In [2]:

```
!pip install plotly"
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

```
Requirement already satisfied: plotly in c:\users\91772\.jupyter\anacondaa\lib\site-
packages (5.9.0)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\91772\.jupyter\anacondaa
\lib\site-packages (from plotly) (8.2.2)
```

In [3]:

```
df = pd.read_csv('Unemployment_Rate_upto_11_2020.csv')
```

In [4]:

```
df.columns
```

Out[4]:

```
Index(['Region', ' Date', ' Frequency', ' Estimated Unemployment Rate (%)',
       ' Estimated Employed', ' Estimated Labour Participation Rate (%)',
       'Region.1', 'longitude', 'latitude'],
      dtype='object')
```

In [5]:

```
df[' Frequency'].value_counts()
```

Out[5]:

```
 M    267
Name:  Frequency, dtype: int64
```

In [6]:

```
print(df['Region.1'].value_counts())
print(df['Region'].value_counts())
```

```
North        79
South        60
West         50
East         40
Northeast    38
Name: Region.1, dtype: int64
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
Goa                 10
Delhi               10
Chhattisgarh        10
Bihar               10
West Bengal         10
Jammu & Kashmir      9
Sikkim               8
Name: Region, dtype: int64
```

In [7]:

```
df.isnull().sum()
```

Out[7]:

```
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 [8]:

```
df.duplicated().sum()
```

Out[8]:

```
0
```

In [9]:

```
print('row count--->',df.shape[0])
print('column count--->',df.shape[1])
```

```
row count---> 267
column count---> 9
```

In [10]:

```
df.dtypes
```

Out[10]:

```
Region                                       object
 Date                                        object
 Frequency                                   object
 Estimated Unemployment Rate (%)            float64
 Estimated Employed                           int64
 Estimated Labour Participation Rate (%)    float64
Region.1                                     object
longitude                                   float64
latitude                                    float64
dtype: object
```

In [11]:

```
df[["day", "month", "year"]] = df[' Date'].str.split("-", expand = True)
df
```

Out[11]:

	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	79.7
1	Andhra Pradesh	29-02-2020	M	5.83	16545652	40.90	South	15.9129	79.7
2	Andhra Pradesh	31-03-2020	M	5.79	15881197	39.18	South	15.9129	79.7
3	Andhra Pradesh	30-04-2020	M	20.51	11336911	33.10	South	15.9129	79.7
4	Andhra Pradesh	31-05-2020	M	17.43	12988845	36.46	South	15.9129	79.7
...	...	...	...	...	...	...	...	...	
262	West Bengal	30-06-2020	M	7.29	30726310	40.39	East	22.9868	87.8
263	West Bengal	31-07-2020	M	6.83	35372506	46.17	East	22.9868	87.8
264	West Bengal	31-08-2020	M	14.87	33298644	47.48	East	22.9868	87.8
265	West Bengal	30-09-2020	M	9.35	35707239	47.73	East	22.9868	87.8
266	West Bengal	31-10-2020	M	9.98	33962549	45.63	East	22.9868	87.8

267 rows × 12 columns

In [12]:

```
df.drop(columns=[' Frequency'],axis=1,inplace=True)
```

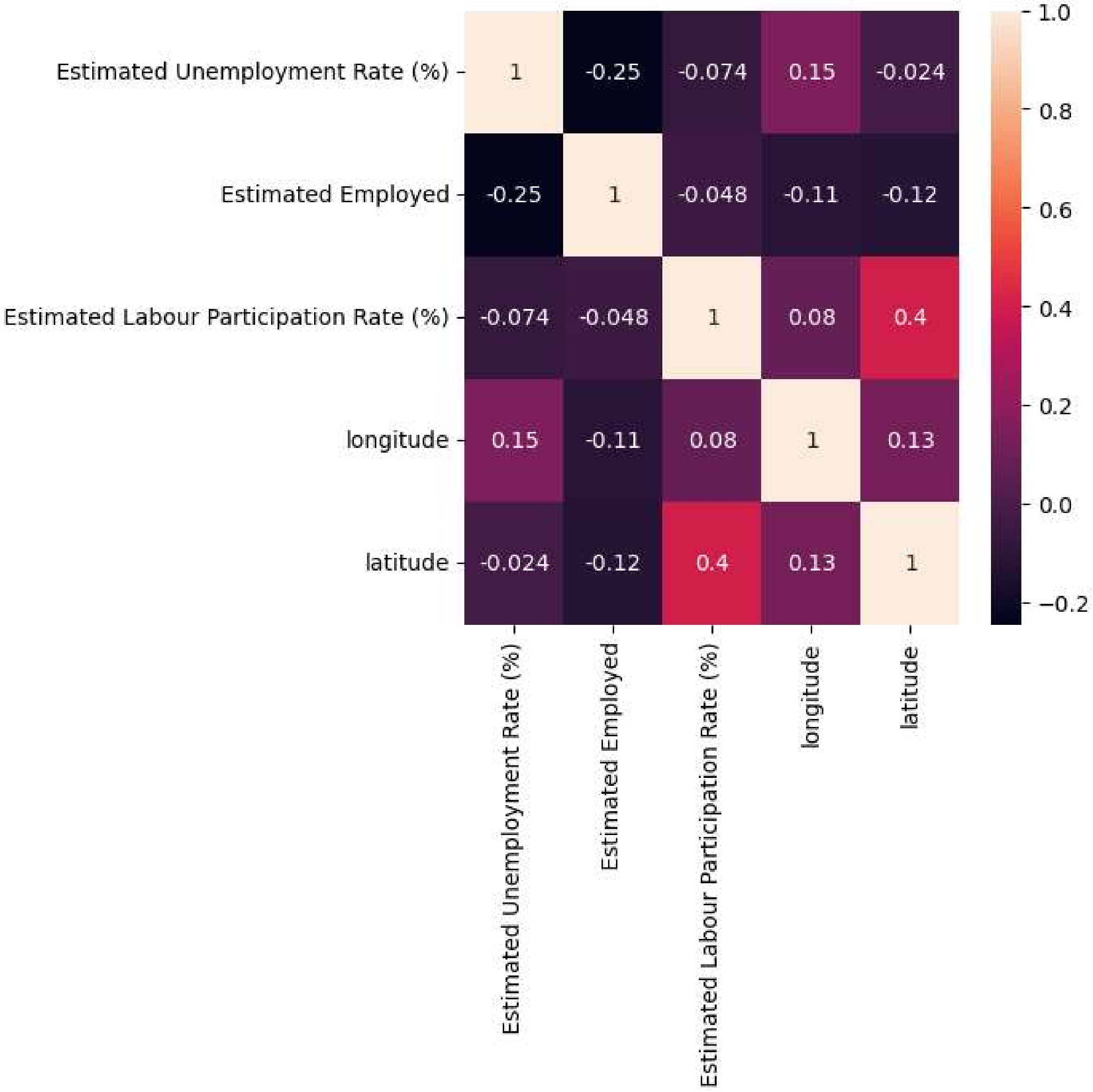
In [13]:

```
import matplotlib.pyplot as plt
```

In [14]:

```
plt.figure(figsize=(5,5))
sns.heatmap(df.corr(),annot=True)
plt.show()
```

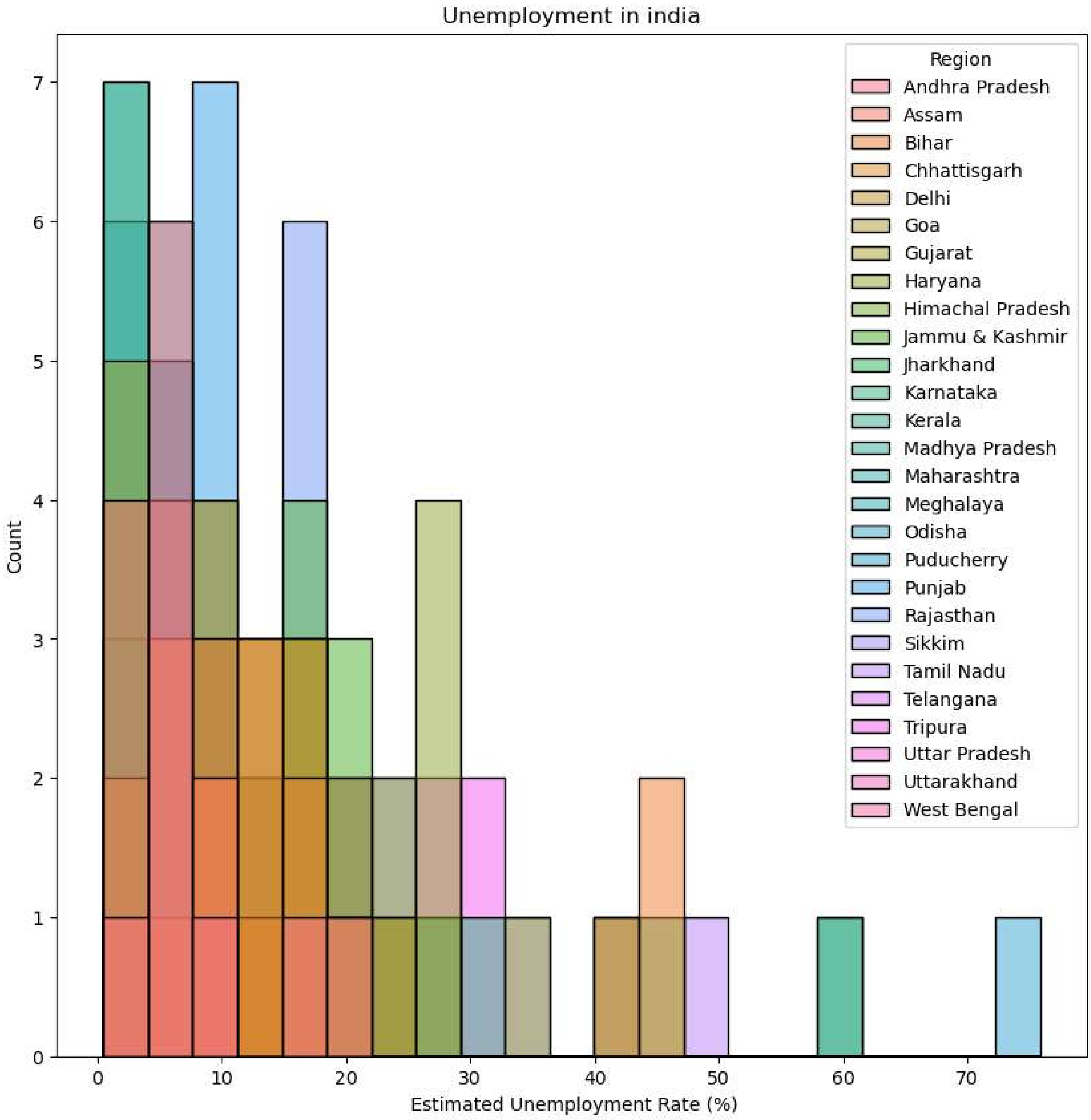
```
C:\Users\91772\AppData\Local\Temp\ipykernel_10180\2919022228.py:2: FutureWarning: Th
e default value of numeric_only in DataFrame.corr is deprecated. In a future versio
n, it will default to False. Select only valid columns or specify the value of numer
ic_only to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```



In [15]:
```
df.columns
```

Out[15]:
```
Index(['Region', ' Date', ' Estimated Unemployment Rate (%)',
       ' Estimated Employed', ' Estimated Labour Participation Rate (%)',
       'Region.1', 'longitude', 'latitude', 'day', 'month', 'year'],
      dtype='object')
```

In [16]:
```
plt.figure(figsize=(10,10))
plt.title("Unemployment in india")
sns.histplot(x=' Estimated Unemployment Rate (%)',hue= "Region", data=df,kde=False)
plt.show()
```

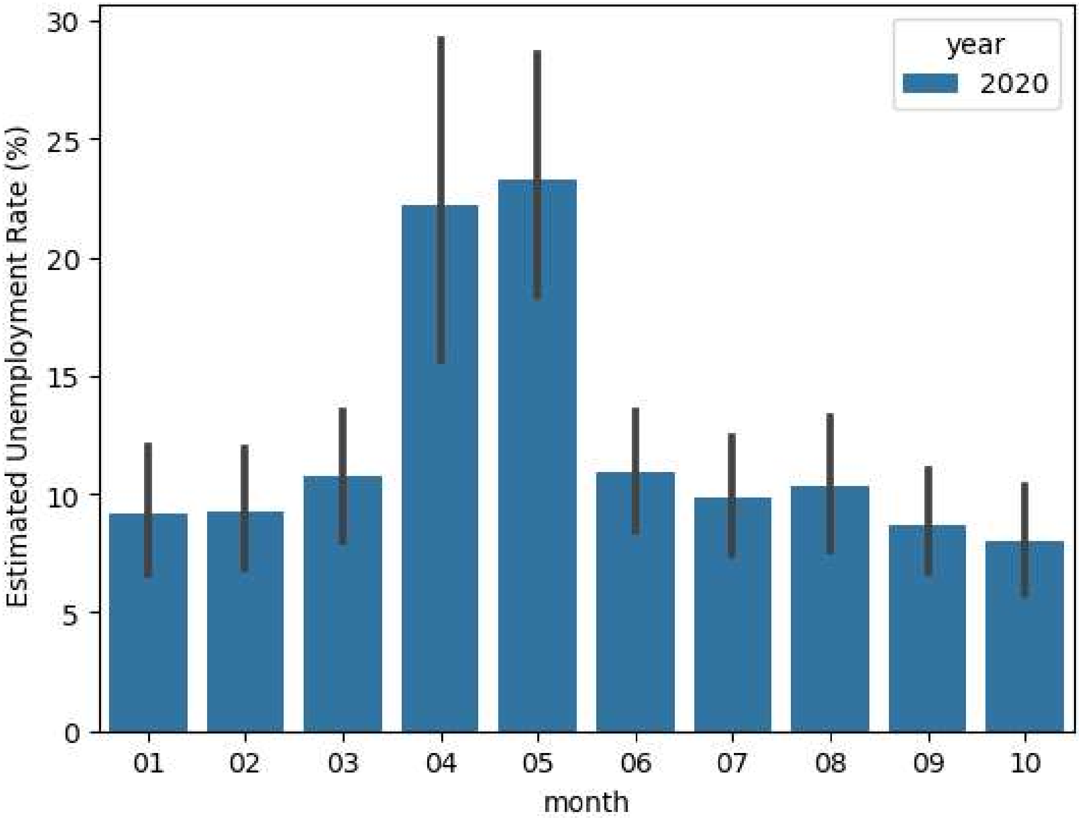


In [17]:
```
df.month.unique()
```

Out[17]:
```
array(['01', '02', '03', '04', '05', '06', '07', '08', '09', '10'],
      dtype=object)
```

In [18]:
```
sns.barplot(x='month',y=' Estimated Unemployment Rate (%)',hue='year',data=df)
```

Out[18]:
```
<Axes: xlabel='month', ylabel=' Estimated Unemployment Rate (%)'>
```

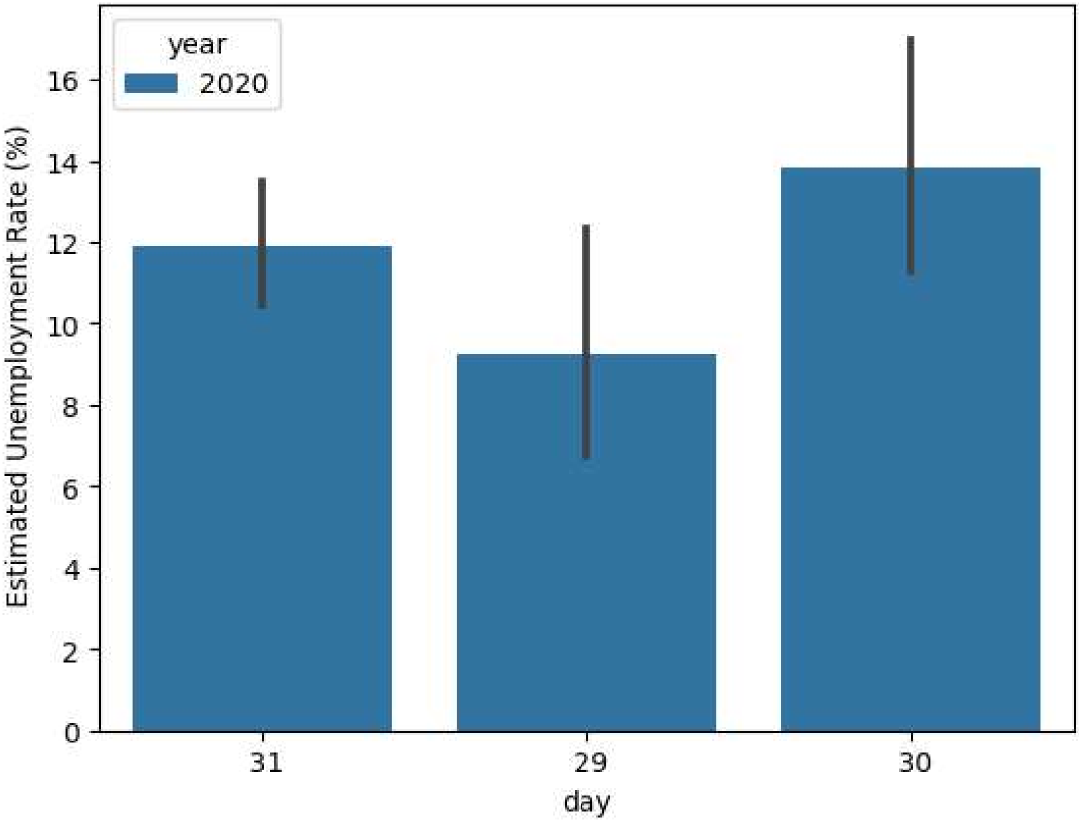


In [19]:
```
df.day.unique()
```

Out[19]:
```
array([' 31', ' 29', ' 30'], dtype=object)
```

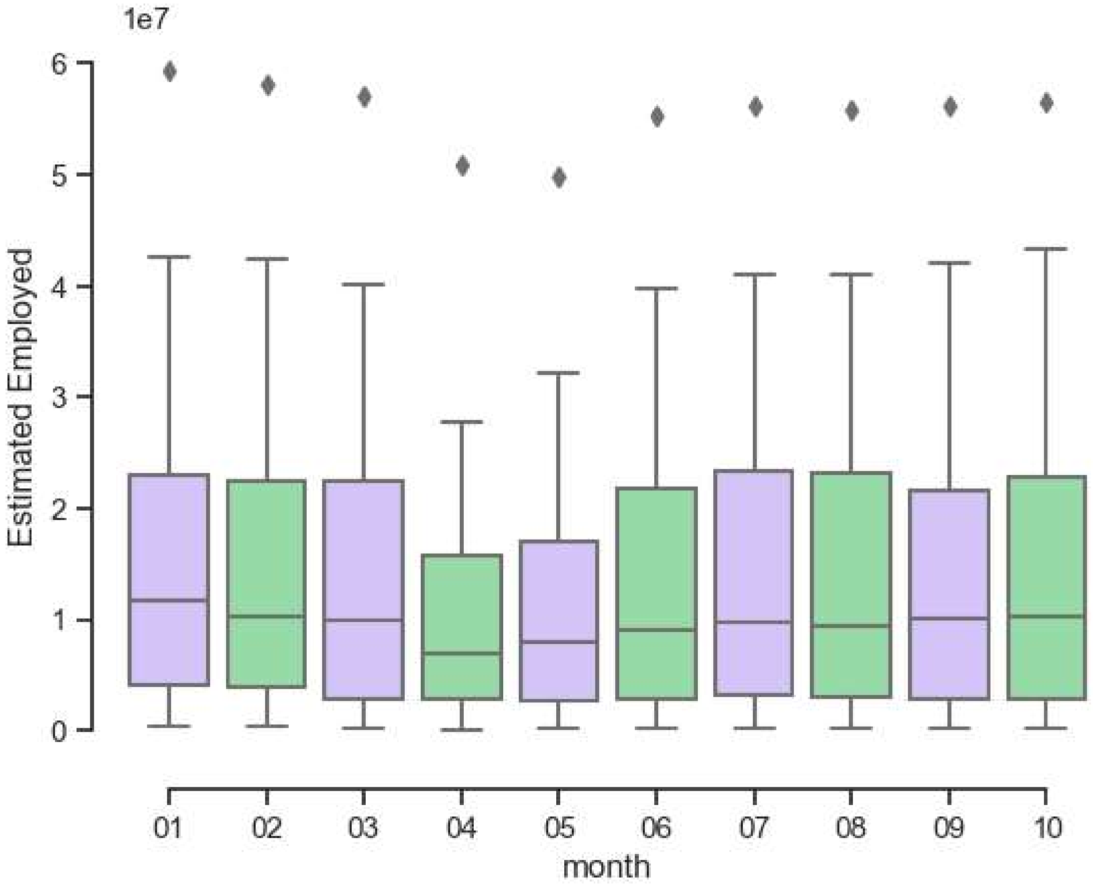
In [20]:
```
sns.barplot(x='day',y=' Estimated Unemployment Rate (%)',hue='year',data=df)
```

Out[20]:
```
<Axes: xlabel='day', ylabel=' Estimated Unemployment Rate (%)'>
```



In [21]:
```
import seaborn as sns
sns.set_theme(style="ticks", palette="pastel")
```

```
# Draw a nested boxplot to show bills by day and time
sns.boxplot(x="month", y=' Estimated Employed', palette=["m", "g"],
            data=df)
sns.despine(offset=10, trim=True)
```



In [22]:
```
#we can also drop this year column,beacause it also contains constant values of year

df.drop('year',axis=1)
```

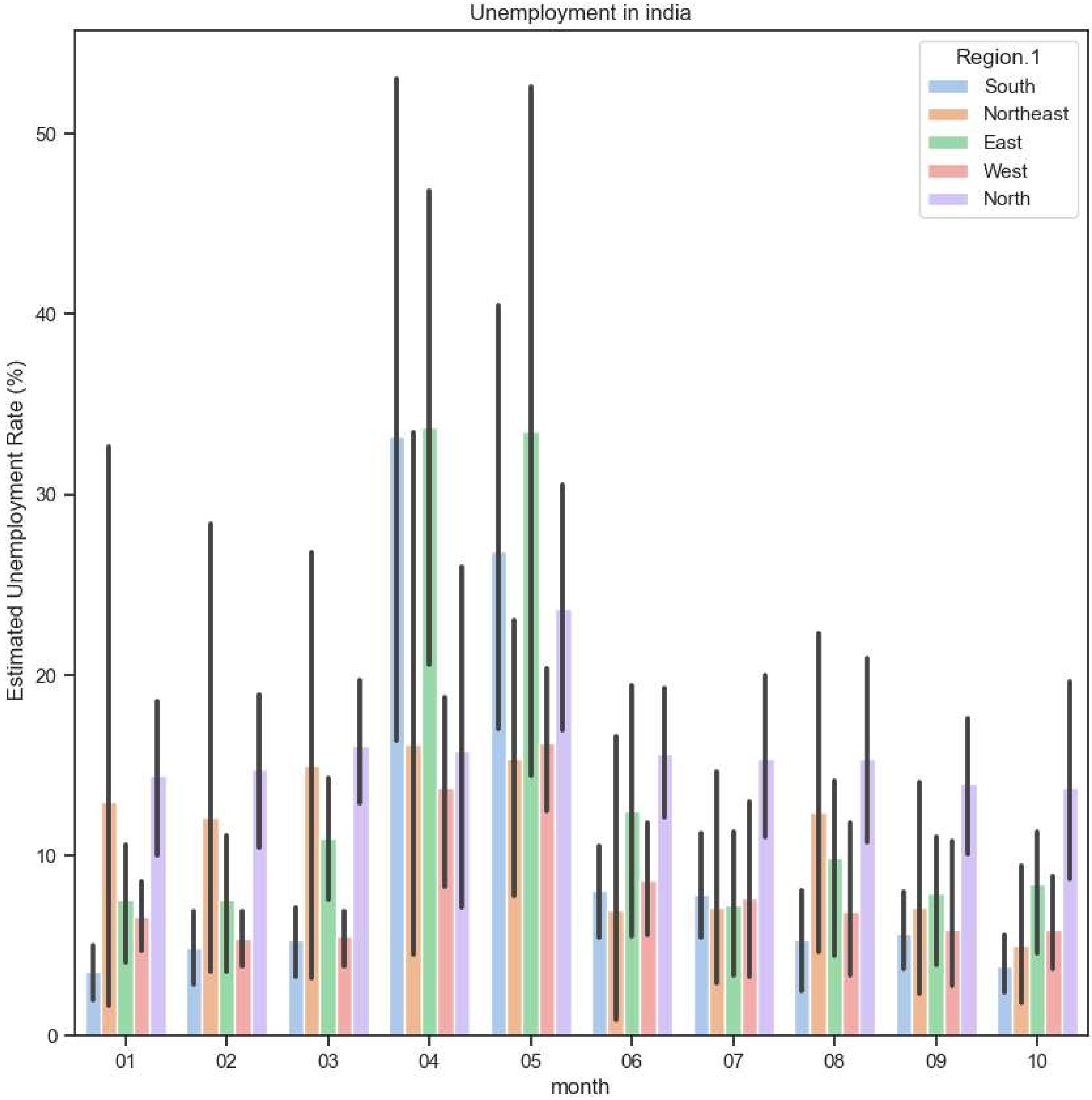
Out[22]:

	Region	Date	Estimated Unemployment Rate (%)	Estimated Employed	Estimated Labour Participation Rate (%)	Region.1	longitude	latitude	day	m
0	Andhra Pradesh	31-01-2020	5.48	16635535	41.02	South	15.9129	79.740	31	
1	Andhra Pradesh	29-02-2020	5.83	16545652	40.90	South	15.9129	79.740	29	
2	Andhra Pradesh	31-03-2020	5.79	15881197	39.18	South	15.9129	79.740	31	
3	Andhra Pradesh	30-04-2020	20.51	11336911	33.10	South	15.9129	79.740	30	
4	Andhra Pradesh	31-05-2020	17.43	12988845	36.46	South	15.9129	79.740	31	
...	...	...	...	...	...	...	...	...	...	
262	West Bengal	30-06-2020	7.29	30726310	40.39	East	22.9868	87.855	30	
263	West Bengal	31-07-2020	6.83	35372506	46.17	East	22.9868	87.855	31	
264	West Bengal	31-08-2020	14.87	33298644	47.48	East	22.9868	87.855	31	
265	West Bengal	30-09-2020	9.35	35707239	47.73	East	22.9868	87.855	30	
266	West Bengal	31-10-2020	9.98	33962549	45.63	East	22.9868	87.855	31	

267 rows × 10 columns

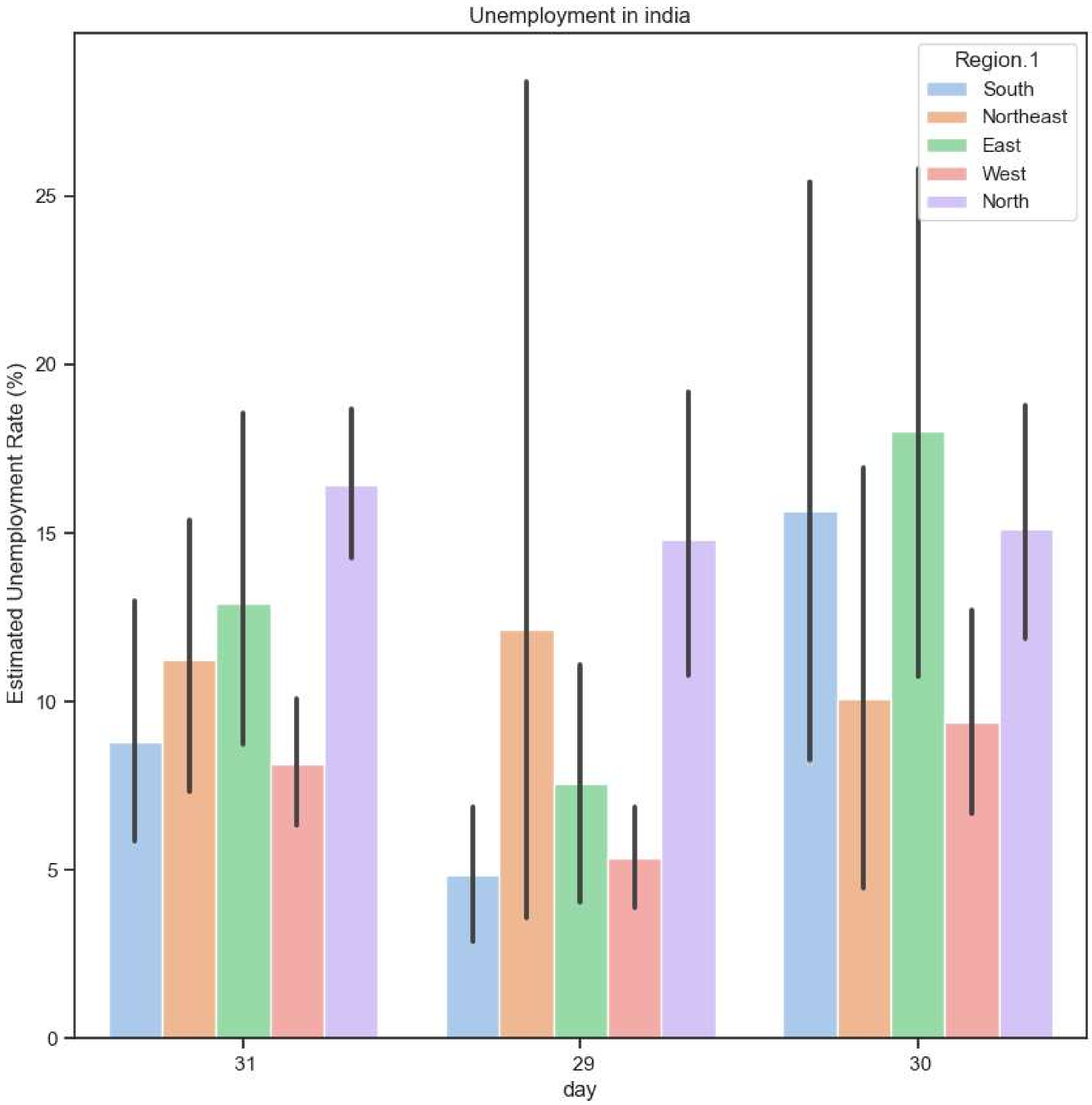
In [23]:

```
plt.figure(figsize=(10,10))
plt.title("Unemployment in india")
sns.barplot(x='month',y =' Estimated Unemployment Rate (%)',hue='Region.1', data=df)
plt.show()
```



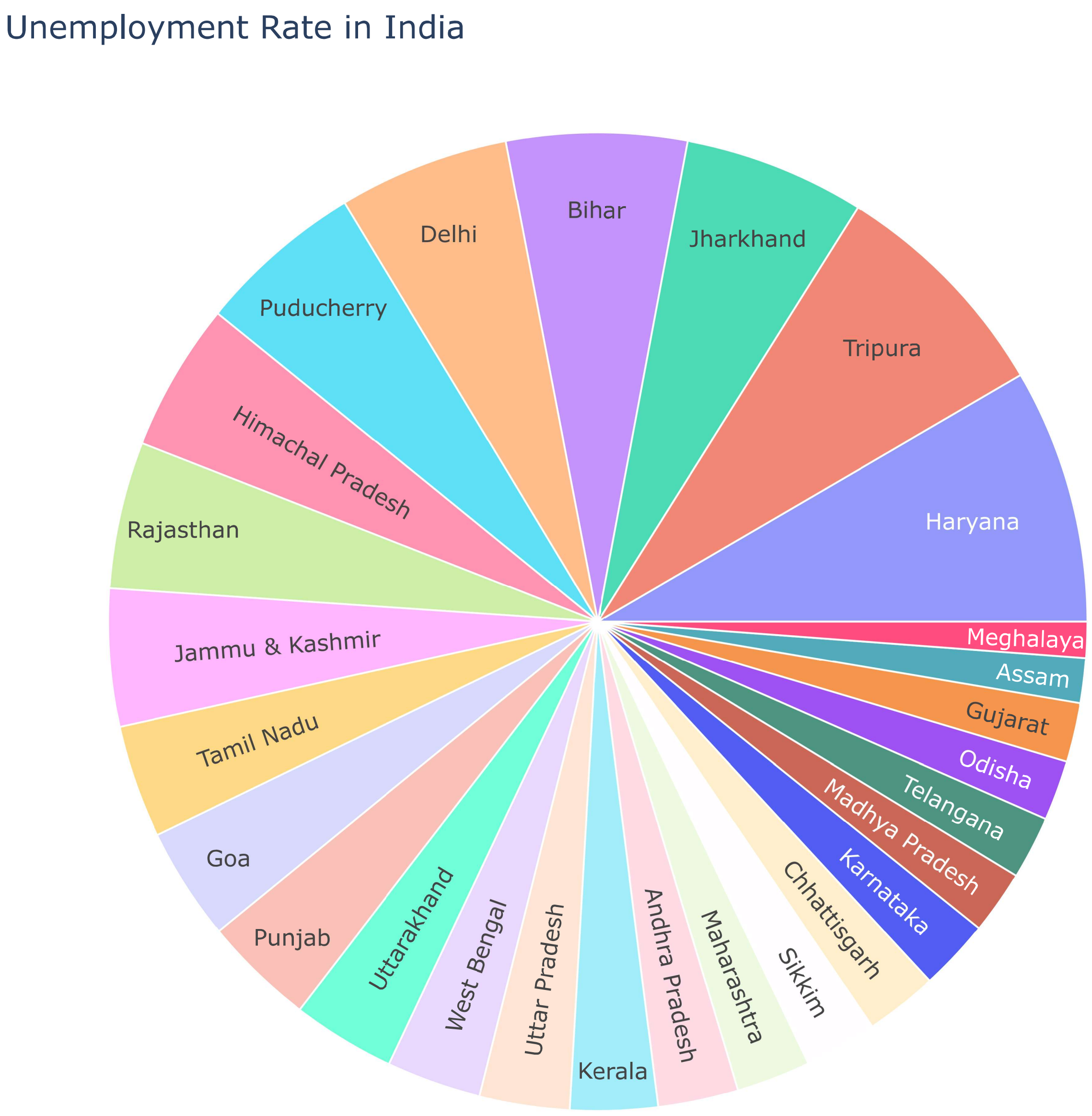
In [24]:

```
plt.figure(figsize=(10,10))
plt.title("Unemployment in india")
sns.barplot(x='day',y =' Estimated Unemployment Rate (%)',hue='Region.1', data=df)
plt.show()
```

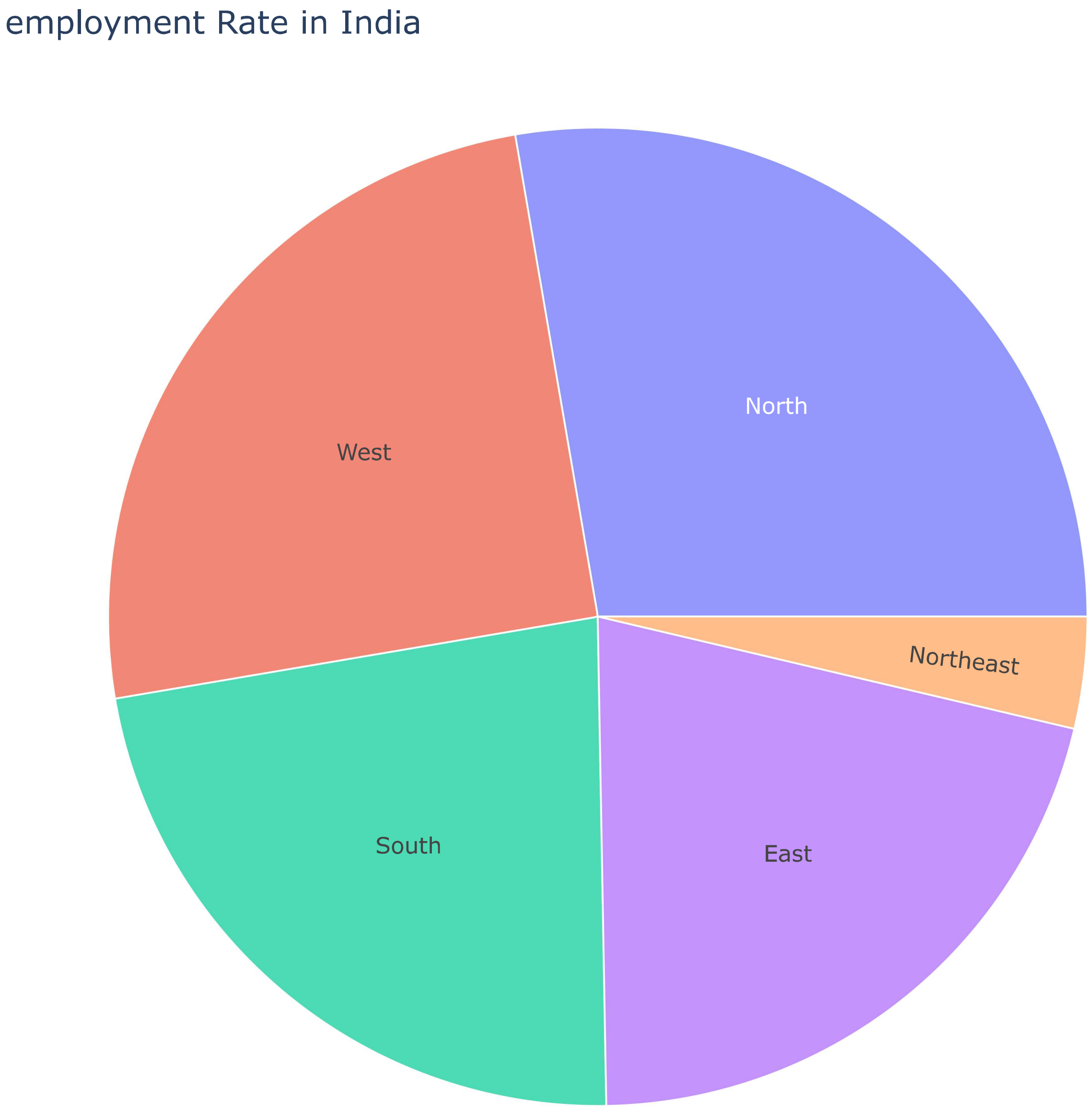


In [25]:

```
unemploment = df[["Region",' Estimated Unemployment Rate (%)']]
figure = px.sunburst(unemploment, path=["Region"],
                     values=' Estimated Unemployment Rate (%)',
                     width=700, height=700, color_continuous_scale="RdY1Gn",
                     title="Unemployment Rate in India")
figure.show()
```



In [26]:
```
unemploment = df[["Region.1",' Estimated Employed']]
figure = px.sunburst(unemploment, path=["Region.1"],
                     values=' Estimated Employed',
                     width=700, height=700, color_continuous_scale="RdY1Gn",
                     title="employment Rate in India")
figure.show()
```



In [ ]: