unstack().count(axis=1)
        print(q.sort_values(ascending=False).head(15))
        q.plot(figsize=(24,6))


media=df[df["Media"]==1]
media_count=media.groupby("Name").count()["Message"]
yy=media_count/media_count.sum()*100
Names=list(df['Name'].unique())
def date_wise_plot_by_eachone(data):
        for x in Names:
            xx=data.groupby(['Date','Name']).count().unstack()
            yy=xx.fillna(0)['Message']
            plt.figure(n)
            fig=plt.figure(figsize=(20,5))
            plt.bar(yy.index,yy[x])
            plt.xticks(rotation=90,fontsize=10)
            plt.ylabel('no. of messages')
            plt.title(x)


            oo=df.groupby(['Date','Name'])["Message"].count().unstack().fillna(0)
            pp=oo.loc[:].cumsum()
            plt.figure(figsize=(26,6))
            plt.plot(pp[x],color='red' ,label= "comulative no. of msg ")
            plt.plot(oo[x] ,label="no. of msg each day")
            plt.xticks(rotation=90, )
            plt.title(x)
            plt.legend()
```

In [23]:

```
```

Out[23]:

	Date	Time	Name	Message	Media
0	20-05-26	16:33	Karamveer	the subject from "JNV Batch 2008-15 wale" to "...	0
1	20-05-26	16:36	Karamveer	this group's icon	0
2	20-05-26	16:41	Arpit Singh	Neelgiri Wali and +91 94585 19878	0
3	20-05-26	16:36	Arpit Singh	Ye new group ku bana h	0
4	20-05-26	16:37	Rehman	Isme bijnore waale nahi hain	0
...	...	...	...	...	...
9890	20-08-31	12:55	Vivek 2	<Media omitted>	1
9891	20-08-31	12:55	Ankit	😂😂	0
9892	20-08-31	19:32	Gagan	if it is asked "akal badi ya bhains " , then a...	0
9893	20-08-31	19:33	Vivek 2	But there is a mistake bro	0
9894	20-08-31	19:35	Vivek 2	U should replace last bhains word by ankit□□♂	0

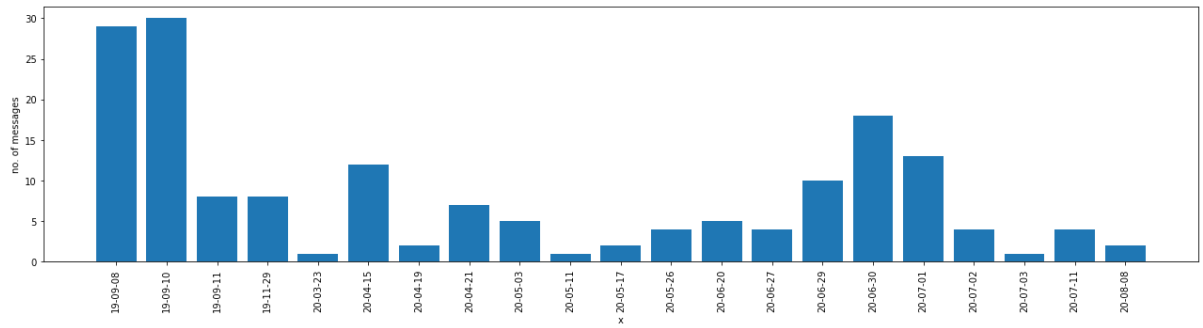
9895 rows × 5 columns

In [8]:

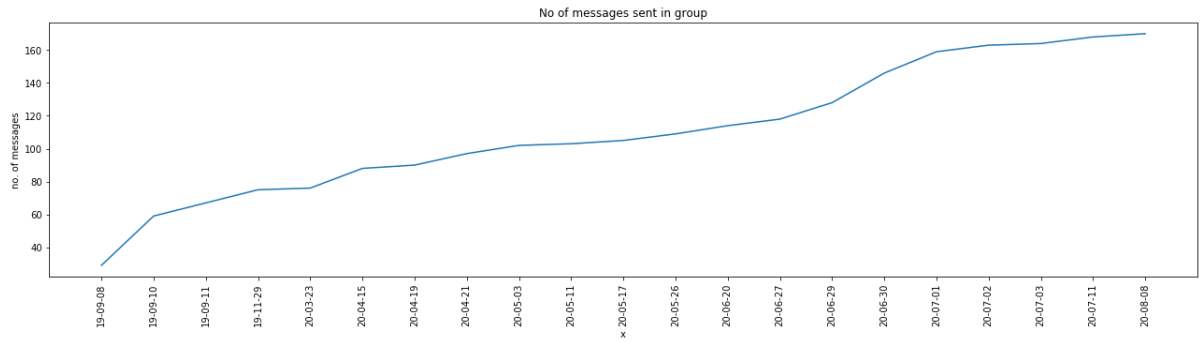
```
def comulative_msg(df):
        comulative=df.groupby(["Date","Name"])["Message"].count().unstack().fillna(0).cums
um()


        comulative.plot(kind="line",figsize=(20,6))
        plt.title("msg count vs date")
        plt.legend(fontsize=10,loc=2)
        plt.xticks(rotation=90)
```

In [23]:

```
Date_wise_overall_msg(df)
```



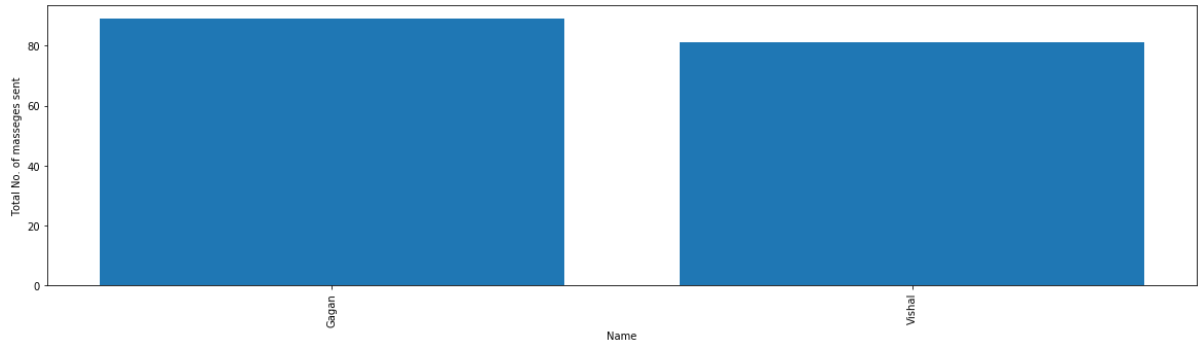
<Figure size 432x288 with 0 Axes>



In [24]:

```
msg_counts(df) # total message sent by each one
```

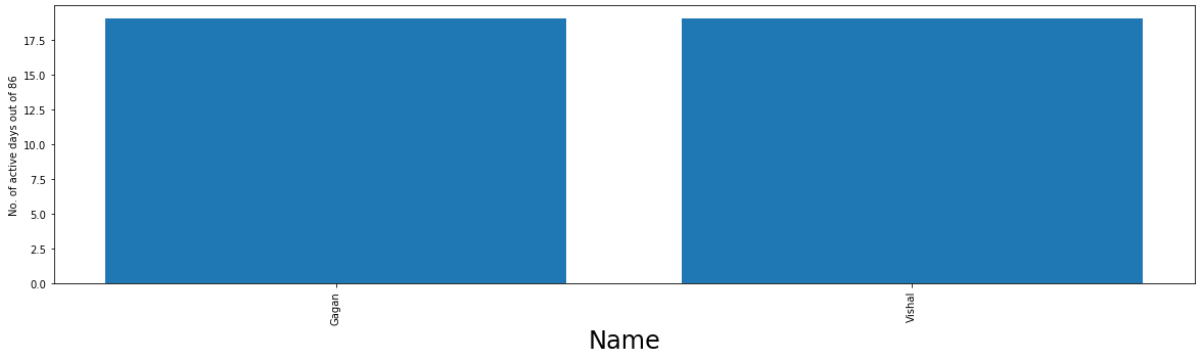
```
Name
Gagan     89
Vishal    81
Name: Message, dtype: int64
```



In [26]:

```
Number_of_active_days_out_of_87(df)
```

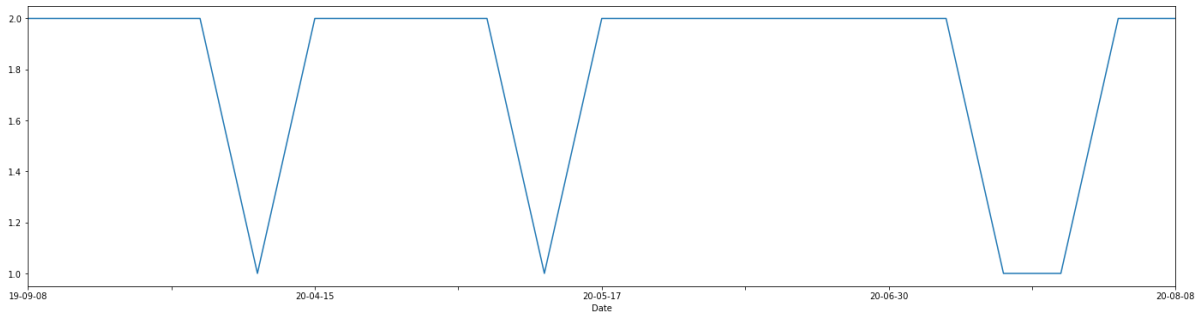
```
Name
Gagan     19
Vishal    19
Name: Date, dtype: int64
```



In [27]:

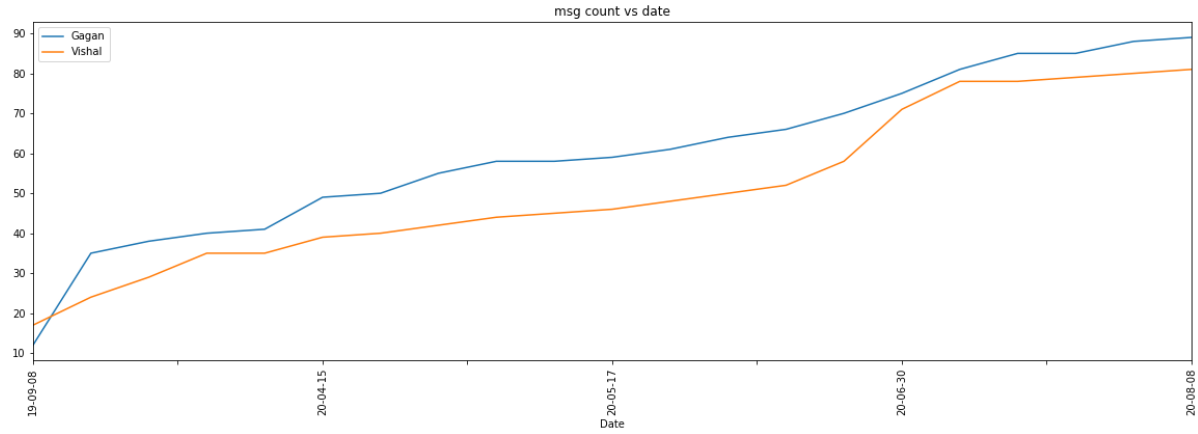
```
number_of_active_people_on_any_given_day(df) #list of days where highest number of people
 were active date  formate=2020-month-date
```

```
Date
20-08-08    2
20-07-11    2
19-09-10    2
19-09-11    2
19-11-29    2
20-04-15    2
20-04-19    2
20-04-21    2
20-05-03    2
20-05-17    2
20-05-26    2
20-06-20    2
20-06-27    2
20-06-29    2
20-06-30    2
dtype: int64
```



In [28]:

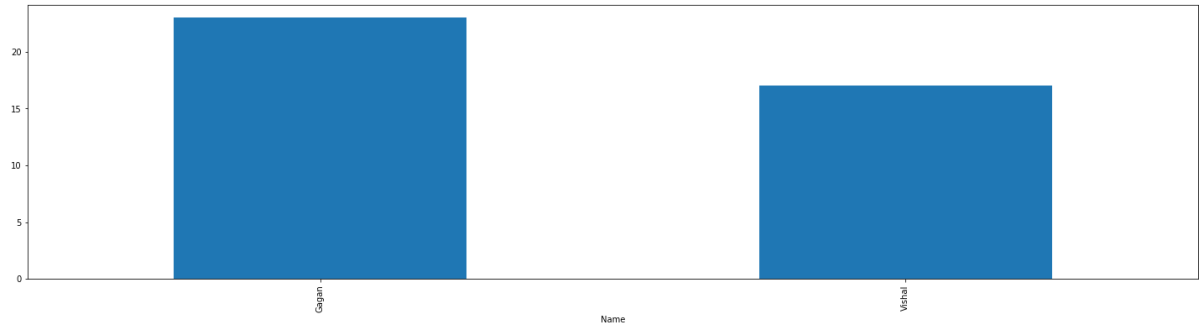
```
comulative_msg(df)
```



In [29]:

```
highest_msgsent_on_anyday_by_individuals(df)
```

```
Name
Gagan     23.0
Vishal    17.0
dtype: float64
```



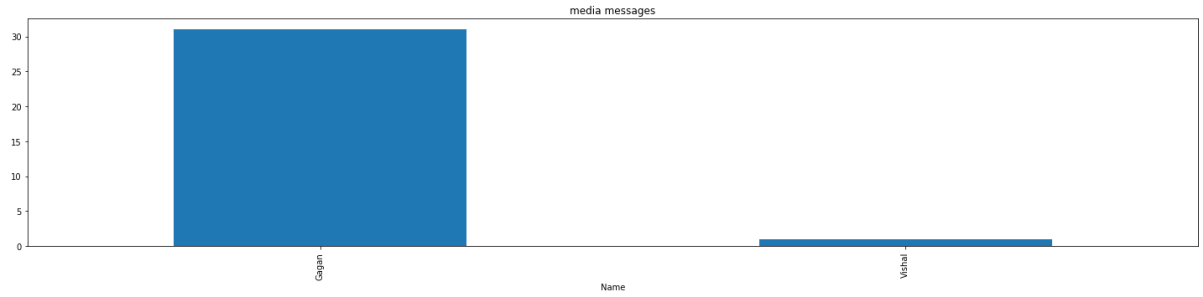
In [30]:

```
print(f'Number of Media msg sent by each {media_count.sort_values(ascending=False)} ')
sorted_media=media_count.sort_values(ascending=False)
sorted_media.sort_values(ascending=False).plot(kind="bar",figsize=(25,5))
plt.title("media messages")
```

```
Number of Media msg sent by each Name
Gagan     31
Vishal     1
Name: Message, dtype: int64
```

Out[30]:

```
Text(0.5, 1.0, 'media messages')
```

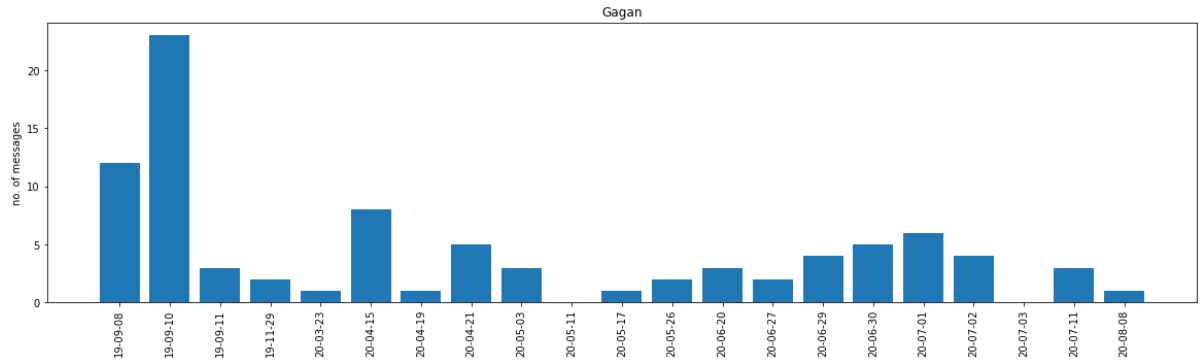


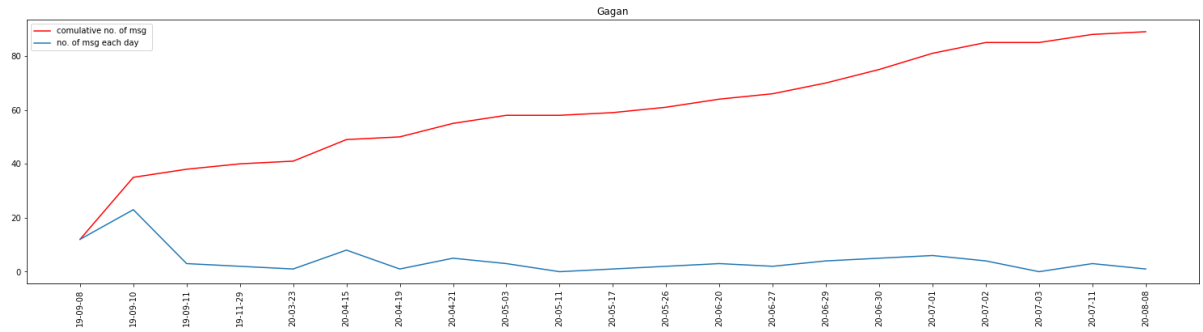
In [ ]:

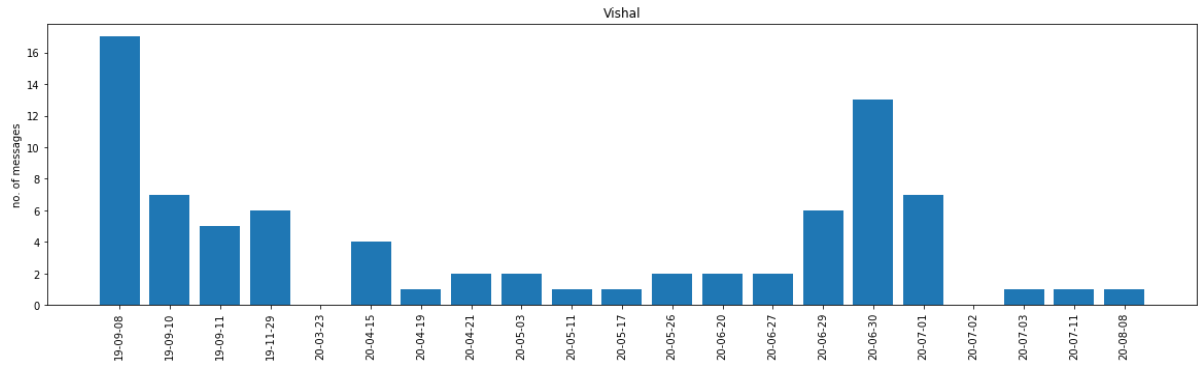
In [31]:

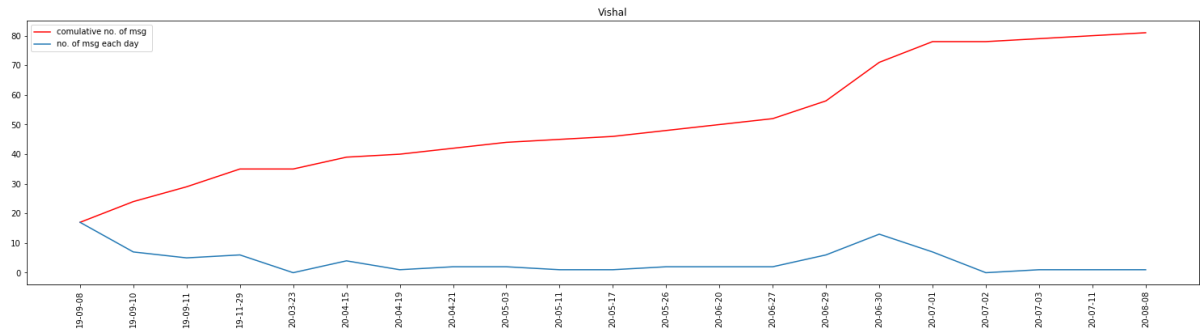
```
date_wise_plot_by_eachone(df)
```

```
<Figure size 432x288 with 0 Axes>
```



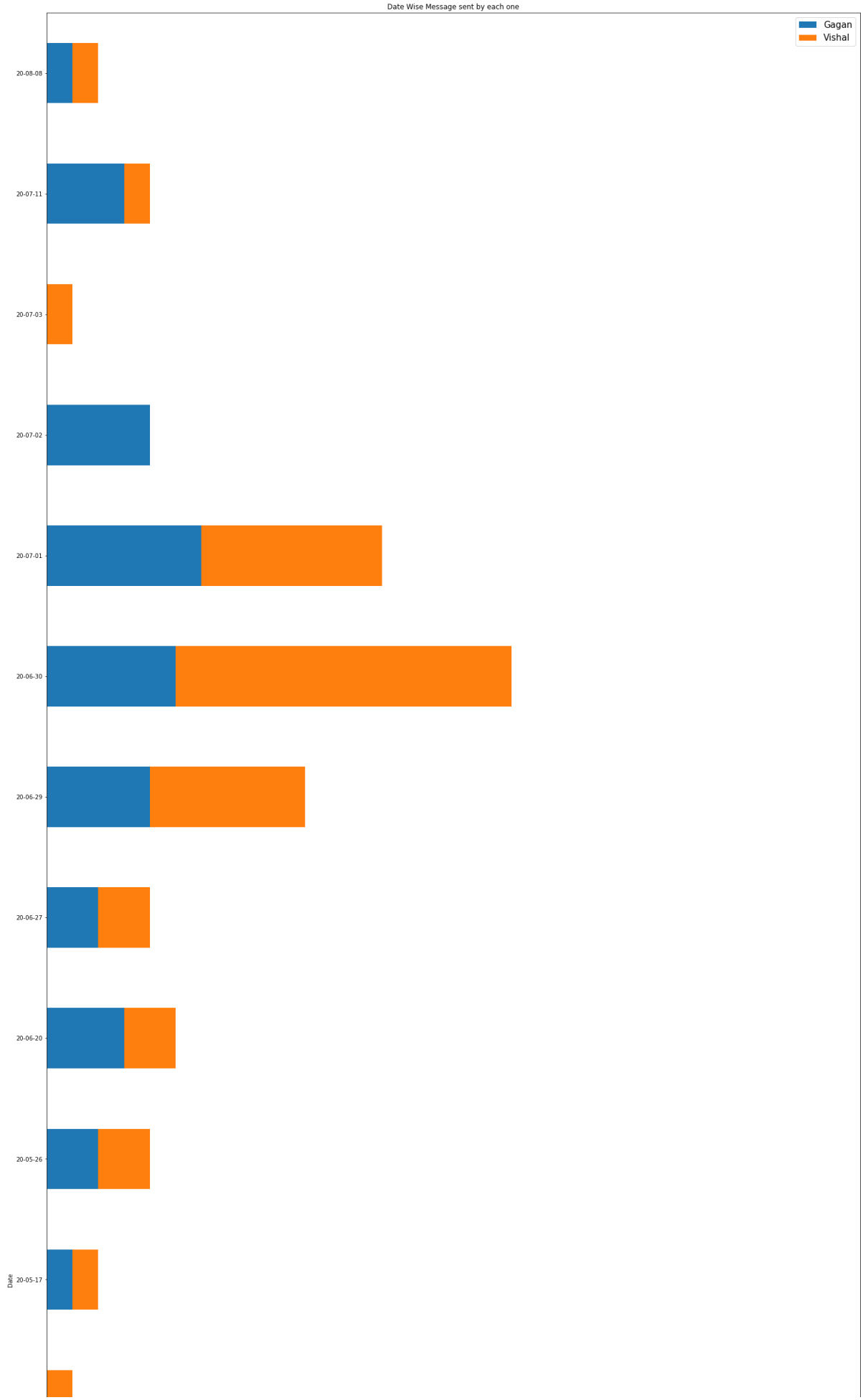


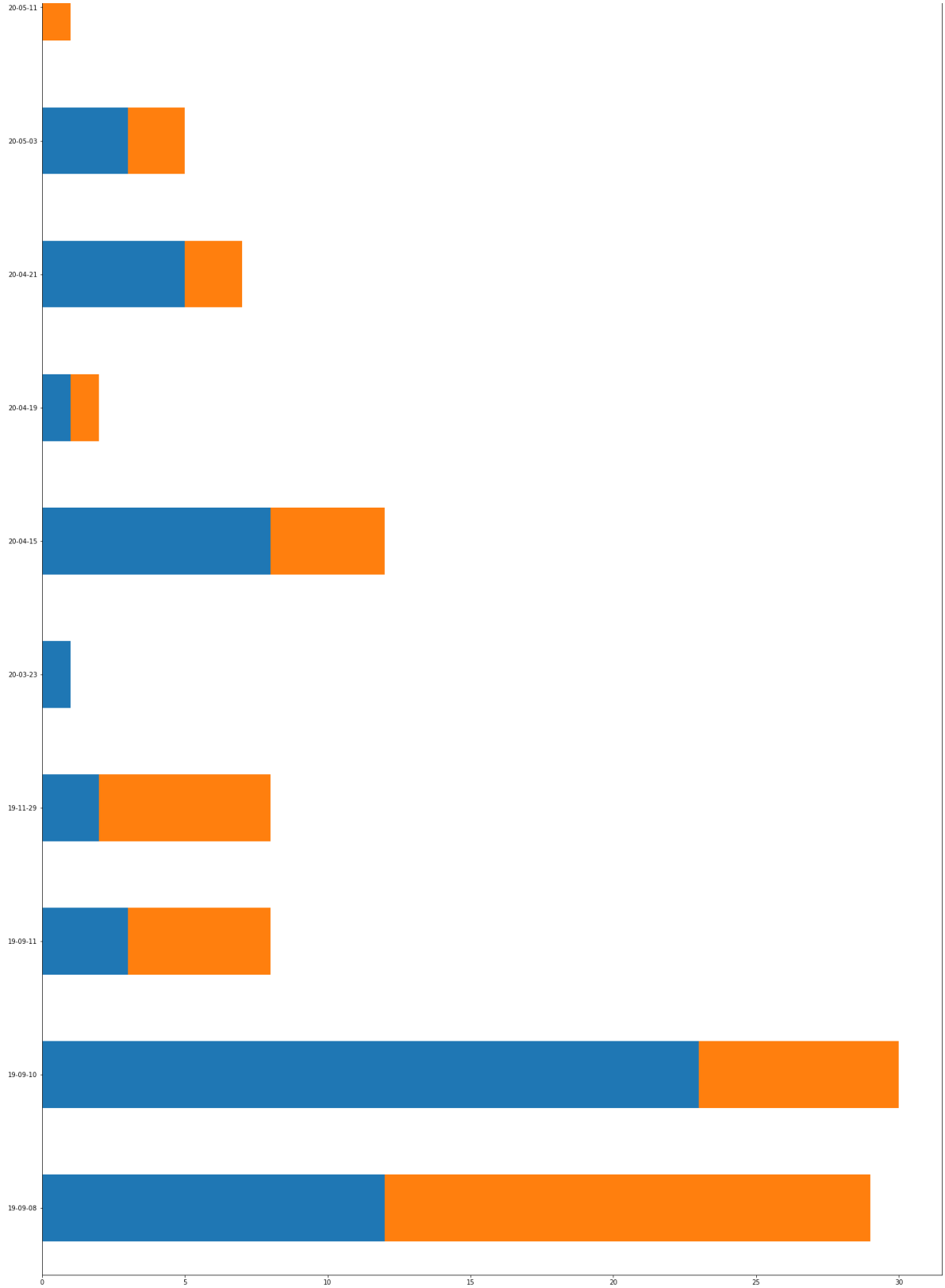




In [32]:

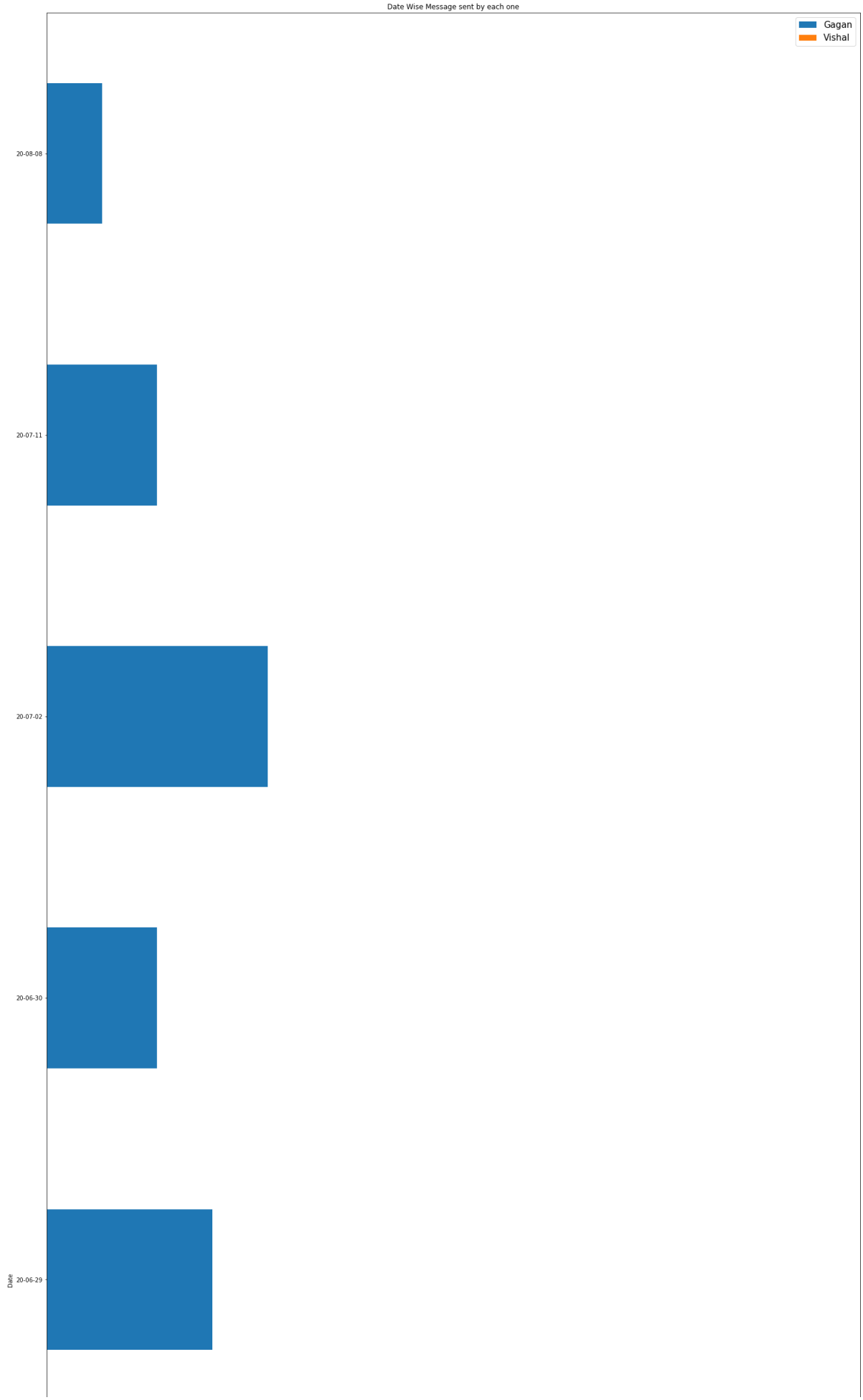
```
def stacked_plot(data):
    data.groupby(["Date","Name"])["Message"].count().unstack().fillna(0).plot(kind='barh',
stacked=True,figsize=(25,80))
    plt.title("Date Wise Message sent by each one")
    plt.legend(prop={'size': 15})
stacked_plot(df)
```

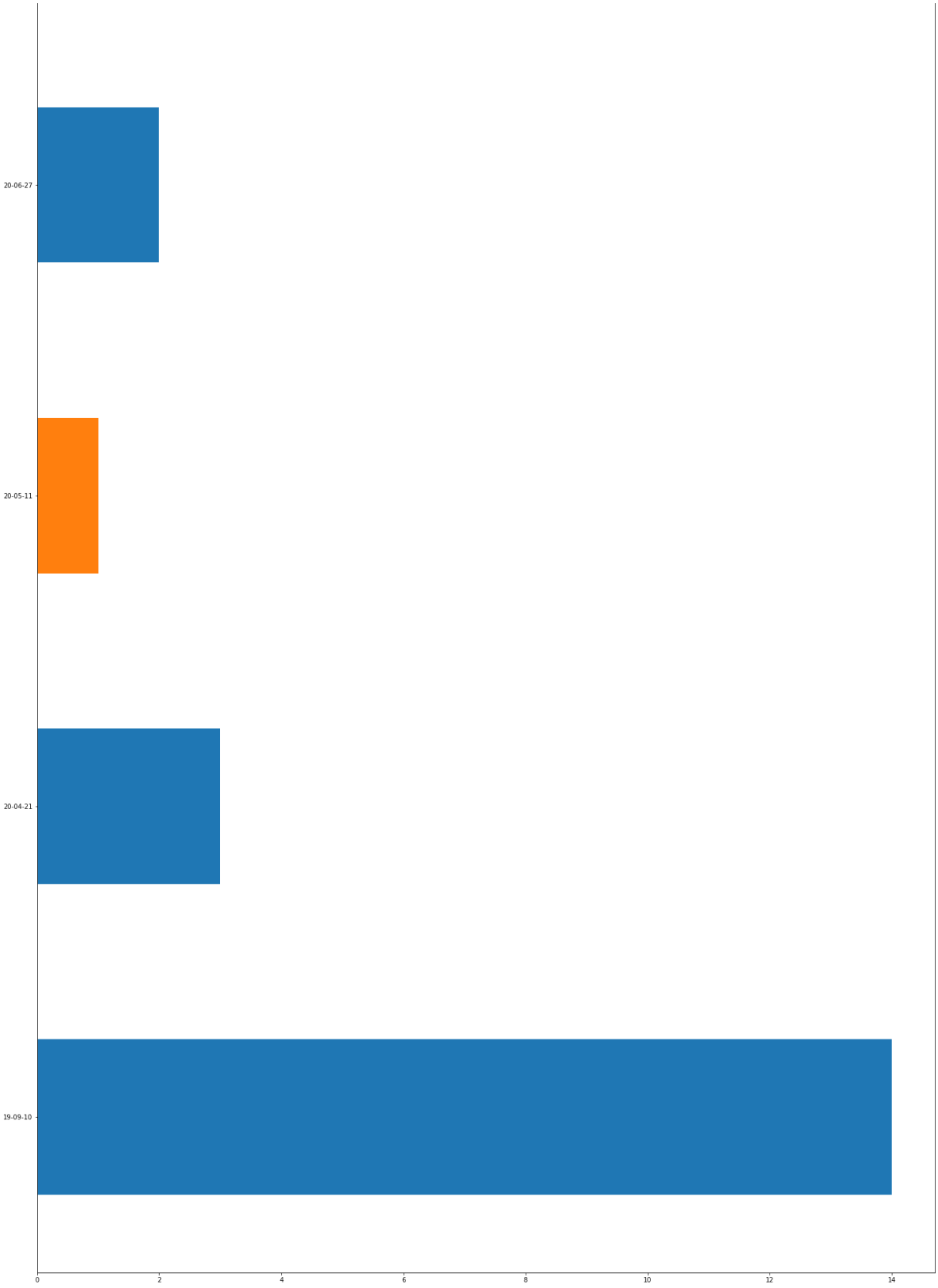

Date Wise Message sent by each one
Gagan
Vishal
20-08-08
20-07-11
20-07-03
20-07-02
20-07-01
20-06-30
20-06-29
20-06-27
20-06-20
20-05-26
20-05-17
Date


20-05-11
20-05-03
20-04-21
20-04-19
20-04-15
20-03-23
19-11-29
19-09-11
19-09-10
19-09-08
0
5
10
15
20
25
30

In [33]:

```
stacked_plot(media)
```


Date Wise Message sent by each one
Gagan
Vishal
20-08-08
20-07-11
20-07-02
20-06-30
20-06-29
Date



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [200]:

```
20-06-18
```

Out[200]:

```
Date
20-05-26    False
20-05-27    False
20-05-28    False
20-05-29    False
20-05-30    False
20-05-31    False
20-06-01    False
20-06-09    False
20-06-10    False
20-06-11    False
20-06-12    False
20-06-13    False
20-06-14    False
20-06-15    False
20-06-16    False
20-06-17    False
20-06-18     True
20-06-19    False
20-06-20    False
Name: Kalavati, dtype: bool
```

In [ ]: