

```
import pandas as pd
```

```
df=pd.read_csv('archive/covid_19_india.csv')
df
```

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3
...
18105	18106	2021-08-11	8:00 AM	Telangana	-	-	638410	3831	650353
18106	18107	2021-08-11	8:00 AM	Tripura	-	-	77811	773	80660
18107	18108	2021-08-11	8:00 AM	Uttarakhand	-	-	334650	7368	342462
18108	18109	2021-08-11	8:00 AM	Uttar Pradesh	-	-	1685492	22775	1708812
18109	18110	2021-08-11	8:00 AM	West Bengal	-	-	1506532	18252	1534999

18110 rows x 9 columns

```
df['State/UnionTerritory'].unique()
```

```
array(['Kerala', 'Telangana', 'Delhi', 'Rajasthan', 'Uttar Pradesh',
       'Haryana', 'Ladakh', 'Tamil Nadu', 'Karnataka', 'Maharashtra',
       'Punjab', 'Jammu and Kashmir', 'Andhra Pradesh', 'Uttarakhand',
       'Odisha', 'Jharkhand', 'West Bengal', 'Chhattisgarh',
       'Chandigarh', 'Gujarat', 'Himachal Pradesh', 'Madhya Pradesh',
       'Bihar', 'Manipur', 'Mizoram', 'Andaman and Nicobar Islands',
       'Goa', 'Unassigned', 'Assam', 'Jharkhand', 'Arunachal Pradesh',
       'Tripura', 'Nagaland', 'Meghalaya',
       'Dadra and Nagar Haveli and Daman and Diu',
       'Cases being reassigned to states', 'Sikkim', 'Daman & Diu',
       'Lakshadweep', 'Telangana', 'Dadra and Nagar Haveli', 'Bihar****',
       'Madhya Pradesh****', 'Himanchal Pradesh', 'Karnataka',
       'Maharashtra****'], dtype=object)
```

```
In [ ]:
```

```
maha = df[df['State/UnionTerritory']=='Maharashtra']
```

```
maha.dropna()
```

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
76	77	2020-03-09	6:00 PM	Maharashtra	2	0	0	0	2
91	92	2020-03-10	6:00 PM	Maharashtra	5	0	0	0	5
97	98	2020-03-11	6:00 PM	Maharashtra	2	0	0	0	2
120	121	2020-03-12	6:00 PM	Maharashtra	11	0	0	0	11
133	134	2020-03-13	6:00 PM	Maharashtra	14	0	0	0	14
...
17950	17951	2021-08-07	8:00 AM	Maharashtra	-	-	6130137	133717	6341759
17986	17987	2021-08-08	8:00 AM	Maharashtra	-	-	6139493	133845	6347820
18022	18023	2021-08-09	8:00 AM	Maharashtra	-	-	6144388	133996	6353328
18058	18059	2021-08-10	8:00 AM	Maharashtra	-	-	6151956	134064	6357833
18094	18095	2021-08-11	8:00 AM	Maharashtra	-	-	6159676	134201	6363442

520 rows x 9 columns

```
maha.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 520 entries, 76 to 18094
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  ---                ---
0   Sno                   520 non-null   int64
1   Date                  520 non-null   object
2   Time                  520 non-null   object
3   State/UnionTerritory  520 non-null   object
4   ConfirmedIndianNational  520 non-null   object
5   ConfirmedForeignNational  520 non-null   object
6   Cured                 520 non-null   int64
7   Deaths               520 non-null   int64
8   Confirmed             520 non-null   int64
dtypes: int64(4), object(5)
memory usage: 40.6+ KB
```

```
maha.describe()
```

	Sno	Cured	Deaths	Confirmed
count	520.000000	5.200000e+02	520.000000	5.200000e+02
mean	8799.448077	1.959164e+06	45648.907692	2.156714e+06
std	5315.265136	1.974083e+06	38067.407872	2.076120e+06
min	77.000000	0.000000e+00	0.000000	2.000000e+00
25%	4190.000000	1.567582e+05	11127.500000	2.821208e+05
50%	8733.500000	1.652928e+06	46638.000000	1.782284e+06
75%	13388.000000	2.439400e+06	55018.250000	2.868141e+06
max	18095.000000	6.159676e+06	134201.000000	6.363442e+06

```
maha.max()
```

Sno	18095
Date	2021-08-11
Time	9:30 PM
State/UnionTerritory	Maharashtra
ConfirmedIndianNational	86
ConfirmedForeignNational	3
Cured	6159676
Deaths	134201
Confirmed	6363442
dtype:	object

```
maha['Deaths'].max() # maximum number of people death in one day
```

```
134201
```

```
maha['Confirmed'].max() # maximum number of people Confirmed in one day
```

```
6363442
```

```
maha['Cured'].max() # maximum number of people Cured in one day
```

```
6159676
```

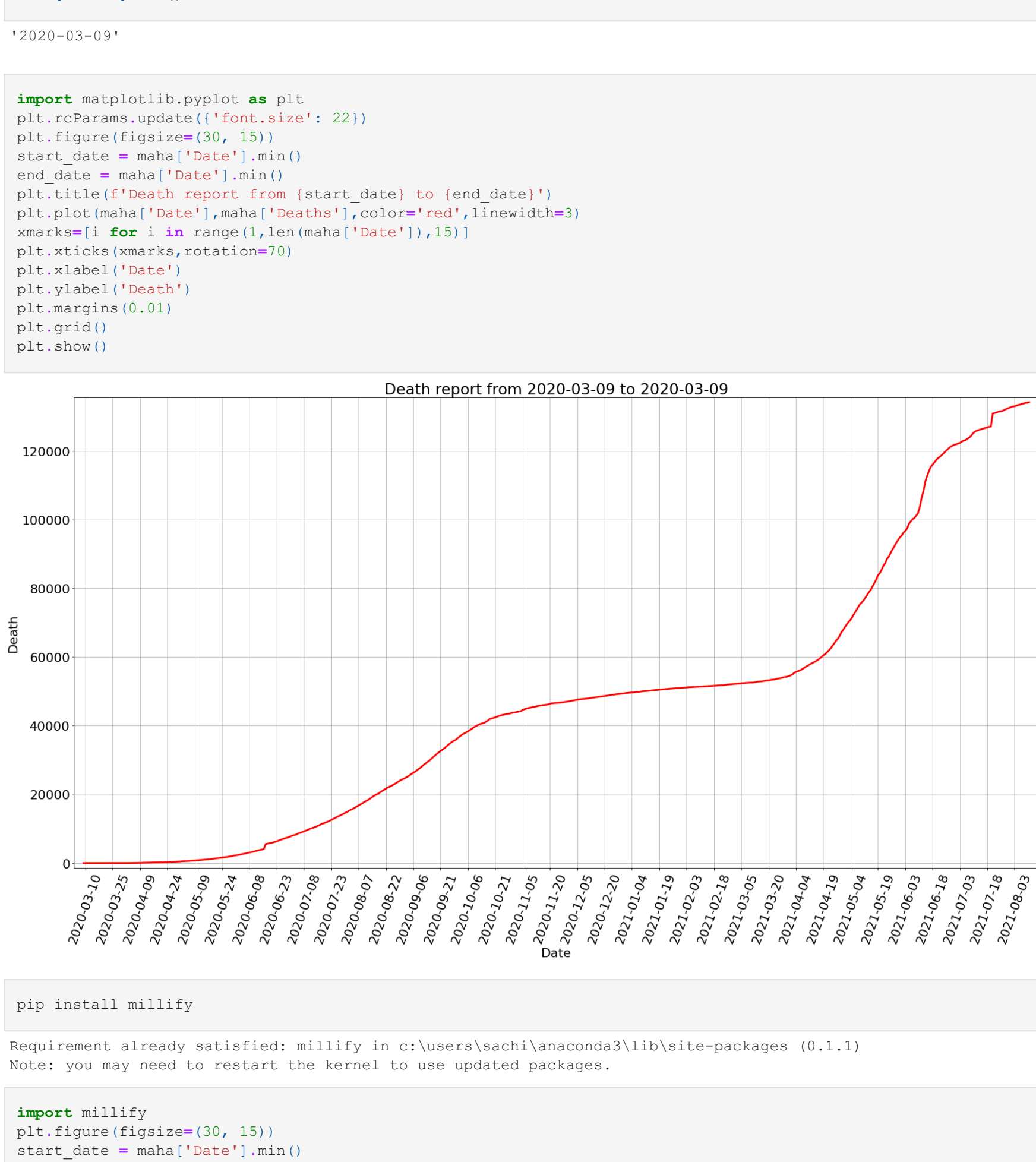
```
maha['Date'].max() # data from date
```

```
'2021-08-11'
```

```
maha['Date'].min() # data to date
```

```
'2020-03-09'
```

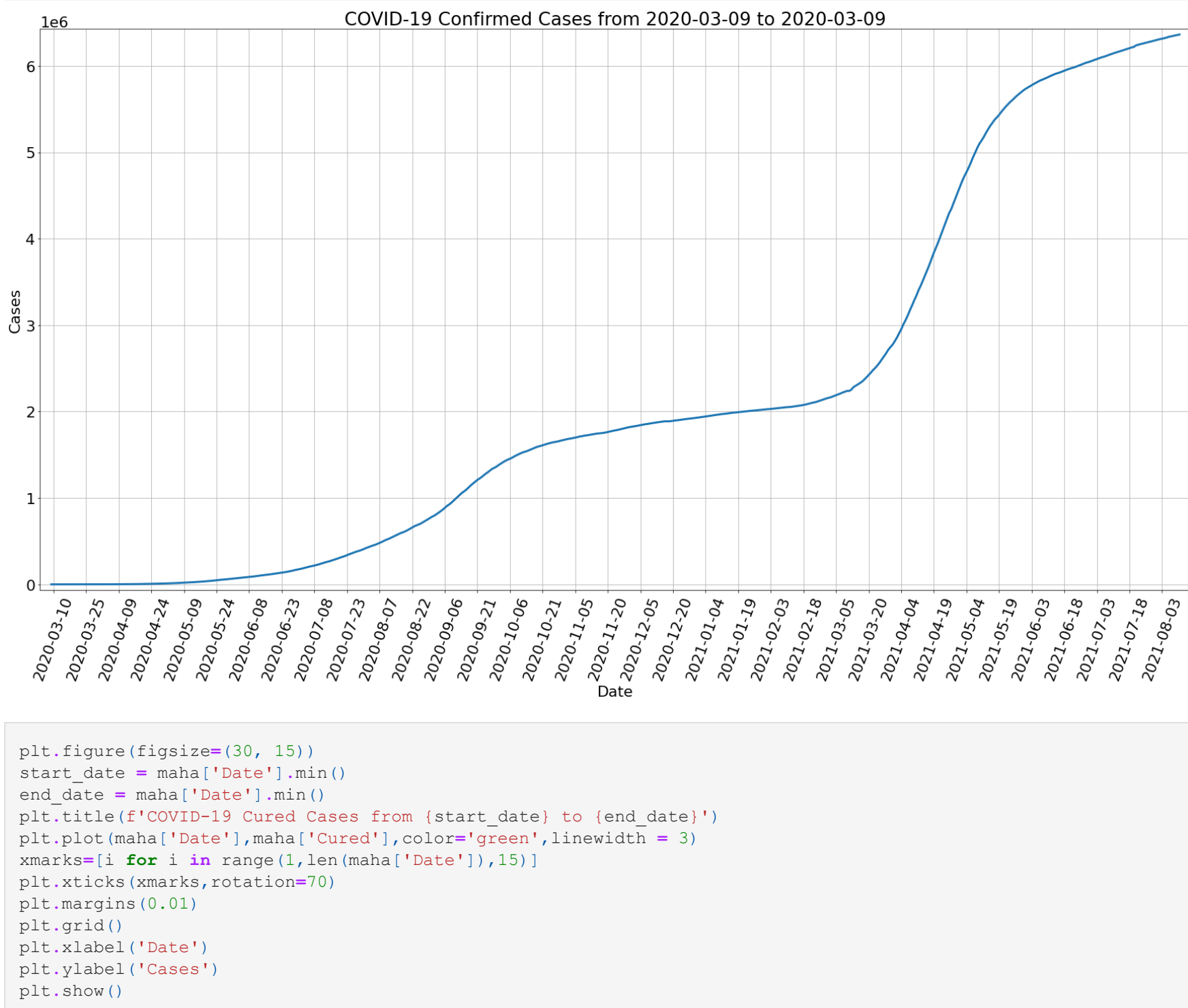
```
import matplotlib.pyplot as plt
plt.rcParams.update({'font.size': 22})
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].max()
plt.title(f'Death report from (start_date) to (end_date)')
plt.plot(maha['Date'],maha['Deaths'],color='red',linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.xlabel('Date')
plt.ylabel('Death')
plt.margins(0.01)
plt.grid()
plt.show()
```



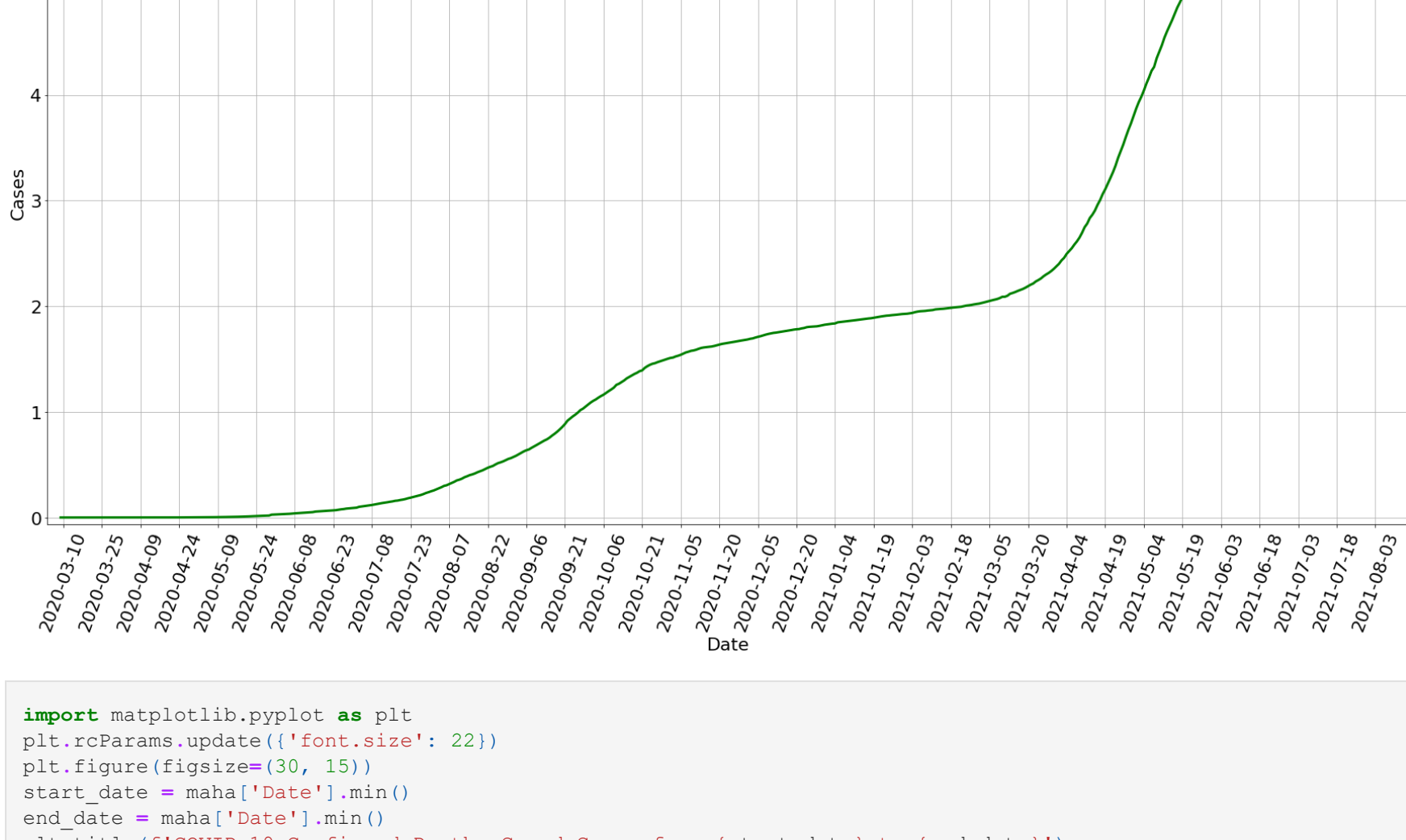
```
pip install millify
```

Requirement already satisfied: millify in c:\users\sachi\anaconda3\lib\site-packages (0.1.1)
Note: you may need to restart the kernel to use updated packages.

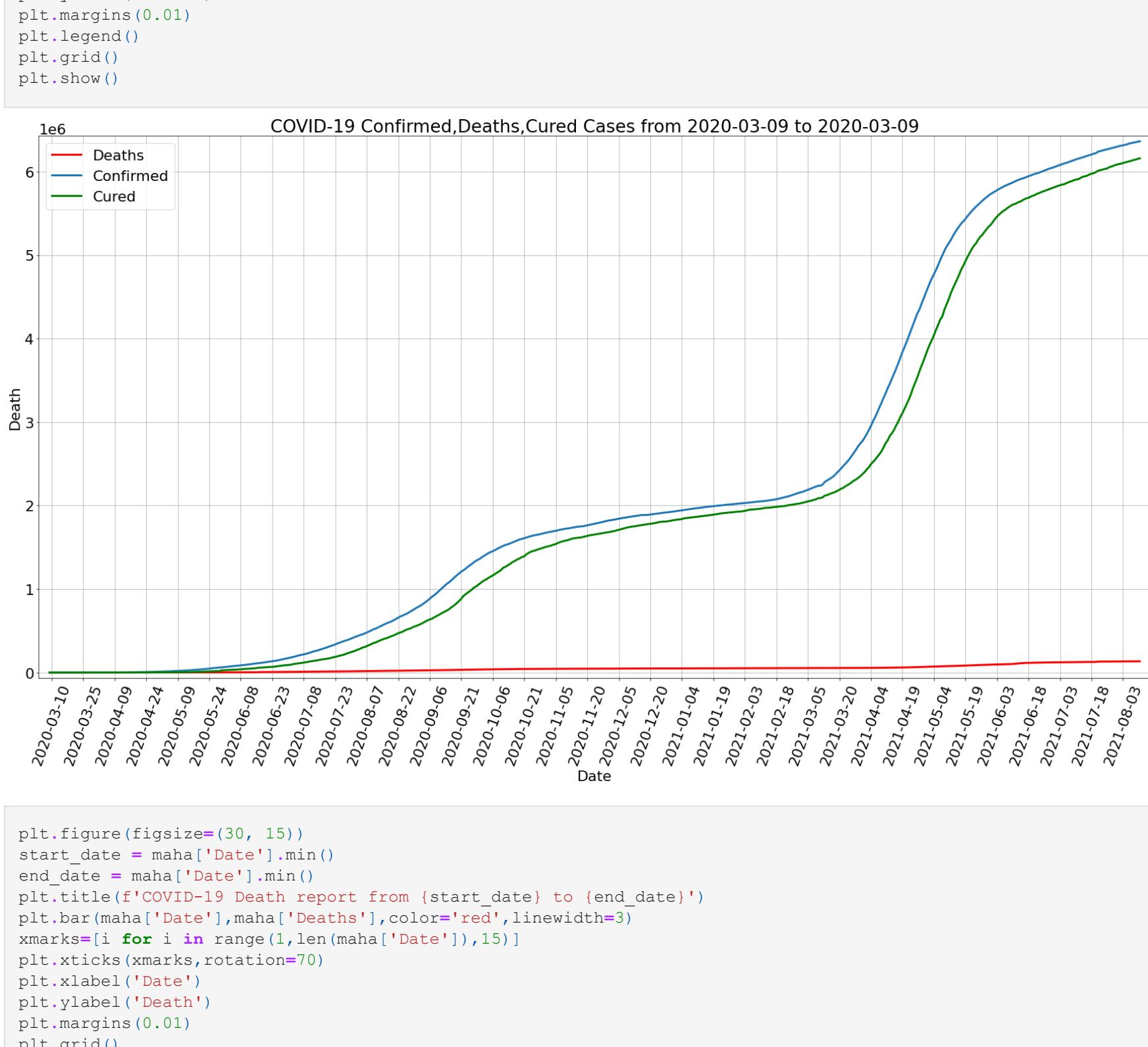
```
import millify
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Confirmed Cases from (start_date) to (end_date)')
plt.plot(maha['Date'],maha['Confirmed'],linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.margins(0.01)
plt.grid()
plt.xlabel('Date')
plt.ylabel('Cases')
plt.show()
```



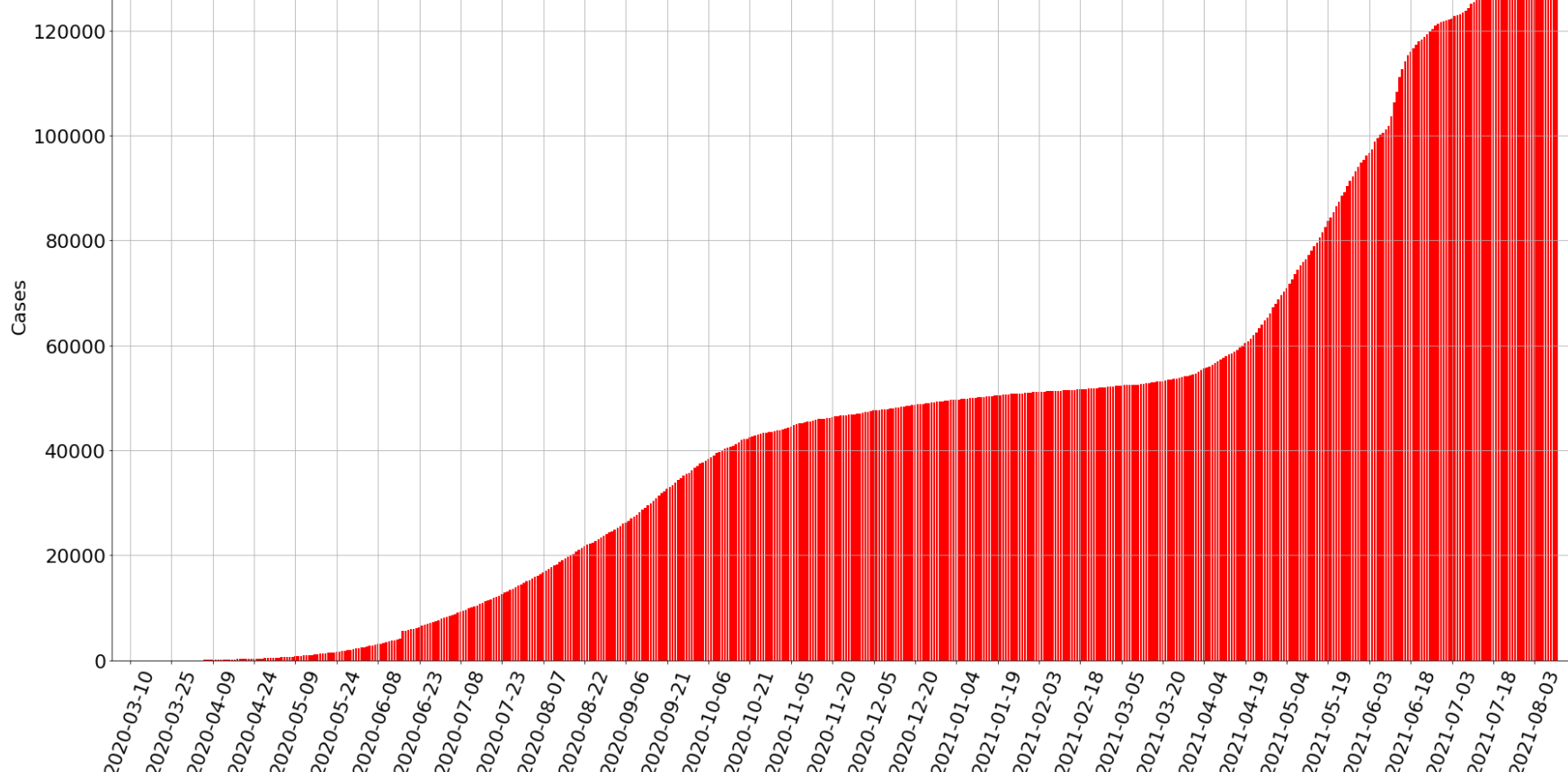
```
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Cured Cases from (start_date) to (end_date)')
plt.plot(maha['Date'],maha['Cured'],color='green',linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.margins(0.01)
plt.grid()
plt.xlabel('Date')
plt.ylabel('Cases')
plt.show()
```



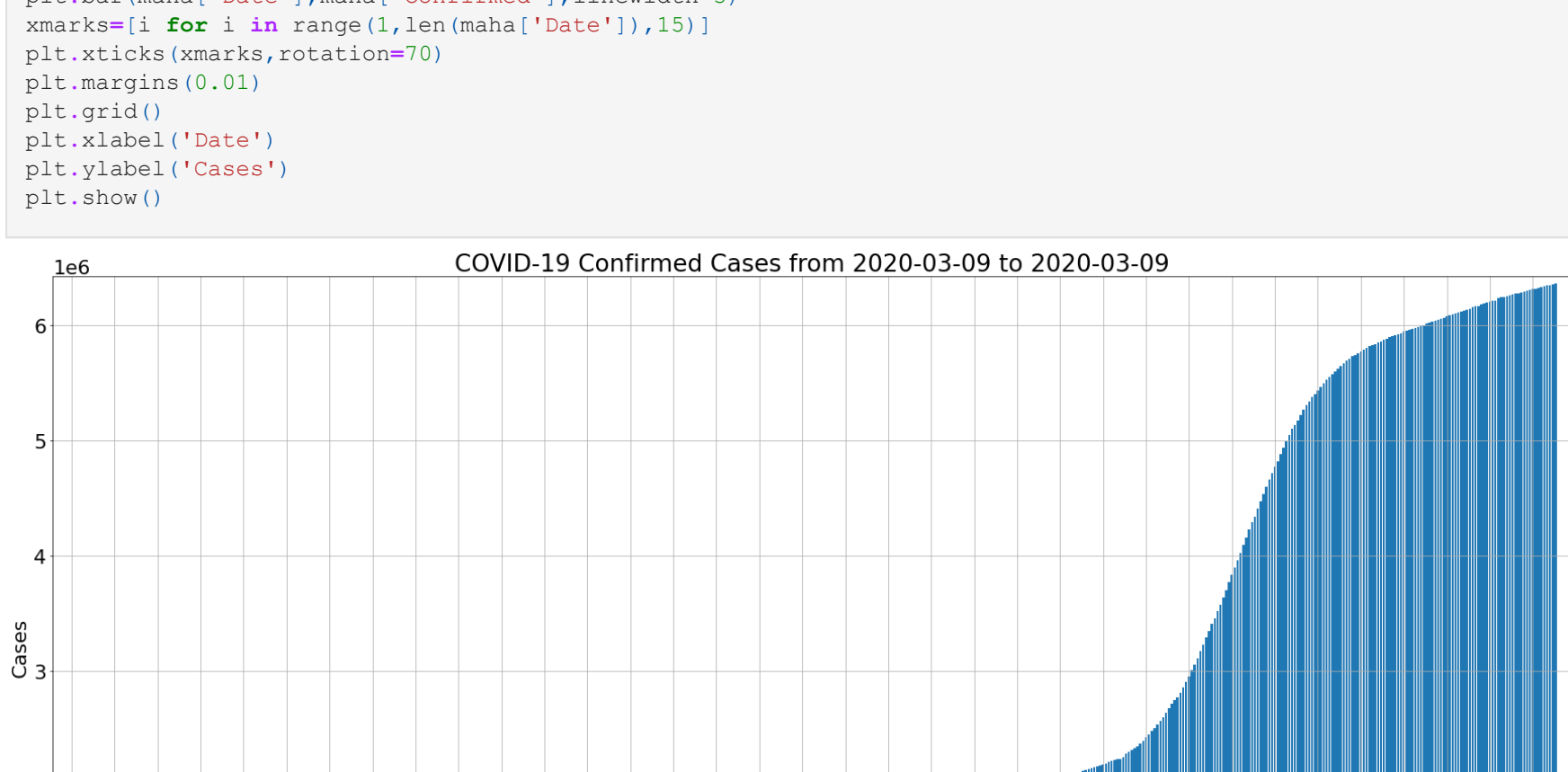
```
import matplotlib.pyplot as plt
plt.rcParams.update({'font.size': 22})
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Confirmed,Deaths,Cured Cases from (start_date) to (end_date)')
plt.plot(maha['Date'],maha['Deaths'],color='red',linewidth=3, label='Deaths' )
plt.plot(maha['Date'],maha['Confirmed'],color='green',linewidth=3, label='Confirmed')
plt.plot(maha['Date'],maha['Cured'],color='green',linewidth=3, label='Cured')
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.xlabel('Date')
plt.ylabel('Death')
plt.margins(0.01)
plt.legend()
plt.grid()
plt.show()
```



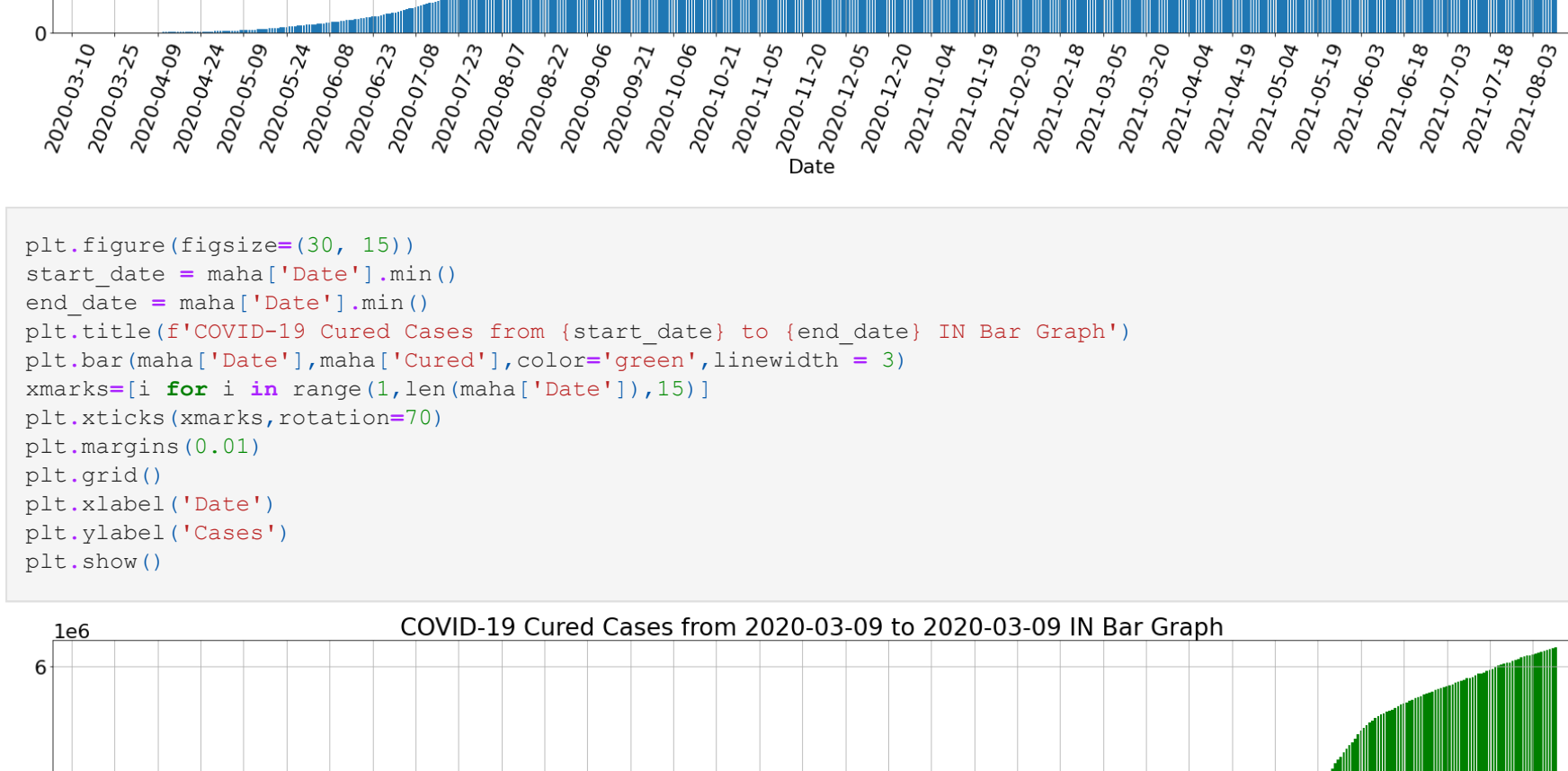
```
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Death report from (start_date) to (end_date)')
plt.bar(maha['Date'],maha['Deaths'],color='red',linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.xlabel('Date')
plt.ylabel('Death')
plt.margins(0.01)
plt.grid()
plt.xlabel('Date')
plt.ylabel('Cases')
plt.show()
```



```
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Confirmed Cases from (start_date) to (end_date)')
plt.bar(maha['Date'],maha['Confirmed'],color='red',linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.margins(0.01)
plt.grid()
plt.xlabel('Date')
plt.ylabel('Cases')
plt.show()
```

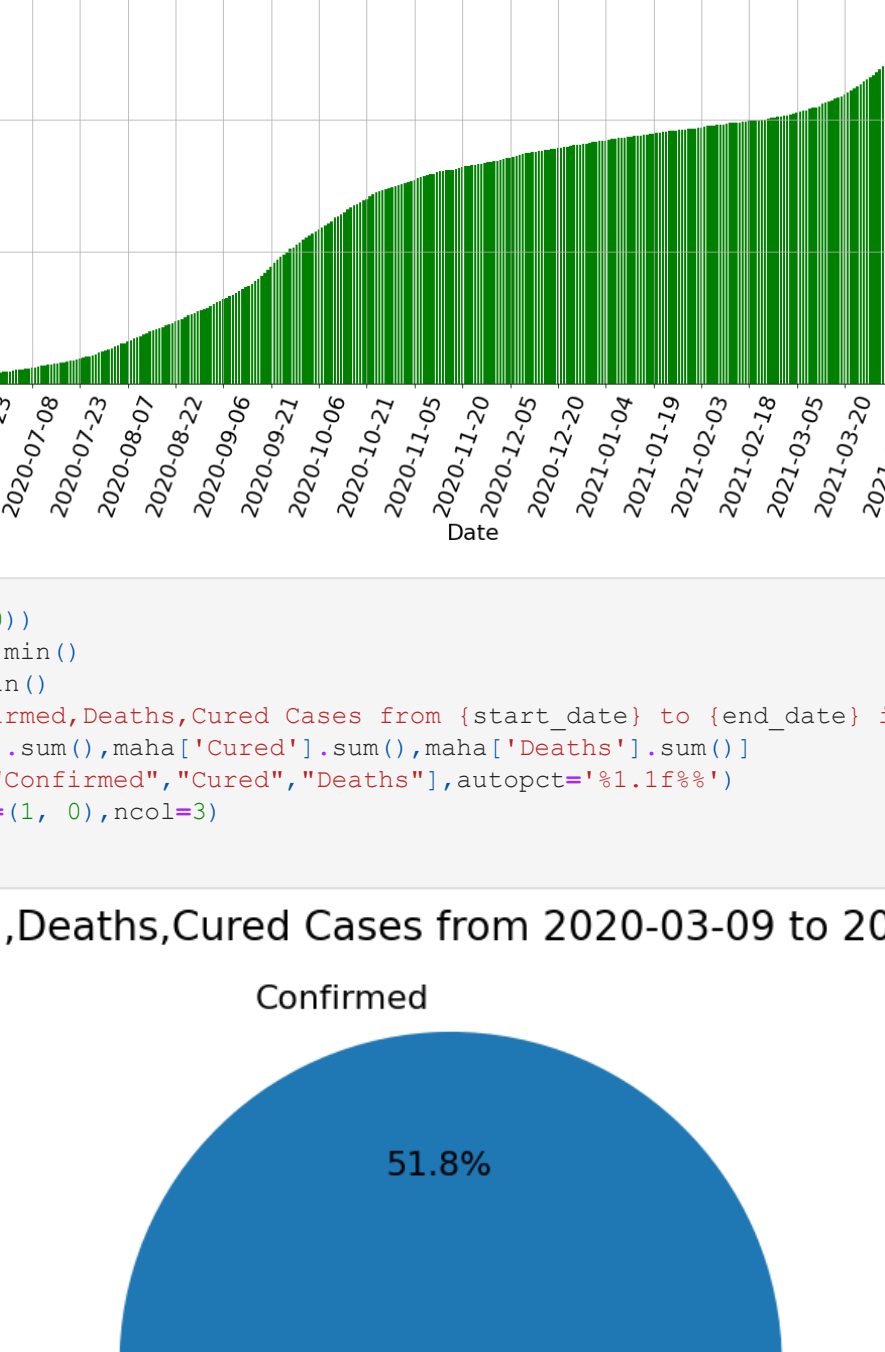


```
plt.figure(figsize=(30, 15))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Cured Cases from (start_date) to (end_date) IN Bar Graph')
plt.bar(maha['Date'],maha['Cured'],color='green',linewidth=3)
xmarks=[i for i in range(1,len(maha['Date']),15)]
plt.xticks(xmarks,rotation=70)
plt.margins(0.01)
plt.grid()
plt.xlabel('Date')
plt.ylabel('Cases')
plt.show()
```



```
plt.figure(figsize=(20, 10))
start_date = maha['Date'].min()
end_date = maha['Date'].min()
plt.title(f'COVID-19 Confirmed,Deaths,Cured Cases from (start_date) to (end_date) in Pie chart')
total = [maha['Confirmed'].sum(),maha['Cured'].sum(),maha['Deaths'].sum()]
plt.pie(total, labels = ["Confirmed","Cured","Deaths"],autopct='%1.1f%%')
plt.legend(bbox_to_anchor=(1, 0),ncol=3)
plt.show()
```

COVID-19 Confirmed,Deaths,Cured Cases from 2020-03-09 to 2020-03-09 in Pie chart



```
In [ ]:
```