

```

pip install numpy pandas matplotlib seaborn plotly

Requirement already satisfied: numpy in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (1.26.4)
Requirement already satisfied: pandas in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (2.2.2)
Requirement already satisfied: matplotlib in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (3.9.0)
Requirement already satisfied: seaborn in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (0.13.1)
Requirement already satisfied: plotly in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (5.22.0)
Requirement already satisfied: python-dateutil<3.0, >2.8.2 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from pandas) (2.9.0.post8)
Requirement already satisfied: pytz>=2020.3 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from pandas) (2024.1)
Requirement already satisfied: fonttools<4.22.0, >4.21.0 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from matplotlib (3.9.0)) (4.22.0)
Requirement already satisfied: kiwisolver>=3.1 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from matplotlib (3.9.0)) (3.0.2)
Requirement already satisfied: packaging>=20.0 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from matplotlib (3.9.0)) (24.0)
Requirement already satisfied: pillow>=8.2.0 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from matplotlib (3.9.0)) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from matplotlib (3.9.0)) (3.1.2)
Requirement already satisfied: six>=1.5 in c:\users\laasya\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil<3.0, >2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.

In [2]:
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt
import calendar
import plotly.graph_objects as go

import warnings
warnings.filterwarnings("ignore")
import matplotlib inline

In [3]:
df = pd.read_csv("unemployment_rate upto_31_2020.csv")
df.head()

Out[4]:
   Region  Date  Frequency  Estimated Unemployment Rate (%)  Estimated Employed  Estimated Labour Participation Rate (%)  Region.1  longitude  latitude
0  Andhra Pradesh  31-01-2020  M  5.48  16635535  41.02  South  15.9129  79.74
1  Andhra Pradesh  29-02-2020  M  5.83  16545652  40.90  South  15.9129  79.74
2  Andhra Pradesh  31-03-2020  M  5.79  15881197  39.18  South  15.9129  79.74
3  Andhra Pradesh  30-04-2020  M  20.51  11336911  33.10  South  15.9129  79.74
4  Andhra Pradesh  31-05-2020  M  17.43  12988845  36.46  South  15.9129  79.74

In [5]:
df.tail()

Out[6]:
   Region  Date  Frequency  Estimated Unemployment Rate (%)  Estimated Employed  Estimated Labour Participation Rate (%)  Region.1  longitude  latitude
262  West Bengal  30-06-2020  M  7.29  30726310  40.39  East  22.9666  87.855
263  West Bengal  31-07-2020  M  6.83  35372006  46.17  East  22.9666  87.855
264  West Bengal  31-08-2020  M  14.87  33298644  47.48  East  22.9666  87.855
265  West Bengal  30-09-2020  M  9.35  35707239  47.73  East  22.9666  87.855
266  West Bengal  31-10-2020  M  9.98  33962549  45.63  East  22.9666  87.855

In [8]:
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 267 entries, 0 to 266
Data columns (total 9 columns):
 #   Column  Non-Null Count  Dtype
---  --
0   state  267 non-null     object
1   date  267 non-null     object
2   frequency  267 non-null     object
3   estimated unemployment rate (%)  267 non-null     float64
4   estimated employed  267 non-null     int64
5   estimated labour participation rate (%)  267 non-null     float64
6   region.1  267 non-null     object
7   longitude  267 non-null     float64
8   latitude  267 non-null     float64
dtypes: float64(4), int64(1), object(4)
memory usage: 18.0+ kb

In [7]:
df.columns = ['state','date','frequency','estimated unemployment rate','estimated employed','estimated labour participation rate','region','longitude','latitude']
df.head()

Out[7]:
   state  date  frequency  estimated unemployment rate  estimated employed  estimated labour participation rate  region  longitude  latitude
0  Andhra Pradesh  31-01-2020  M  5.48  16635535  41.02  South  15.9129  79.74
1  Andhra Pradesh  29-02-2020  M  5.83  16545652  40.90  South  15.9129  79.74
2  Andhra Pradesh  31-03-2020  M  5.79  15881197  39.18  South  15.9129  79.74
3  Andhra Pradesh  30-04-2020  M  20.51  11336911  33.10  South  15.9129  79.74
4  Andhra Pradesh  31-05-2020  M  17.43  12988845  36.46  South  15.9129  79.74

In [9]:
df.shape
Out[9]:
(267, 9)

In [9]:
df.columns
Out[9]:
Index(['state', 'date', 'frequency', 'estimated unemployment rate', 'estimated employed', 'estimated labour participation rate', 'region', 'longitude', 'latitude'],
      dtype='object')

In [10]:
df.describe()

Out[10]:
   estimated unemployment rate  estimated employed  estimated labour participation rate  longitude  latitude
count  267.000000  2.670000e+02  267.000000  267.000000  267.000000
mean  12.236929  1.396211e+07  41.681573  22.626048  80.524245
std  10.403283  1.336632e+07  7.845419  6.270731  5.631738
min  0.500000  1.175400e+05  16.770000  10.850500  71.192400
25%  4.845000  2.839000e+06  37.265000  18.112400  76.085600
50%  6.650000  9.732417e+06  40.390000  23.610200  79.019000
75%  16.755000  2.167896e+07  44.055000  27.274600  85.279900
max  75.850000  5.943376e+07  69.690000  33.778200  92.937600

In [11]:
df.isnull().sum()
Out[11]:
state 0
date 0
frequency 0
estimated unemployment rate 0
estimated employed 0
estimated labour participation rate 0
region 0
longitude 0
latitude 0
dtypes: int64 0

In [12]:
df.duplicated().any()
Out[12]:
False

In [13]:
df.state.value_counts()

Out[13]:
state
Andhra Pradesh 10
Assam 10
Uttarakhand 10
Uttar Pradesh 10
Tamil Nadu 10
Telangana 10
Tamil Nadu 10
Rajasthan 10
Punjab 10
Puducherry 10
Odisha 10
Meghalaya 10
Maharashtra 10
Madhya Pradesh 10
Kerala 10
Karnataka 10
Jharkhand 10
Himachal Pradesh 10
Haryana 10
Gujarat 10
Goa 10
Delhi 10
Chhattisgarh 10
Bihar 10
West Bengal 10
Jammu & Kashmir 9
Sikkim 9
Name: count, dtype: int64

In [14]:
df['date'] = pd.to_datetime(df['date'],dayfirst = True)
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 267 entries, 0 to 266
Data columns (total 9 columns):
 #   Column  Non-Null Count  Dtype
---  --
0   state  267 non-null     object
1   date  267 non-null     datetime64[ns]
2   frequency  267 non-null     object
3   estimated unemployment rate  267 non-null     float64
4   estimated employed  267 non-null     int64
5   estimated labour participation rate  267 non-null     float64
6   region  267 non-null     object
7   longitude  267 non-null     float64
8   latitude  267 non-null     float64
dtypes: datetime64[ns](1), float64(4), int64(1), object(3)
memory usage: 18.0+ kb

In [15]:
df['month_int'] = df['date'].dt.month
df.head()

Out[15]:
   state  date  frequency  estimated unemployment rate  estimated employed  estimated labour participation rate  region  longitude  latitude  month_int
0  Andhra Pradesh  2020-01-31  M  5.48  16635535  41.02  South  15.9129  79.74  1
1  Andhra Pradesh  2020-02-29  M  5.83  16545652  40.90  South  15.9129  79.74  2
2  Andhra Pradesh  2020-03-31  M  5.79  15881197  39.18  South  15.9129  79.74  3
3  Andhra Pradesh  2020-04-30  M  20.51  11336911  33.10  South  15.9129  79.74  4
4  Andhra Pradesh  2020-05-31  M  17.43  12988845  36.46  South  15.9129  79.74  5

In [16]:
df['month'] = df['month_int'].apply(lambda x: calendar.month_abbr[x])
df.head()

Out[16]:
   state  date  frequency  estimated unemployment rate  estimated employed  estimated labour participation rate  region  longitude  latitude  month_int  month
0  Andhra Pradesh  2020-01-31  M  5.48  16635535  41.02  South  15.9129  79.74  1  Jan
1  Andhra Pradesh  2020-02-29  M  5.83  16545652  40.90  South  15.9129  79.74  2  Feb
2  Andhra Pradesh  2020-03-31  M  5.79  15881197  39.18  South  15.9129  79.74  3  Mar
3  Andhra Pradesh  2020-04-30  M  20.51  11336911  33.10  South  15.9129  79.74  4  Apr
4  Andhra Pradesh  2020-05-31  M  17.43  12988845  36.46  South  15.9129  79.74  5  May

In [17]:
data = df.groupby('month')[['estimated
```



```
fig.update_layout(axis={'categoryorder':'total ascending'})
fig.show()
```