OASIS INFOBYTE INTERNSHIP

TASK-2

AUTHOR- KRIPA JOSE

UNEMPLOYMENT ANALYSIS WITH PYTHON

Importing libraries and data cleaning

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
```

Read the dataset into a pandas dataframe.

In [2]:
```
df=pd.read_csv('Unemployment_Rate_upto_11_2020.csv')
df.head()
```

Out[2]:

	Region	Date	Frequency	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude	latitu
0	Andhra Pradesh	31-01-2020	M	5.48	16635535	41.02	South	15.9129	79
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129	79
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129	79
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129	79
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129	79

Check the shape, data types, and summary statistics of the dataset.

In [3]:
```
df.shape
```

Out[3]:
```
(267, 9)
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 267 entries, 0 to 266
Data columns (total 9 columns):
 #   Column                                   Non-Null Count  Dtype  
---  ------                                   --------------  -----  
 0   Region                                   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: 18.9+ KB
```

Check for missing values and duplicate rows in the dataset.

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
Region                                   0
 Date                                    0
 Frequency                               0
 Estimated Unemployment Rate (%)         0
 Estimated Employed                      0
 Estimated Labour Participation Rate (%)    0
Region.1                                 0
longitude                                0
latitude                                 0
dtype: int64
```

In [6]:
```
df.duplicated().sum()
```

Out[6]:
```
0
```

Formatting columns

In [7]:
```
df.columns = ['States', 'Date', 'Frequency', 'Estimated Unemployment Rate', 'E
              'Estimated Labour Participation Rate', 'Region', 'longitude', 'l
df['Frequency'] = df['Frequency'].astype('category')
df['Region'] = df['Region'].astype('category')
```

observing basic statistical values

In [8]:
```
df[['Estimated Unemployment Rate', 'Estimated Employed','Estimated Labour Part
```

Out[8]:

	count	mean	std	min	25%	50%	
Estimated Unemployment Rate	267.0	1.223693e+01	1.080328e+01	0.50	4.845	9.65	1.675500
Estimated Employed	267.0	1.396211e+07	1.336632e+07	117542.00	2838930.500	9732417.00	2.187869
Estimated Labour Participation Rate	267.0	4.168157e+01	7.845419e+00	16.77	37.265	40.39	4.405500

checking the unique values of 'States'

In [9]:
```
df['States'].value_counts()
```

Out[9]:
```
Andhra Pradesh      10
Assam               10
Uttarakhand         10
Uttar Pradesh       10
Tripura             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
```

checking the unique values of 'Region'

In [10]:
```
df['Region'].value_counts()
```

Out[10]:
```
North        79
South        60
West         50
East         40
Northeast    38
Name: Region, dtype: int64
```

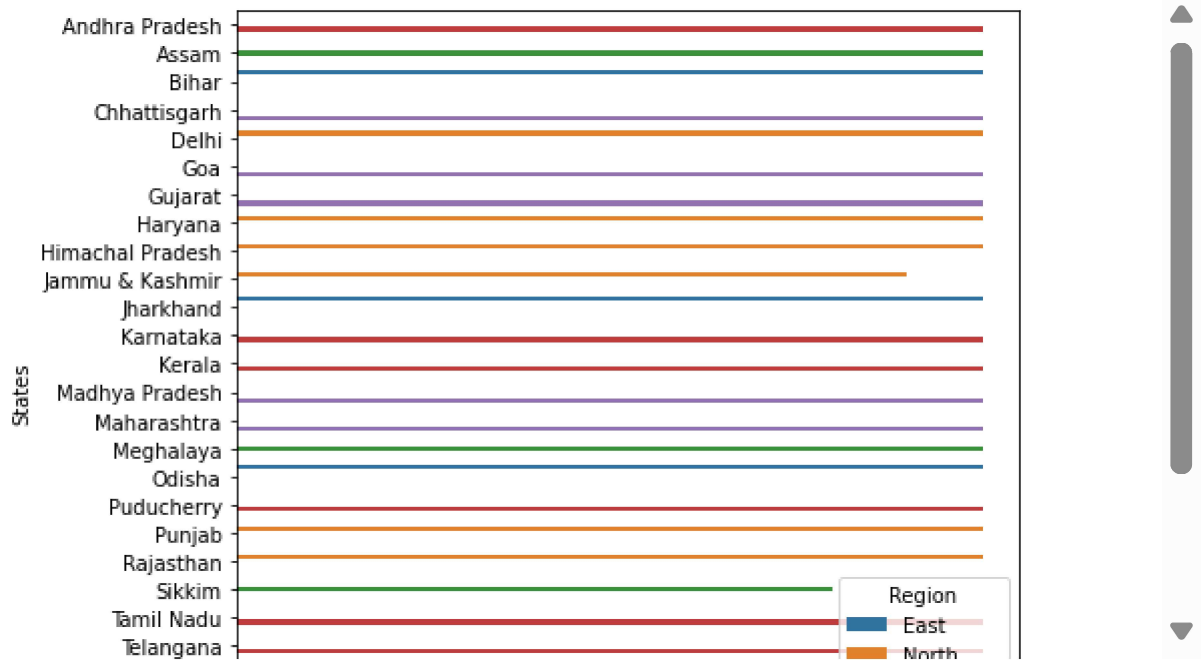
Data Visualization

checking the correlation between attributes using heatmap

In [11]:
```
sns.heatmap(df.corr(),cmap='summer',annot=True)
plt.show()
```

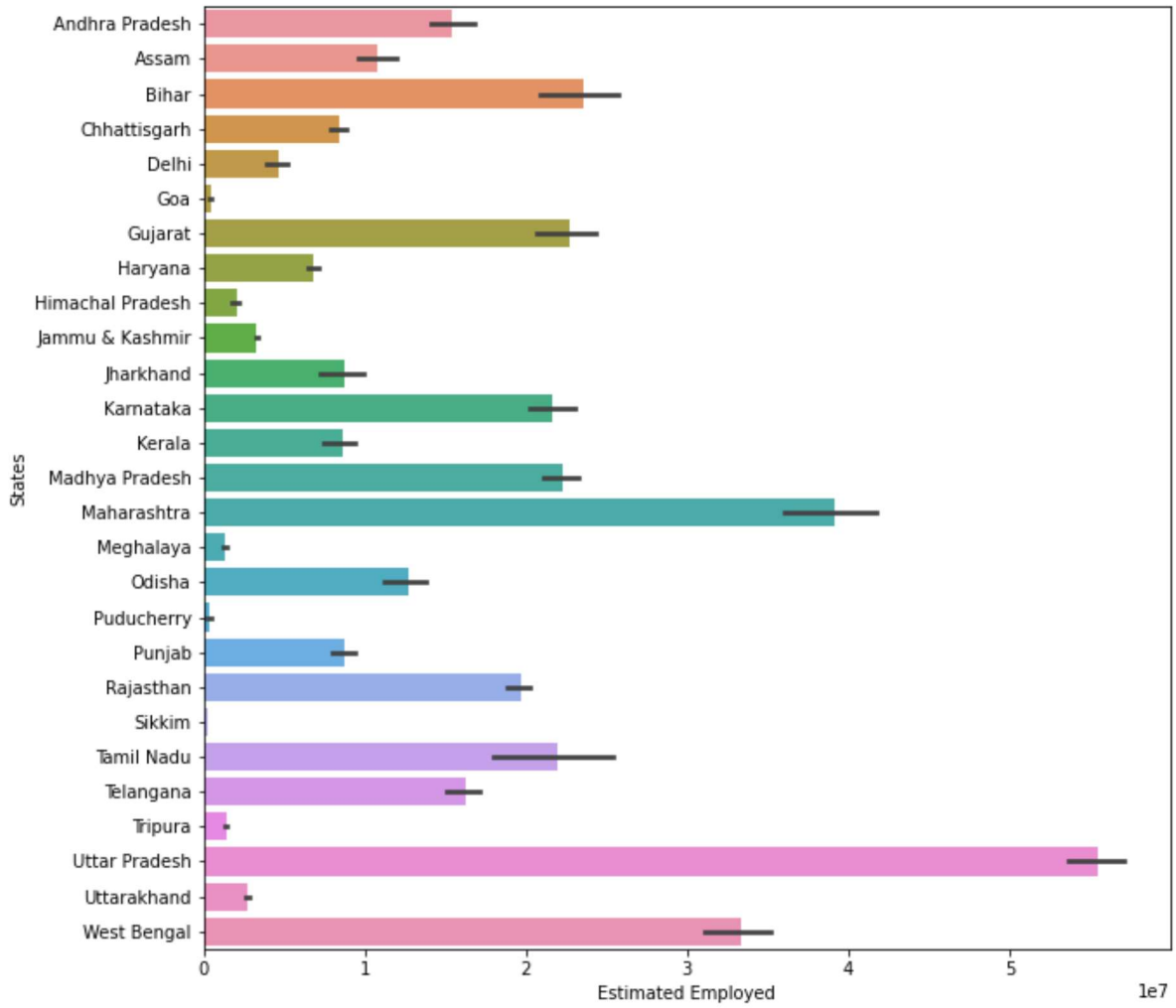
Create a countplot of the 'Region' column to visualize the distribution of the data across regions.

In [12]:
```
plt.figure(figsize=(7,7))
sns.countplot(y=df['States'],hue=df['Region'])
plt.show()
```



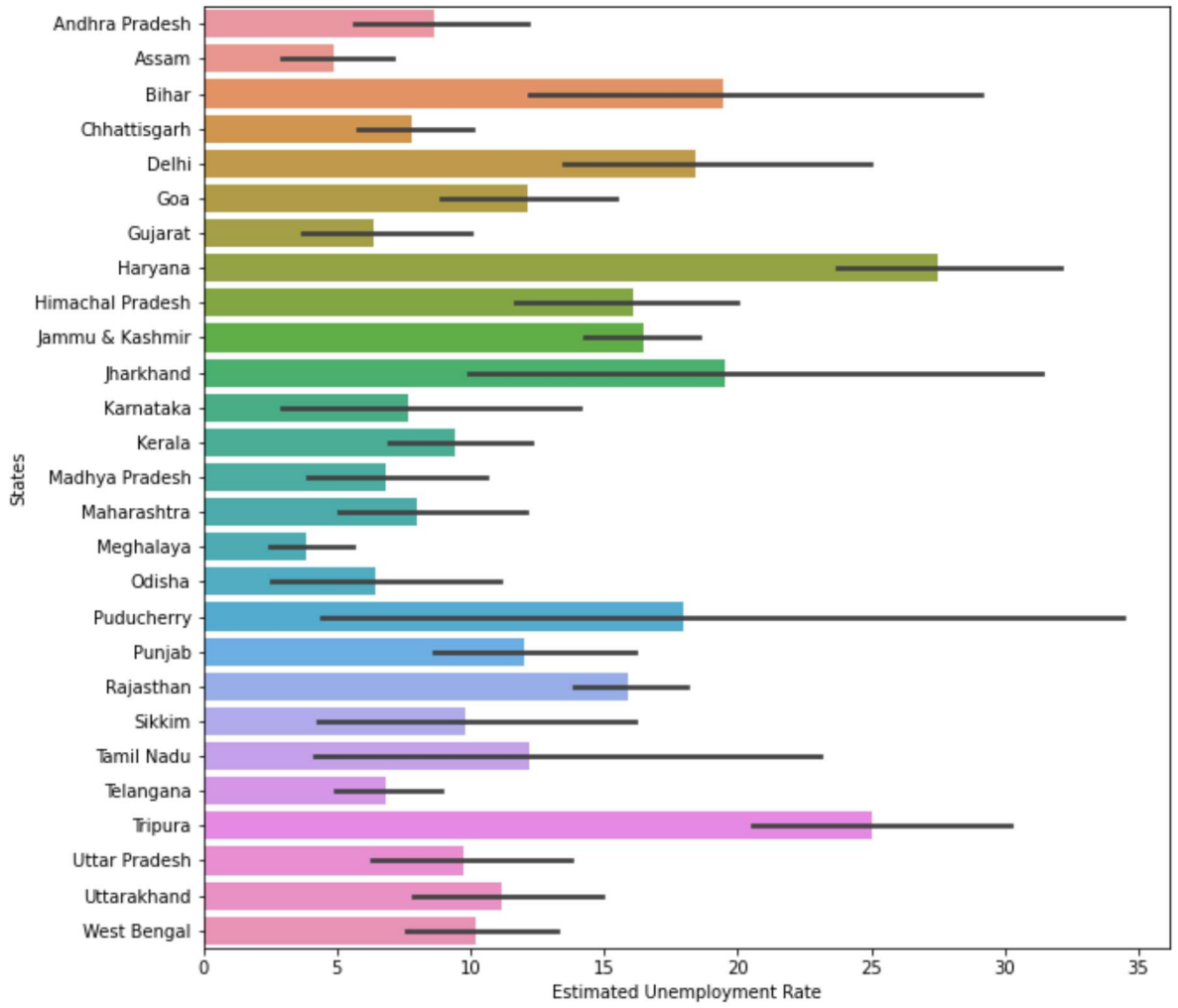
Create bar plots to visualize the estimated employed, unemployment rate, and labour participation rate for each state.

In [13]:
```
plt.figure(figsize=(10,10))
sns.barplot(y="States", x="Estimated Employed",data=df)
plt.show()
```



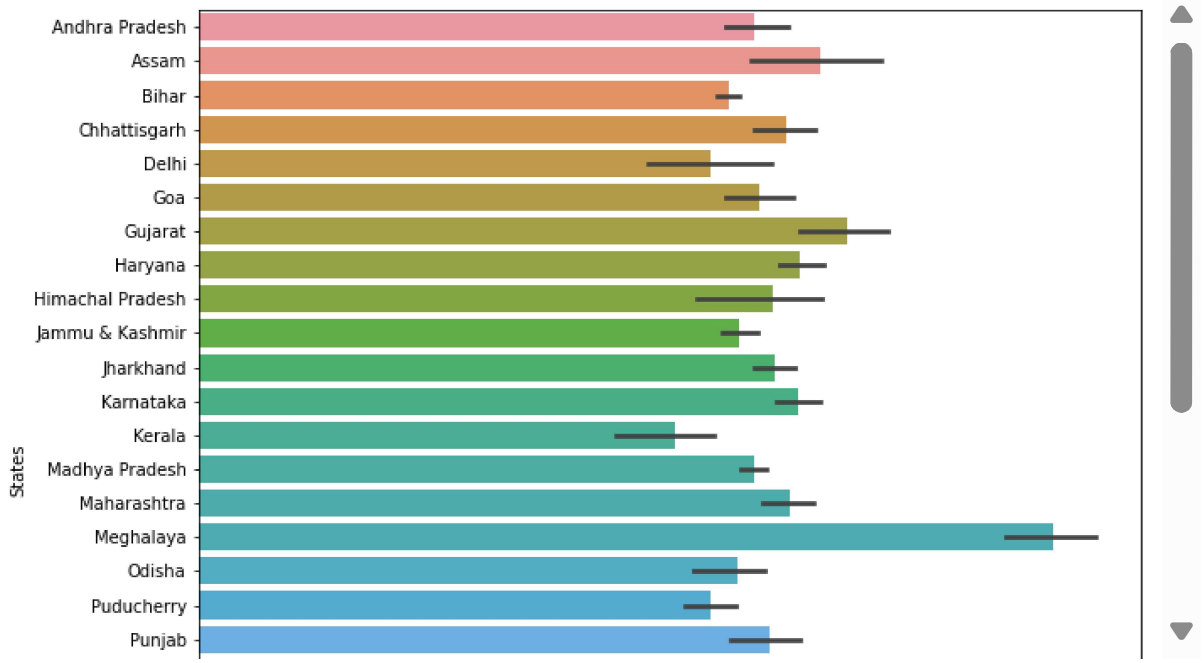
The highest number of people employed is in Uttar Pradesh and the lowest is in Sikkim. The data provided for Sikkim is less compared to other states. This can also be the reason for the decreased number of employed people in that state.

In [14]:
```
# Barplot- Estimated Unemployment Rate V/S Region

plt.figure(figsize=(10,10))
sns.barplot(y="States", x="Estimated Unemployment Rate",data=df)
plt.show()
```



The highest unemployment is in Haryana, while the lowest is in Meghalaya.

In [15]:
```
# Barplot- Estimated Labour Participation Rate V/S Region

plt.figure(figsize=(10,10))
sns.barplot(y="States", x="Estimated Labour Participation Rate",data=df)
plt.show()
```



The estimated labour participation rate is high in Meghalaya whereas its comparitively low in Kerala. This might be the reason why Meghalaya has lower unemployment rate compared to other states.

Create a sunburst chart to visualize the unemployment rate in each region and state.

In [16]:
```
unemplo_df = df[['States', 'Region', 'Estimated Unemployment Rate', 'Estimated
unemplo = unemplo_df.groupby(['Region', 'States'])['Estimated Unemployment Rat
fig = px.sunburst(unemplo, path=['Region', 'States'], values='Estimated Unempl
                  color_continuous_scale='Plasma', title='Unemployment rate in
                  height=650, template='ggplot2')
fig.show()
```

INFERENCE

The heatmap shows that there is a negative correlation between the estimated unemployment rate and the estimated employed. This means that as the number of people employed increases, the unemployment rate decreases. There is also a positive correlation between the estimated unemployment rate and the estimated labour participation rate. This means that as the number of people participating in the labour force increases, the unemployment rate also increases.

The countplot shows that the most observations in the dataset are from the North region. This is likely due to the fact that the North region has the largest population in India.

The bar plots show that the state with the highest number of employed people is Uttar Pradesh. The state with the highest unemployment rate is Haryana.

In [ ]:

In [ ]: