

## ▼ Indian Tourism Analysis

Tourism is one the largest service industry in India. It is important for the country's economy and is growing rapidly. The World Travel and Tourism Council calculated that tourism generated ₹16.91 lakh crore (US\$220 billion) or 9.2% of India's GDP in 2018 and supported 42.673 million jobs, 8.1% of its total employment. Tourism is estimated to have directly contributed 2.7 per cent to GDP and 6.7 per cent to the employment of the country in 2019-20. The main reason why India is famous for tourism is because all types of tourist destinations exist in one place. Each state in India is famous for different tourist attractions. India is the right place for wild safaris, desert safaris, nature seeking places, honeymoon spot, adventurous activities, spiritual places, and historical places of India.

So from the data I have obtained, I have analyzed how tourism has been faring in India during all these years.

This notebook focuses on Tourism analysis of India (primarily of foreign tourists) from the data obtained from [data.gov.in](#).

Modules used:

- Pandas and Numpy: For loading and manipulating data.
- Matplotlib and Seaborn: For visualization.

The dataset is stored in the `./dataset` folder

So.. Let's begin the analysis!!

### Importing the modules

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline

sns.set(style="darkgrid")
```

## Statistics of International Tourists to India [1981 - 2020]

The dataset used here is the statistics of the foreign tourists who visited India from 1981 to 2020. This dataset also includes the segregation of the International tourists (ITAs) into foreign tourist arrivals (FTAs) and Non-resident Indians (NRIs). The percentage changes between consecutive values is included in the dataset but it isn't considered here for visualization.

Before visualizing the dataset, we need to prepare the data for Visualization.

### Preparing the Dataset

```
# Fetching data from the csv file
df=pd.read_csv("./dataset\India-Tourism-Statistics-1981-2020-fta_nri_ita.csv")
fta_nri_ita=df.copy()
fta_nri_ita.head()
```

```
# renaming and dropping columns
fta_nri_ita.rename(columns={'FTAs in India (in million)': 'FTA','NRIs arrivals in India (in million)':'NRI','ITAs in India (in million)':'ITA'})
fta_nri_ita.drop(fta_nri_ita.iloc[:,2::2],axis=1,inplace=True)
fta_nri_ita.head()
```

	Year	FTA	NRI	ITA
0	1981	1.28	NaN	NaN
1	1991	1.68	NaN	NaN
2	2001	2.54	NaN	NaN
3	2002	2.38	NaN	NaN
4	2003	2.73	NaN	NaN

# Combining columns FTA,NRI,ITA into a single column 'value' and column 'variable' will distinguish these values

```
df=pd.melt(fta_nri_ita,['Year']).rename(columns={'variable':'Tourists'})
```

```
df
```

	Year	Tourists	value
0	1981	FTA	1.28
1	1991	FTA	1.68
2	2001	FTA	2.54
3	2002	FTA	2.38
4	2003	FTA	2.73
...	...	...	...
61	2016	ITA	15.03
62	2017	ITA	16.81
63	2018	ITA	17.42
64	2019	ITA	17.91
65	2020	ITA	6.33

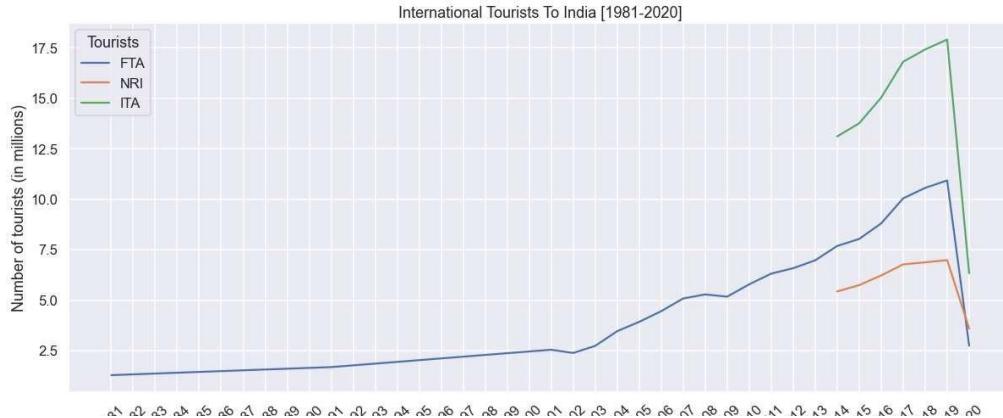
66 rows × 3 columns

## Visualizing the Dataset

```
# plotting df in a line graph
plt.figure(figsize=(20,8)) # setting figure size

sns.set_context("talk") #https://seaborn.pydata.org/generated/seaborn.set_context.html
g=sns.lineplot(x='Year', y='value', hue='Tourists', data=df)
g.set(xlabel="Year",ylabel="Number of tourists (in millions)", title="International Tourists To India [1981-2020]")
plt.xticks(
    range(1981,2021),
    rotation=45,
    horizontalalignment='right',
    fontweight='light'
)
plt.plot()
```

[ ]



As we can see from the above **multiline graph**, there has been a steady increase in the number of tourists visiting India until 2020 as **COVID-19** restrictions led to a huge blow on the Tourism sector. The pandemic has not only jeopardised the growth prospects of the sector but also caused a significant economic slump, which the sector would take some time to recover from.

Also, the growth in FTAs in India over the years, viz. 2003 to 2007, 2010, 2014 and 2017 saw a double-digit positive growth, while there was negative growth in the years 1991, 2001, 2002 (September 11 attacks i.e 9/11 attacks in U.S and many other factors) and 2009 (because of the 26/11 attack in Mumbai (Bombay), Maharashtra, India).

Note: Ministry of tourism initiated compilation and dissemination of arrivals of Non-Resident Indians (NRIs) and the Foreign Tourist Arrivals (FTAs) since 2014.

## ▼ Tourist segregation based on age [2001 - 2019]

The dataset used here is the statistics of the distribution of FTAs to India by age from 2001 to 2019 in percentages. These percentages will be later on converted to numbers.

Before visualizing the dataset, we need to prepare the data for Visualization.

## ▼ FTAs segregated based on age [2001 - 2019]

### Preparing the Dataset

```
# fetching data from the csv file
agegroup=pd.read_csv("./dataset\India-Tourism-Statistics-2001-2019-agegroup.csv")
agegroup.head()
```

Year	FTAs	% distribution by Age-Group (in Year) - 0-14	% distribution by Age-Group (in Year) - 15-24	% distribution by Age-Group (in Year) - 25-34	% distribution by Age-Group (in Year) - 35-44	% distribution by Age-Group (in Year) - 45-54	% distribution by Age-Group (in Year) - 55-64
0	2001 253/282	7.0	10.8	20.1	21.1	19.4	11
1	2002 238/4364	9.2	10.0	19.4	21.6	19.4	11
2	2003 272/6214	7.2	10.0	19.5	21.6	19.4	11
3	2004 345/7477	8.5	9.8	18.8	21.3	19.4	12

```
# Renaming the columns
agegroup_elaborate=agegroup.copy().rename(columns={
    '% distribution by Age-Group (in Year) - 0-14':'0-14',
    '% distribution by Age-Group (in Year) - 15-24':'15-24',
    '% distribution by Age-Group (in Year) - 25-34':'25-34',
    '% distribution by Age-Group (in Year) - 35-44':'35-44',
    '% distribution by Age-Group (in Year) - 45-54':'45-54',
    '% distribution by Age-Group (in Year) - 55-64':'55-64',
    '% distribution by Age-Group (in Year) - 65 and above':'65 and above',
    '% distribution by Age-Group (in Year) - Not Reported':'Not Reported',})
```

})

```
# converting %ages to actual values and shifting the decimal places by 3 to the left
agegroup_elaborate['0-14']=agegroup_elaborate['0-14']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['15-24']=agegroup_elaborate['15-24']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['25-34']=agegroup_elaborate['25-34']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['35-44']=agegroup_elaborate['35-44']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['45-54']=agegroup_elaborate['45-54']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['55-64']=agegroup_elaborate['55-64']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['65 and above']=agegroup_elaborate['65 and above']*agegroup_elaborate['FTAs']/100 * 10**-3
agegroup_elaborate['Not Reported']=agegroup_elaborate['Not Reported']*agegroup_elaborate['FTAs']/100 * 10**-3

agegroup_elaborate.head()
```

	Year	FTAs	0-14	15-24	25-34	35-44	45-54	55-64	65 and above
0	2001	2537282	177.609740	274.026456	509.993682	535.366502	492.232708	301.936558	169.997894
1	2002	2384364	219.361488	238.436400	462.566616	515.022624	462.566616	274.201860	183.596028
2	2003	2726214	196.287408	272.621400	531.611730	588.862224	528.885516	313.514610	209.918478
3	2004	3457477	293.885545	338.832746	650.005676	736.442601	670.750538	442.557056	283.513114
4	2005	3918610	337.000460	376.186560	736.698680	834.663930	764.128950	509.419300	340.919070

## ▼ Visualizing the Dataset

```
# dropping columns
agegroup_elaborate.drop(['FTAs','Not Reported'],axis=1,inplace=True)
# Combining columns w.r.t 'Year' into a single column 'value' and column 'variable' will distinguish these values
df=pd.melt(agegroup_elaborate,['Year']).rename(columns={"variable":"Age Group"})

# plotting the multiline graph
plt.figure(figsize=(20,8))

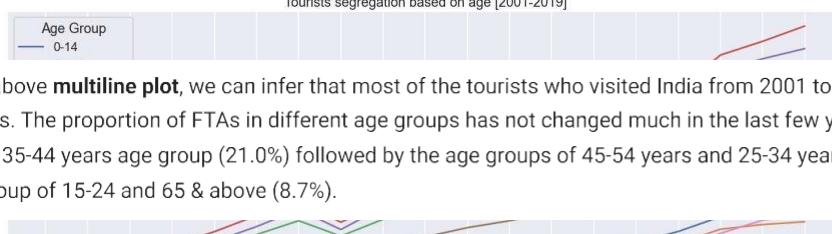
sns.set_context("talk")

g=sns.lineplot(x='Year', y='value', hue='Age Group', data=df)
g.set(xlabel="Year",ylabel="Number of tourists (in thousands)", title="Tourists segregation based on age [2001-2019]")
plt.xticks(
    range(2001,2020),
    rotation=45,
    # horizontalalignment='right',
    fontweight='light'
)

# display the chart
plt.plot()
```



[ ]



## ▼ Average %age distribution of Tourists based on age [2001 - 2019]



## ▼ Preparing the Dataset

```
# finding average %ages for each agegroup from their actual values
tot_fta=agegroup['FTAs'].sum()

# 10**5 (100 * 10**3) since distribution among agegroups was converted to thousands previously
avg_agegroup_pctg=[
    agegroup_elaborate['0-14'].sum()*10**5/tot_fta,
    agegroup_elaborate['15-24'].sum()*10**5/tot_fta,
    agegroup_elaborate['25-34'].sum()*10**5/tot_fta,
    agegroup_elaborate['35-44'].sum()*10**5/tot_fta,
    agegroup_elaborate['45-54'].sum()*10**5/tot_fta,
    agegroup_elaborate['55-64'].sum()*10**5/tot_fta,
    agegroup_elaborate['65 and above'].sum()*10**5/tot_fta
]

print(avg_agegroup_pctg)
```

```
[9.290327244694959, 8.526830283828492, 18.353340597195853, 20.92885960109853, 19.620265319294138, 13.637821302553538, 8.356931097199208]
```

## ▼ Visualizing the Dataset

```
colors = ['maroon', 'crimson', 'dodgerblue', 'aqua', 'lightgreen','gold','yellowgreen']
agegroup_list=['0-14','15-24','25-34','35-44','45-54','55-64','65 and above']

# plotting data on chart
plt.figure(figsize=(8,6))

pie=plt.pie(avg_agegroup_pctg, colors=colors,
            autopct='%.1f%%',shadow=True, startangle=0, textprops={'fontsize': 15},pctdistance = 1.15,)
# plt.legend()
plt.legend(pie[0],agegroup_list, bbox_to_anchor=(1.5,0.7), loc="upper right", fontsize=15,
           bbox_transform=plt.gcf().transFigure)

plt.title("Average % of tourists based on age [2001-2019]",fontsize=20,fontweight='bold')

# displaying chart
plt.show()
```

**Average % of tourists based on age [2001-2019]**

As per the above **pie chart**, it can be deduced that the highest percentage of tourists visiting India are youths and the middleagers. This can be due to the fact that the youngsters aren't that well off when it comes to capital and the senior citizens would prefer to travel to places that are relaxing and comfortable or India doesn't end up attracting tourists from these age groups.

As much as the elderly might avoid and refrain; travelling is the perfect resort for them to rejuvenate themselves and finally enjoy life the way they want to. Just like kids or the young, even they need a vacation for a variety of reasons- the primary one being that it boosts their physical, mental, and emotional health. But, Most people, feel that the biggest deterrent to tourism is lack of infrastructure, Lack of good budget hotels , proper roads as well as hygiene and sanitation are problems that put off most would-be travellers to India.

The youngsters on the other hand could contribute largely to Indian tourism especially through education. Although the main purpose of international student mobility is to study, it can also be considered a type of tourist activity. India is still not seen by international applicants as a destination for education because of its education system which needs a lot of improvement. If Indian education does improve and gets recognized by the world, then there could be a rise in the international students to India which could indirectly improve the contribution by the youngsters to Indian Tourism.

## ▼ Indian Tourism Statewise Statistics [2019 - 2020]

The dataset used here is the statistics of the Domestic and foreign tourists to India during 2019 and 2020 per state. This dataset also includes the growth rate which won't be considered for visualization.

Before visualizing the dataset, we need to prepare the data for Visualization.

## ▼ Foreign tourists per State/UTs [2019 - 2020]

### ▼ Preparing the Dataset

```
# fetching data from the csv file
statewise_data=pd.read_csv("./dataset\India-Tourism-Statistics-statewise_2019-2020 Domestic_Foreign.csv")
statewise_data.head()
```

S. No.	States/UTs	Domestic -2019	Foreign - 2019	Domestic -2020	Foreign - 2020	Growth rate - DTV 2020/19	Growth rate - FTV 2020/19
0	1 A&N Island	505398	16206	191207	5412	-62.17	-66.6
1	2 Andhra Pradesh	237051508	280356	70828590	67591	-70.12	-75.89
2	3 Arunachal Pradesh	555639	7825	42871	961	-92.28	-87.72

```
# Parsing dataframe
foreign_statewise=statewise_data[['States/UTs','Foreign - 2019','Foreign - 2020']].copy()
# sorting by Foreign-2019 in descending order
foreign_statewise.sort_values(by=['Foreign - 2019'], ascending=False, inplace=True)
foreign_statewise=foreign_statewise[foreign_statewise['States/UTs'].isin(['Grand Total'])==False]
```

```
# shifting the decimal places by 5 to the left
foreign_statewise['Foreign - 2019']=foreign_statewise['Foreign - 2019'] * 10**-5
foreign_statewise['Foreign - 2020']=foreign_statewise['Foreign - 2020'] * 10**-5
foreign_statewise.head()
```

	States/UTs	Foreign - 2019	Foreign - 2020
31	Tamil Nadu	68.66327	12.28323
21	Maharashtra	55.28704	12.62409
34	Uttar Pradesh	47.45181	8.90932
9	Delhi	29.83436	6.81230
36	West Bengal	16.56145	4.63285

## ▼ Visualizing the Dataset

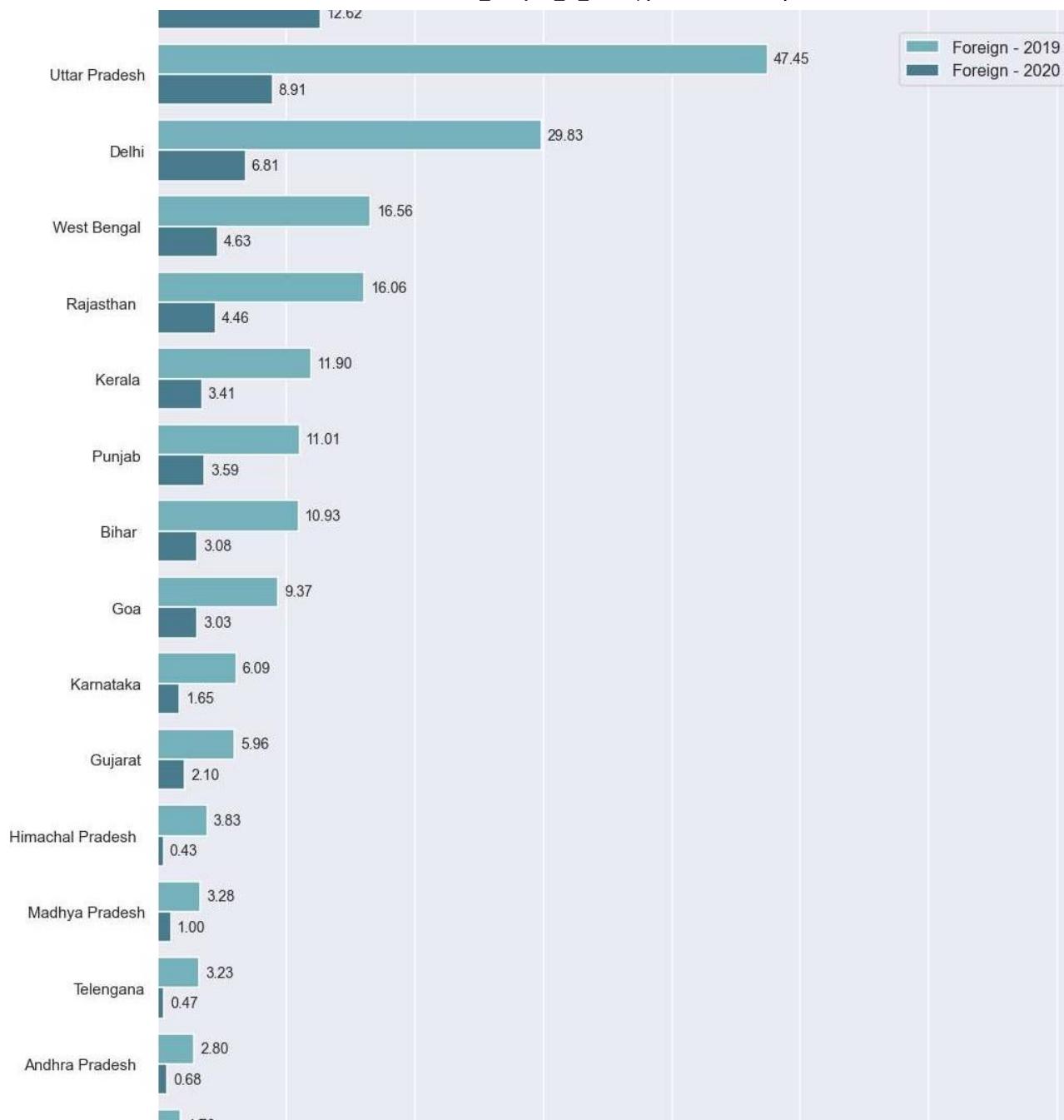
```
# Combining columns w.r.t 'States/UTs' into a single column 'value' and column 'variable' will distinguish these values
df=pd.melt(foreign_statewise,['States/UTs']).rename(columns={"variable":"Year"})

# plotting data on horizontal bar graph
plt.figure(figsize=(16,50))
ax = sns.barplot(x="value", y="States/UTs", data=df, hue='Year', palette='GnBu_d')

for p in ax.patches:
    width = p.get_width()      # get bar length
    ax.text(width + 0.5,       # set the text at 1 unit right of the bar
            p.get_y() + p.get_height() / 2, # get Y coordinate + X coordinate / 2
            '{:1.2f}'.format(width), # set variable to display, 2 decimals
            ha = 'left',   # horizontal alignment
            va = 'center',  # vertical alignment
            fontsize=14)

plt.title("Foreign tourists per State/UTs [2019, 2020]", fontsize=20, fontweight='bold')
plt.xlabel("Arrivals (in lakhs)", fontsize=18)
plt.ylabel("States/UTs", fontsize=18)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)
plt.legend(bbox_to_anchor=(0.99,0.95), prop={'size': 16})

# display the chart
plt.show()
```



As per the above **horizontal barplot**, it can be observed that during the year 2019, Tamil Nadu, Maharashtra, Uttar Pradesh, Delhi and others had the highest number of foreign tourists to their state. It can also be inferred that significant revenue were derived from Tourism in these states. Thus during 2020, due to Covid-19 restrictions, there was an enormous decrease in the Tourists which led to a huge blow to the Tourism sector in these states.

Tamil Nadu has the largest tourism industry in India with a percentage share of 21.31% and 21.86% of domestic and foreign tourist visits in the country. Thus the steep decline in tourist traffic in 2020 on the advent of pandemic severely affected the Tourism sector of the state.

## ▼ Total tourists per State/UTs [2019 - 2020]

### ▼ Preparing the Dataset

```
# parsing the dataframe
total_statewise=statewise_data[['States/UTs','Domestic -2019','Foreign - 2019','Domestic -2020','Foreign - 2020']].copy()
# calculating total tourists to States/UTs
total_statewise['Total Tourists']= total_statewise['Domestic -2019']+ total_statewise['Foreign - 2019']+ total_statewise['Domestic -2020']
# dropping columns
total_statewise.drop(['Domestic -2019','Foreign - 2019','Domestic -2020','Foreign - 2020'], axis=1, inplace=True)
# sorting by Total Tourists in descending order
total_statewise.sort_values(by=["Total Tourists"], ascending=False, inplace=True)
total_statewise=total_statewise[total_statewise['States/UTs'].isin(['Grand Total'])==False]
# shifting the decimal places by 6 to the left
total_statewise["Total Tourists"]=total_statewise['Total Tourists']* 10**-6

# fetching the 10 most visited States/UTs
total_statewise=total_statewise.iloc[0:10]

total_statewise.head()
```

	States/UTs	Total Tourists
31	Tamil Nadu	643.611148
34	Uttar Pradesh	627.613568
1	Andhra Pradesh	308.228045
16	Karnataka	306.162132
21	Maharashtra	195.320407

### ▼ Visualizing the Dataset

```
# plotting the bar graph
plt.figure(figsize=(20,8))

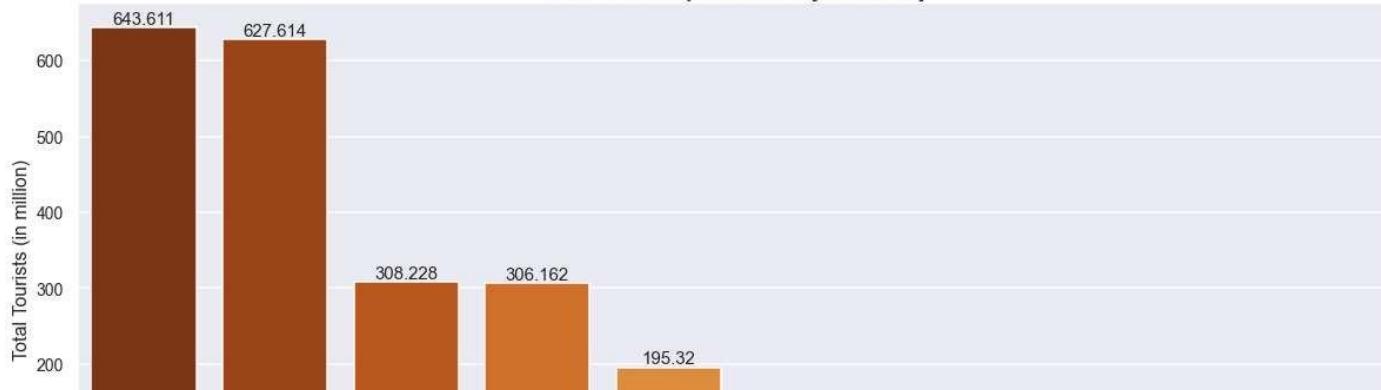
g=sns.barplot(x=total_statewise['States/UTs'], y=total_statewise['Total Tourists'], palette='YlOrBr_r')

plt.title("Total Tourists by States/UTs [2019-2020]", fontsize=18)
plt.xlabel("States/UTs", fontsize=16)
plt.ylabel("Total Tourists (in million)", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)

g.bar_label(g.containers[0], fontsize=15) # labelling the bars

# displaying the chart
plt.show()
```

## Total Tourists by States/UTs [2019-2020]



Thus from the above **bar plot**, it can be inferred that overall Tamil Nadu stood first in the Tourism sector followed by Uttar Pradesh during the years 2019 and 2020 and it is the same even now.

## ▼ Popular monuments during 2019,2020,2021

The dataset used here is the statistics of the number of Domestic and foreign tourists to popular monuments in India during 2019,2020 and 2021 along with their regions. This dataset also includes the growth rate which won't be considered for visualization.

Before visualizing the dataset, we need to prepare the data for Visualization.

### ▼ Top 10 Tourism Destinations visited by foreigners 2019

#### ▼ Preparing the Dataset

```
# fetching data from the csv file
monuments=pd.read_csv('.\dataset\India-Tourism-Statistics-2021-monuments.csv')
monuments.head()
```

Circle	Name of the Monument	Domestic-2019-20	Foreign-2019-20	Domestic-2020-21	Foreign-2020-21	% Growth 2021-21/2019-20-Domestic	% Growth 2021-21/2019-20-Foreign
0	Agra	4429710	645415	1259892	9034	-71.56	-98.60
1	Agra	1627154	386522	371242	2810	-77.18	-99.27
2	Agra	454376	184751	107835	574	-76.27	-99.69
3	Agra	229270	19625	99509	321	-56.60	-98.36

```
# parsing the dataframe
circlewise_tourist=monuments[monuments['Name of the Monument']=='Total'].copy()
circlewise_tourist=circlewise_tourist[['Circle','Foreign-2019-20','Domestic-2019-20']]
# sorting by Foreign-2019-20 in descending order
circlewise_tourist.sort_values(by=['Foreign-2019-20'],ascending=False,inplace= True)

# shifting the decimal places by 3 to the left
circlewise_tourist['Foreign-2019-20']= circlewise_tourist['Foreign-2019-20']* 10**-3
circlewise_tourist['Domestic-2019-20']= circlewise_tourist['Domestic-2019-20']* 10**-3

circlewise_tourist.head()
```

## ▼ Visualizing the Dataset

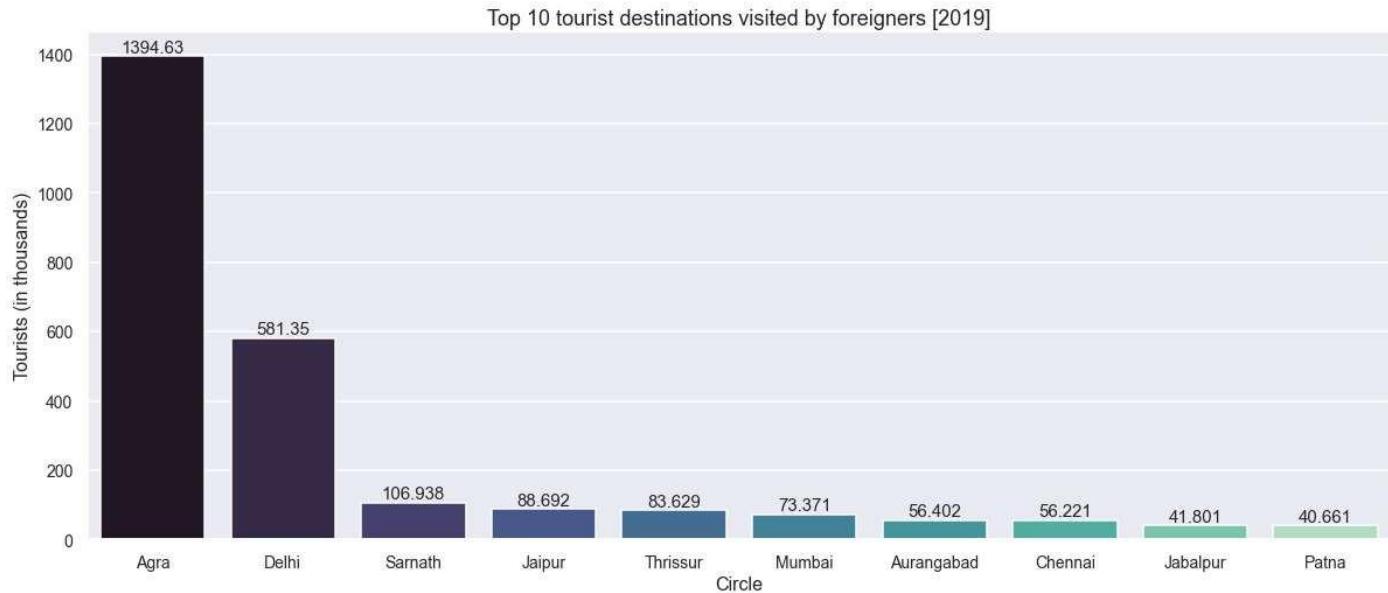
```
# plotting the bar graph
plt.figure(figsize=(20,8))

g=sns.barplot(x=circlewise_tourist['Circle'].iloc[0:10], y=circlewise_tourist['Foreign-2019-20'].iloc[0:10], palette='mako')

plt.title("Top 10 tourist destinations visited by foreigners [2019]", fontsize=18)
plt.xlabel("Circle", fontsize=16)
plt.ylabel("Tourists (in thousands)", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)

g.bar_label(g.containers[0], fontsize=15) # labelling the bars

# displaying the chart
plt.show()
```



As per the above **bar plot**, it can be observed that **Agra** was and is the most popular Tourist spot followed by **Delhi**. This can be due to the fact that Taj Mahal which is India's most recognizable iconic monument, is situated in Agra. And it is also to be noted that since Delhi is just 200km away from Agra and is also quite famous for its unique monuments, it surely does piques the interests of many tourists visiting India.

## ▼ Top 10 Tourism Destinations visited by domestic travellers 2019

### Preparing and Visualizing the Dataset

```
# sorting by Domestic-2019-20 in descending order
circlewise_tourist.sort_values(by=['Domestic-2019-20'], ascending=False, inplace=True)

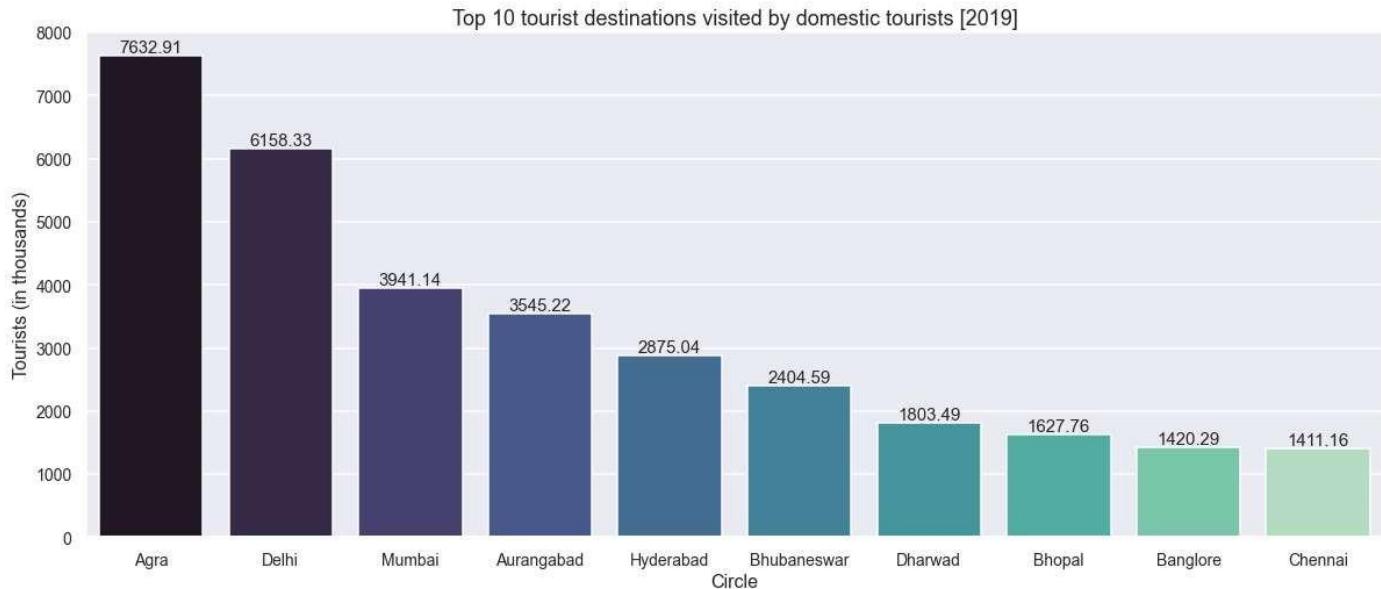
# plotting data on bar graph
plt.figure(figsize=(20,8))

g=sns.barplot(x=circlewise_tourist['Circle'].iloc[0:10], y=circlewise_tourist['Domestic-2019-20'].iloc[0:10], palette='mako')

plt.title("Top 10 tourist destinations visited by domestic tourists [2019]", fontsize=18)
plt.xlabel("Circle", fontsize=16)
plt.ylabel("Tourists (in thousands)", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)

g.bar_label(g.containers[0], fontsize=15) #labelling bars
```

```
# displaying chart
plt.show()
```



A clear pattern similar to the previous graph can be observed in the above **bar plot**. Agra and Delhi are still the most popular tourist destinations even among domestic travellers, while the rest of the popular tourist zones which are popular for their heritage or their infrastructure are pretty much newer destinations which are not so popular among the foreign tourists.

## ▼ Top 10 Monuments visited by foreigners 2019

### ▼ Preparing the Dataset

```
# Parsing the dataframe
monument_tourist=monuments[monuments['Name of the Monument'].isin(['Total','Grand Total'])==False].copy()
# Modifying the contents
monument_tourist['Name of the Monument'] = monument_tourist['Name of the Monument']+("+"+monument_tourist['Circle']+")"

# parsing the dataframe
monument_tourist=monument_tourist[['Name of the Monument','Foreign-2019-20','Domestic-2019-20']]
# sorting by Foreign-2019-20 in descending order
monument_tourist.sort_values(by=['Foreign-2019-20'],ascending=False,inplace= True)

# shifting the decimal places by 3 to the left
monument_tourist['Foreign-2019-20']= monument_tourist['Foreign-2019-20']* 10**-3
monument_tourist['Domestic-2019-20']= monument_tourist['Domestic-2019-20']* 10**-3

monument_tourist.head()
```

	Name of the Monument	Foreign-2019-20	Domestic-2019-20
0	Taj Mahal (Agra)	645.415	4429.710
1	Agra Fort (Agra)	386.522	1627.154
127	Qutub Minar (Delhi)	255.318	2048.146
126	Humayun Tomb (Delhi)	202.288	658.318
2	Fatehpur Sikri (Agra)	184.751	454.376

### Visualizing the Dataset

```
# displaying data on bar chart
plt.figure(figsize=(20,8))

# picking top 10 visited monuments
x=monument_tourist['Name of the Monument'].iloc[0:10]

g=sns.barplot(x=[ str(i) for i in range(1,11)], y=monument_tourist['Foreign-2019-20'].iloc[0:10], palette='rocket')

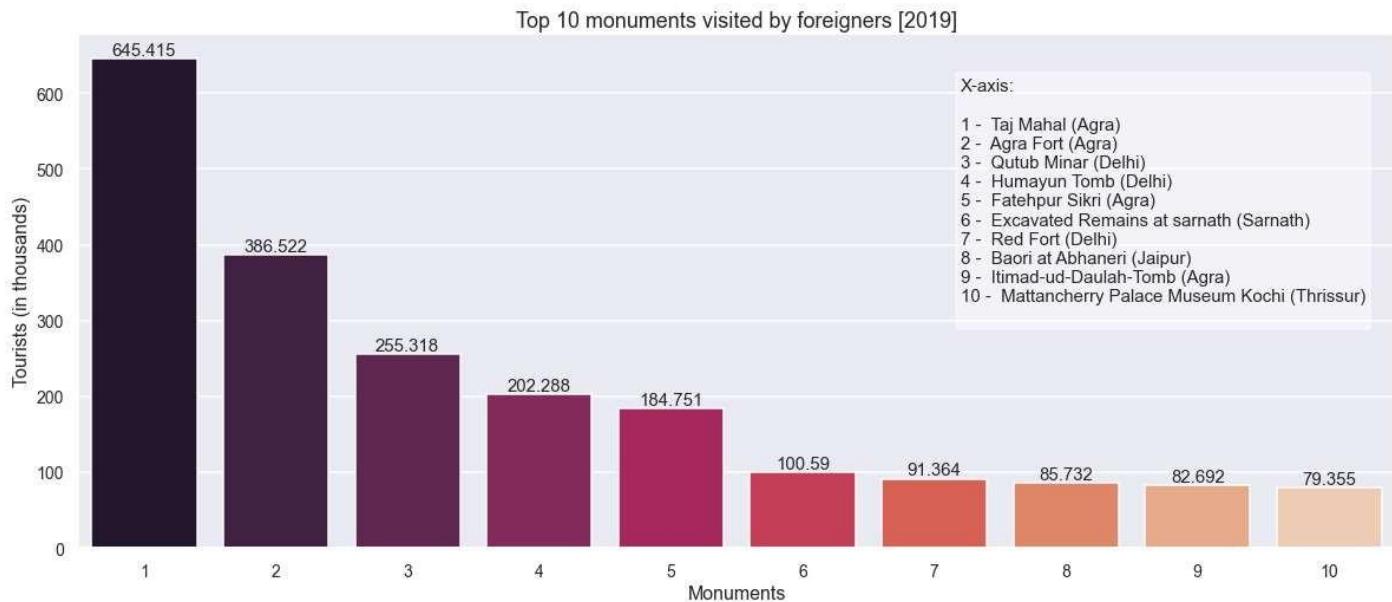
plt.title("Top 10 monuments visited by foreigners [2019]", fontsize=18)
plt.xlabel("Monuments", fontsize=16)
plt.ylabel("Tourists (in thousands)", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)

# Textbox contents
text="X-axis:\n\n"
for i,value in zip(range(1,11),x):
    text= text + str(i) + " - " + value + "\n"

plt.text(6.2, 300, text, fontsize = 15,
        bbox = dict(facecolor = 'white', alpha = 0.5))

g.bar_label(g.containers[0], fontsize=15) #labelling bars

# displaying chart
plt.show()
```



As seen in the popular tourism destinations plot, it is evident from the above **bar plot** that most of the popular monuments visited by the FTAs are situated in Agra and Delhi, famous among them being one of the seven wonders of the world '**Taj Mahal**'.

## ▼ Top 10 Monuments visited by domestic travellers 2019

### ▼ Preparing and Visualizing the Dataset

```
# sorting by Domestic-2019-20 in descending order
monument_tourist.sort_values(by=['Domestic-2019-20'], ascending=False, inplace=True)

# displaying data on bar chart
plt.figure(figsize=(20,8))

# picking top 10 visited monuments
x=monument_tourist['Name of the Monument'].iloc[0:10]

g=sns.barplot(x=[ str(i) for i in range(1,11)], y=monument_tourist['Domestic-2019-20'].iloc[0:10], palette='rocket')
```

```

plt.title("Top 10 monuments visited by domestic tourists [2019]", fontsize=18)
plt.xlabel("Monuments", fontsize=16)
plt.ylabel("Tourists (in thousands)", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)

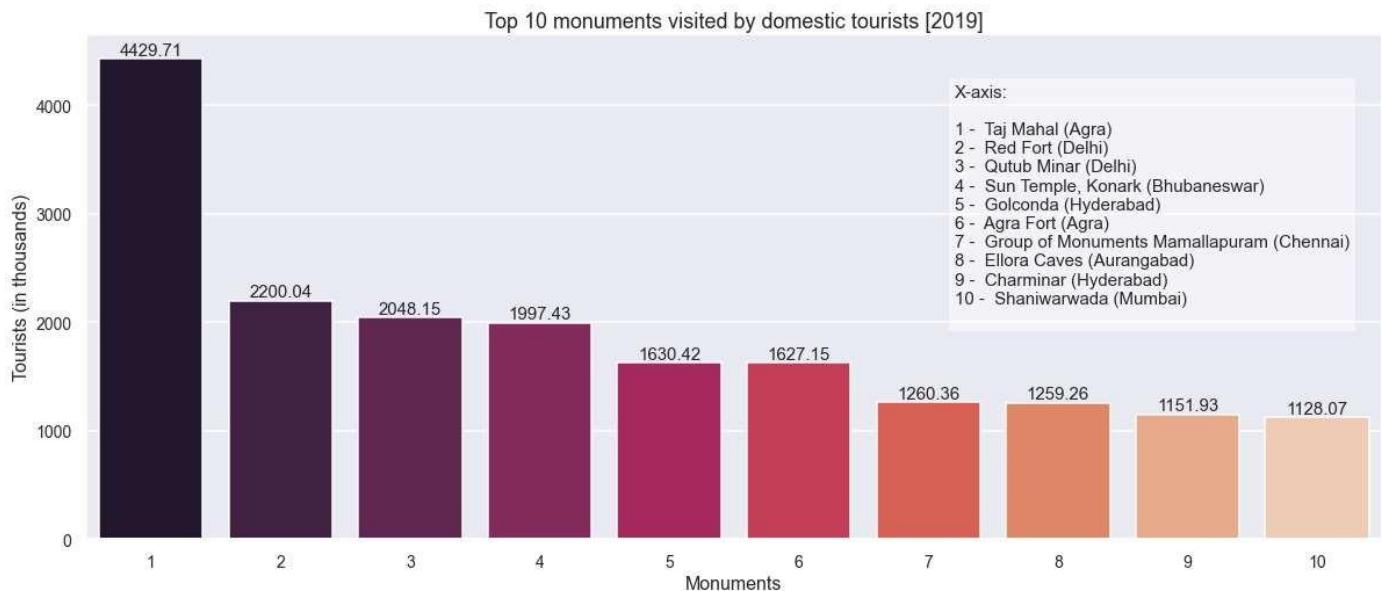
# Textbox contents
text="X-axis:\n\n"
for i,value in zip(range(1,11),x):
    text= text + str(i) + " - " + value + "\n"

plt.text(6.2, 2000, text, fontsize = 15,
        bbox = dict(facecolor = 'white', alpha = 0.5))

g.bar_label(g.containers[0], fontsize=15) #labelling bars

# displaying chart
plt.show()

```



Just like the previous graph, it can be observed in the above **bar plot** as well that the monuments located in Agra and Delhi are popular enough, while it can also be observed that certain monuments which are not so popular have surely piqued the interests of the domestic travellers.

It can be observed from these plots that all the popular places that are mentioned here mostly belong to North India. But it doesn't mean the other parts of India aren't faring well! As we can observe from the popular tourism state plots, it can be concluded that **Tamil Nadu** which is pretty popular for its impressive temples and which receives the highest tourists throughout India is a state from the South!, which is also followed by South Indian states such as Andhra Pradesh and Karnataka.

But **North-East** India lags in tourism as a result of insufficient funds, lack of infrastructure, transportation, lack of alternative means of transport, marketing, boundary issues, terror effect and permit period. If the government were to take the initiative of developing these states, then there would be justice done to the tourism sector of these states. After all, North East India is home to some of the most stunning, diverse and unspoilt scenery in India. From the snow capped peaks of the soaring Himalayan mountains, to the grassy plains of Assam and the steamy forests and majestic waterfalls of Meghalaya – North East India showcases unspoilt nature at its finest!

## Position of Indian Tourism in the world [2001 - 2021]

The dataset used here is the comparison between world tourism and Indian tourism along with its rank in the world from 2001 to 2021. This dataset also includes the percentage share which won't be considered here for visualization.

Before visualizing the dataset, we need to prepare the data for Visualization.

## ▼ Preparing the Dataset

```
# fetching data from the csv file
worldvsindia= pd.read_csv('.\dataset\India-Tourism-Statistics-2001-2019-worldvsindia.csv')
worldvsindia.head()
```

Year	World - Number (in million)	World - % Change	India - Number (in million)	India - % Change	Percentage Share of India	Rank of India
0 2001	683.4	0.0	2.54	-4.2	0.37	51st
1 2002	703.2	2.9	2.38	-6.3	0.34	54th
2 2003	691.0	-1.7	2.73	14.7	0.40	51st
3 2004	762.0	10.3	3.46	26.7	0.45	44th
4 2005	803.4	5.4	3.92	13.3	0.49	43rd

```
# Parsing the dataframe
worldvsindia_rank=worldvsindia[['Year','World - Number (in million)','India - Number (in million)','Rank of India']].copy()
worldvsindia_rank.fillna('NA',inplace=True)
worldvsindia_rank
```

Year	World - Number (in million)	India - Number (in million)	Rank of India
0 2001	683.4	2.54	51st
1 2002	703.2	2.38	54th
2 2003	691.0	2.73	51st
3 2004	762.0	3.46	44th
4 2005	803.4	3.92	43rd
5 2006	846.6	4.45	44th
6 2007	894.0	5.08	41st
7 2008	917.0	5.28	41st
8 2009	883.0	5.17	41st
9 2010	948.0	5.78	42nd
10 2011	994.0	6.31	38th
11 2012	1039.0	6.58	41st
12 2013	1087.0	6.97	41st
13 2014	1137.0	13.11	24th
14 2015	1195.0	13.77	24th
15 2016	1241.0	15.02	26th
16 2017	1332.0	16.81	26th
17 2018	1407.0	17.42	22nd
18 2019	1459.0	17.91	23rd
19 2020	400.0	6.30	NA
20 2021	415.0	5.10	NA

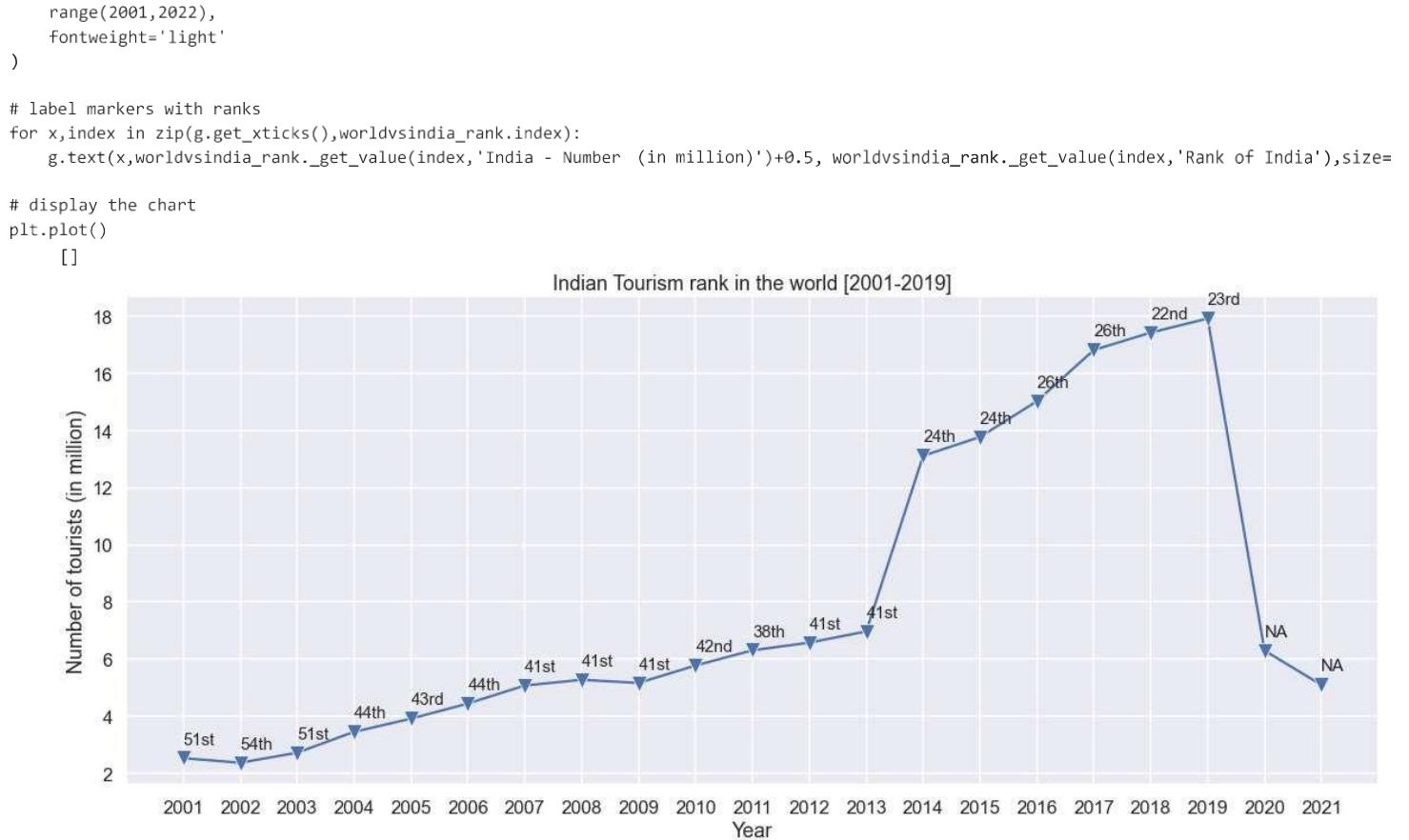
## ▼ Visualizing the Dataset

```
# df=pd.melt(worldvsindia_rank,['Year','Rank of India']).rename(columns={"variable":"World vs India"})

# displaying data on line chart
plt.figure(figsize=(20,8))

# sns.set_context("talk")

g=sns.lineplot(x='Year', y='India - Number (in million)', data=worldvsindia_rank, marker= 'v', markersize=14)
g.set(xlabel="Year",ylabel="Number of tourists (in million)", title="Indian Tourism rank in the world [2001-2019]")
plt.xticks()
```



From the above **line plot**, we can observe that Indian Tourism was improving until 2020 as **COVID-19** restrictions led to a steep decline in tourism. As NRIs were included in the tourism statistics of India ever since 2014, there was a steep increase in the number of tourists to India as well as its position w.r.t Tourism in the world.

Indian tourism developed a lot since the year of **2010** as the investment for the domestic tourism industry and the foreign tourism industry was increased to a great extent. Many other factors like transportation, hotel management system and other necessary tourism components got major upgrades.

Hopefully, now that it's the post pandemic period, India will work towards improving tourism and bring it back to its former glory so it can continue to improve from here.