Forest Fires in Brazil Analysis Using Python

The Forest Fires in Brazil Analysis set has the information about the Forest Fires.

The Data set available from Flexible which is a Third Party Forest Fires in Brazil, and available on Kaggle dataset for free.

Import Library

In [1]:
```
import pandas as pd
```

In [34]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

Uploading Csv fle

In [35]:
```
df = pd.read_excel(r"C:\Users\Syed Arif\Downloads\amazon.xls")
```

Data Preprocessing

.head()

head is used show to the By default = 5 rows in the dataset

In [36]:
```
df.head()
```

Out[36]:

	year	state	month	number	date
0	1998	Acre	Janeiro	0.0	1998-01-01
1	1999	Acre	Janeiro	0.0	1999-01-01
2	2000	Acre	Janeiro	0.0	2000-01-01
3	2001	Acre	Janeiro	0.0	2001-01-01
4	2002	Acre	Janeiro	0.0	2002-01-01

.tail()

tail is used to show rows by Descending order

In [37]:
```
df.tail()
```

Out[37]:

	year	state	month	number	date
6449	2012	Tocantins	Dezembro	128.0	2012-01-01
6450	2013	Tocantins	Dezembro	85.0	2013-01-01
6451	2014	Tocantins	Dezembro	223.0	2014-01-01
6452	2015	Tocantins	Dezembro	373.0	2015-01-01
6453	2016	Tocantins	Dezembro	119.0	2016-01-01

.shape

It show the total no of rows & Column in the dataset

In [38]:
```
df.shape
```

Out[38]:
```
(6454, 5)
```

.Columns

It show the no of each Column

In [39]:
```
df.columns
```

Out[39]:
```
Index(['year', 'state', 'month', 'number', 'date'], dtype='object')
```

.dtypes

This Attribute show the data type of each column

In [40]:
```
df.dtypes
```

Out[40]:
```
year              int64
state            object
month            object
number          float64
date      datetime64[ns]
dtype: object
```

.unique()

In a column, It show the unique value of specific column.

In [41]:
```
df["month"].unique()
```

Out[41]:
```
array(['Janeiro', 'Fevereiro', 'Mar�o', 'Abril', 'Maio', 'Junho', 'Julho',
       'Agosto', 'Setembro', 'Outubro', 'Novembro', 'Dezembro'],
      dtype=object)
```

.nuique()

It will show the total no of unque value from whole data frame

In [42]:
```
df.nunique()
```

Out[42]:
```
year        20
state       23
month       12
number    1479
date        20
dtype: int64
```

.describe()

It show the Count, mean , median etc

In [43]:
```
df.describe()
```

Out[43]:

	year	number
count	6454.000000	6454.000000
mean	2007.461729	108.293163
std	5.746654	190.812242
min	1998.000000	0.000000
25%	2002.000000	3.000000
50%	2007.000000	24.000000
75%	2012.000000	113.000000
max	2017.000000	998.000000

.value_counts

It Shows all the unique values with their count

In [44]:
```
df["state"].value_counts()
```

Out[44]:
```
Rio                 717
Paraiba             478
Mato Grosso         478
Alagoas             240
Acre                239
Sergipe             239
Sao Paulo           239
Santa Catarina      239
Roraima             239
Rondonia            239
Piau                239
Pernambuco          239
Minas Gerais        239
Par?                 239
Maranhao            239
Goias               239
Espirito Santo      239
Distrito Federal    239
Ceara               239
Bahia               239
Amazonas            239
Amapa               239
Tocantins           239
Name: state, dtype: int64
```

.isnull()

It shows the how many null values

In [45]:
```
df.isnull().sum()
```

Out[45]:
```
year      0
state     0
month     0
number    0
date      0
dtype: int64
```

How Many Duplicates are Present Find and remove

In [46]:
```
df[df.duplicated()]
```

Out[46]:

	year	state	month	number	date
259	2017	Alagoas	Janeiro	38.0	2017-01-01
2630	1998	Mato Grosso	Janeiro	0.0	1998-01-01
2650	1998	Mato Grosso	Fevereiro	0.0	1998-01-01
2670	1998	Mato Grosso	Mar�o	0.0	1998-01-01
2690	1998	Mato Grosso	Abril	0.0	1998-01-01
2710	1998	Mato Grosso	Maio	0.0	1998-01-01
3586	1998	Paraiba	Janeiro	0.0	1998-01-01
3606	1998	Paraiba	Fevereiro	0.0	1998-01-01
3621	2013	Paraiba	Fevereiro	9.0	2013-01-01
3626	1998	Paraiba	Mar�o	0.0	1998-01-01
3646	1998	Paraiba	Abril	0.0	1998-01-01
3666	1998	Paraiba	Maio	0.0	1998-01-01
4542	1998	Rio	Janeiro	0.0	1998-01-01
4562	1998	Rio	Fevereiro	0.0	1998-01-01
4582	1998	Rio	Mar�o	0.0	1998-01-01
4585	2001	Rio	Mar�o	0.0	2001-01-01
4590	2006	Rio	Mar�o	8.0	2006-01-01
4602	1998	Rio	Abril	0.0	1998-01-01
4608	2004	Rio	Abril	3.0	2004-01-01
4613	2009	Rio	Abril	1.0	2009-01-01
4622	1998	Rio	Maio	0.0	1998-01-01
4631	2007	Rio	Maio	2.0	2007-01-01
4632	2008	Rio	Maio	0.0	2008-01-01
4645	2001	Rio	Junho	13.0	2001-01-01
4781	1998	Rio	Janeiro	0.0	1998-01-01
4800	2017	Rio	Janeiro	28.0	2017-01-01
4801	1998	Rio	Fevereiro	0.0	1998-01-01
4821	1998	Rio	Mar�o	0.0	1998-01-01
4841	1998	Rio	Abril	0.0	1998-01-01
4861	1998	Rio	Maio	0.0	1998-01-01
4864	2001	Rio	Maio	4.0	2001-01-01
4910	2007	Rio	Julho	7.0	2007-01-01

In [47]:
```
df = df.drop_duplicates()
```

In [48]:
```
df.shape
```

Out[48]:
```
(6422, 5)
```

In [49]:
```
6454 - 6422
```

Out[49]:
```
32
```

Rename Month Name in English

In [50]:
```
df['Month_Name'] = df['month'].map({'Janeiro': 'Jan',
                                    'Fevereiro': 'Feb',
                                    'Mar?o': 'March',
                                    'Abril': 'April',
                                    'Maio': 'May',
                                    'Junho': 'Jun',
                                    'Julho': 'July',
                                     'Agosto': 'Aug',
                                    'Setembro': 'Sep',
                                    'Outubro': 'Oct',
                                    'Novembro': 'Novem',
                                    'Dezembro': 'Dec'})
```

In [51]:
```
df
```

Out[51]:

	year	state	month	number	date	Month_Name
0	1998	Acre	Janeiro	0.0	1998-01-01	Jan
1	1999	Acre	Janeiro	0.0	1999-01-01	Jan
2	2000	Acre	Janeiro	0.0	2000-01-01	Jan
3	2001	Acre	Janeiro	0.0	2001-01-01	Jan
4	2002	Acre	Janeiro	0.0	2002-01-01	Jan
...	...	...	...	...	...	...
6449	2012	Tocantins	Dezembro	128.0	2012-01-01	Dec
6450	2013	Tocantins	Dezembro	85.0	2013-01-01	Dec
6451	2014	Tocantins	Dezembro	223.0	2014-01-01	Dec
6452	2015	Tocantins	Dezembro	373.0	2015-01-01	Dec
6453	2016	Tocantins	Dezembro	119.0	2016-01-01	Dec

6422 rows × 6 columns

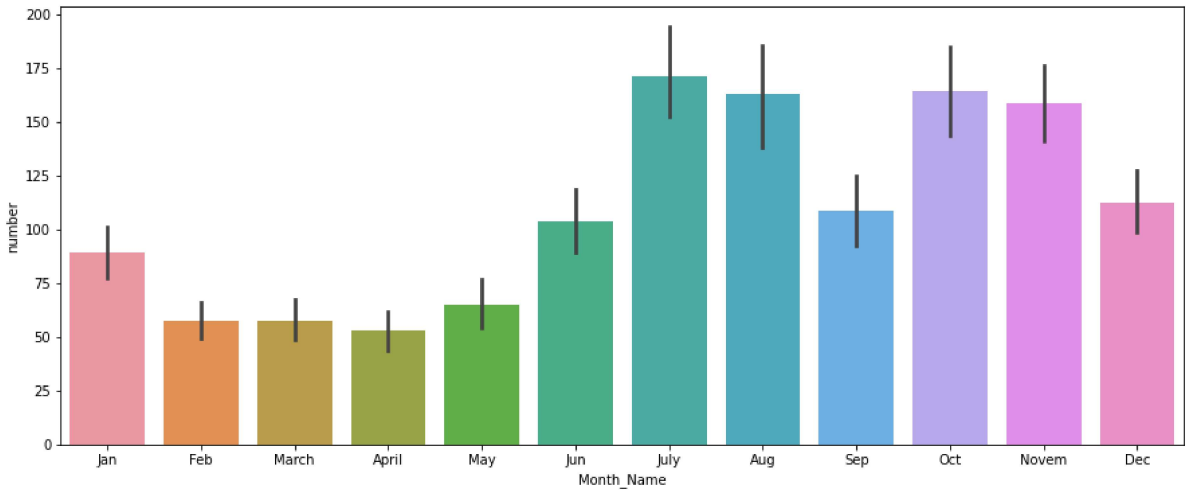
In Which Month Maximum Number of forest fire

In [63]:
```
df.groupby('Month_Name')['number'].sum()
```

Out[63]:
```
Month_Name
April    28184.770
Aug      88050.435
Dec      57535.480
Feb      30839.050
Jan      47681.844
July     92319.113
Jun      55997.675
March    30709.405
May      34725.363
Novem    85508.054
Oct      88681.579
Sep      58578.305
Name: number, dtype: float64
```

In [64]:
```
plt.figure(figsize = (15,6))
sns.barplot(x = "Month_Name" , y = 'number', data = df)
```

Out[64]:
```
<AxesSubplot:xlabel='Month_Name', ylabel='number'>
```



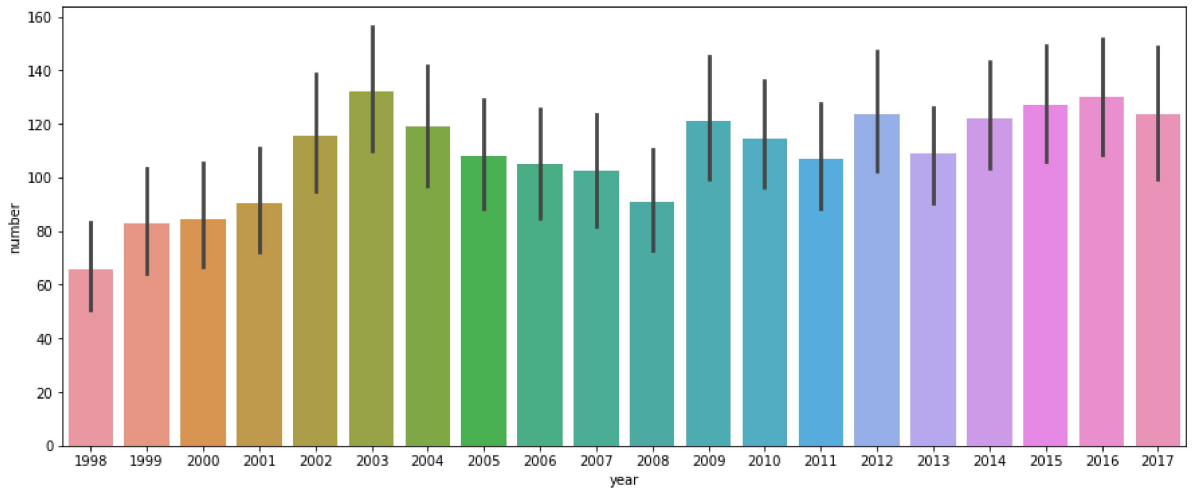
In Which Year Maximum Number of forest fire cases reported

In [65]:
```
df.groupby('year')['number'].sum()
```

Out[65]:
```
year
1998    20013.971
1999    26882.821
2000    27351.251
2001    29054.612
2002    37390.600
2003    42760.674
2004    38450.163
2005    35004.965
2006    33824.161
2007    33028.413
2008    29378.964
2009    39116.178
2010    37037.449
2011    34633.545
2012    40084.860
2013    35137.118
2014    39621.183
2015    41208.292
2016    42212.229
2017    36619.624
Name: number, dtype: float64
```

In [66]:
```
plt.figure(figsize = (15,6))
sns.barplot(x = "year" , y = 'number', data = df)
```

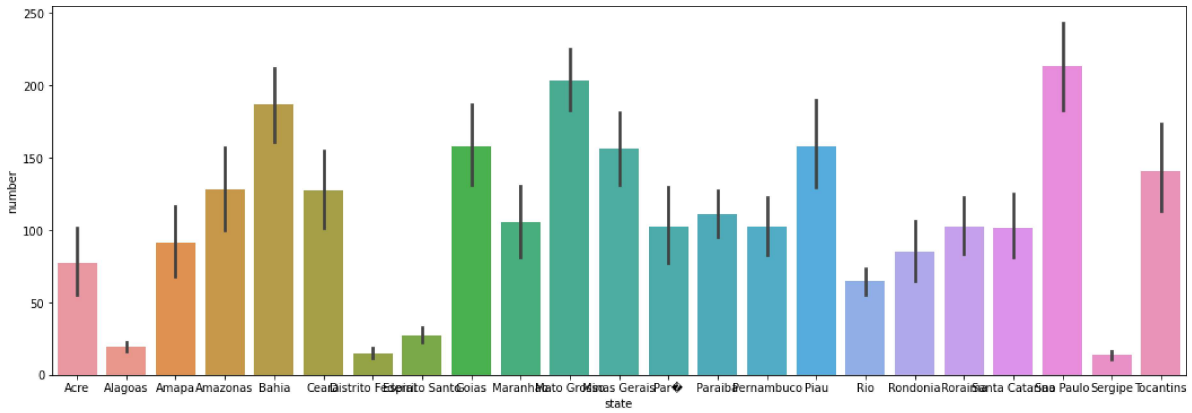
Out[66]:
```
<AxesSubplot:xlabel='year', ylabel='number'>
```



In Which State Maximum Number of forest fire cases reported

In [67]:
```
plt.figure(figsize = (18,6))
sns.barplot(x = "state" , y = 'number', data = df)
```

Out[67]: <AxesSubplot:xlabel='state', ylabel='number'>



Find Total Number of Fires were Reported in Amazonas

In [68]:
```
df[df["state"] == "Amazonas"]["number"].sum()
```

Out[68]: 30650.129

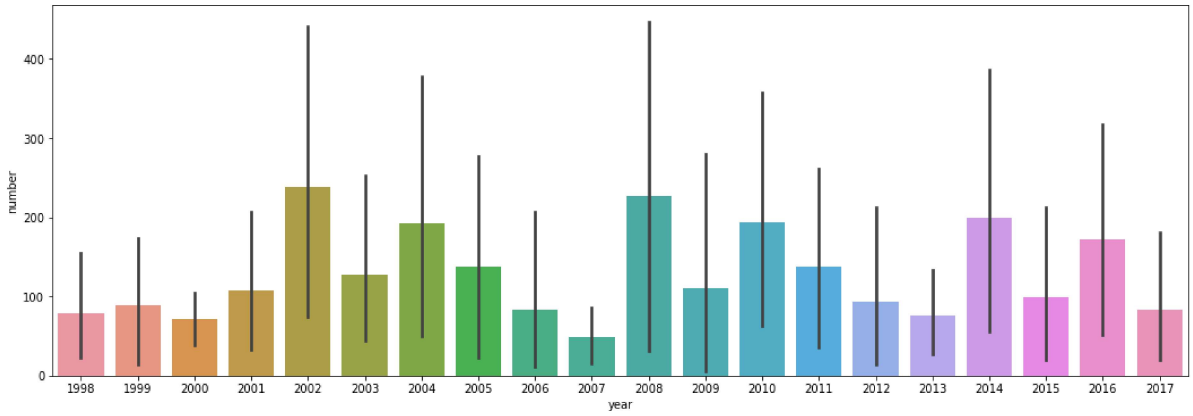
Display Number of Fires were reported in Amazonas

In [70]:
```
df=df[df["state"] =="Amazonas"]
df.groupby("year")["number"].sum()
```

Out[70]:
```
year
1998     946.000
1999    1061.000
2000     853.000
2001    1297.000
2002    2852.000
2003    1524.268
2004    2298.207
2005    1657.128
2006     997.640
2007     589.601
2008    2717.000
2009    1320.601
2010    2324.508
2011    1652.538
2012    1110.641
2013     905.217
2014    2385.909
2015    1189.994
2016    2060.972
2017     906.905
Name: number, dtype: float64
```

In [71]:
```
plt.figure(figsize = (18,6))
sns.barplot(x = "year" , y = 'number', data = df)
```

Out[71]:
```
<AxesSubplot:xlabel='year', ylabel='number'>
```



Display Number of Fires were reported in Amazonas (Day wise)

In [82]:
```
df1 =df[df["state"] =="Amazonas"]
```

In [101]:
```
day = df1.groupby(df["date"].dt.dayofweek).sum().number

import calendar
day.index = [calendar.day_name[x] for x in range (0,7)]
day = day.reset_index()
day
```

Out[101]:

	index	number
0	Monday	1886.601
1	Tuesday	6474.217
2	Wednesday	3910.177
3	Thursday	5754.802
4	Friday	5446.480
5	Saturday	4162.666
6	Sunday	3015.186

In [ ]:
```

```