```
#import dataset
import pandas as pd
df_f0=pd.read_excel('/content/0-14 female.xlsx')
df_f0.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 0-14, female | SP.POP.0014.FE.IN | India | IND | 183993964 | 182951247 | 181831596 | 1 |

```
#import dataset 0-14 male
df_m0=pd.read_excel('/content/0-14 male.xlsx')
df_m0.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 0-14, male | SP.POP.0014.MA.IN | India | IND | 202296224 | 201020492 | 199649050 | 1 |

```
#load 15-64 male and female dataset
df_f15=pd.read_excel('/content/15-64 female.xlsx')
df_f15.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 15-64, female | SP.POP.1564.FE.IN | India | IND | 412907483 | 420161099 | 427477653 | 4 |

```
df_m15=pd.read_excel('/content/15-64 male.xlsx')
df_m15.head()
```

| | Country Name | Country Code | Series Name | Series Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | |
|---|---|---|---|---|---|---|---|---|
| 0 | India | IND | Population ages 15-64, male | SP.POP.1564.MA.IN | 443434941 | 451259170 | 459164278 | 4 |

```
df_f65=pd.read_excel('/content/65 &above female.xlsx')
df_f65.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | [Y |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 65 and above, female | SP.POP.65UP.FE.IN | India | IND | 37109634 | 38613415 | 40217956 | 4 |

```
df_m65=pd.read_excel('/content/65 & above male.xlsx')
df_m65.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | [ |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 65 and above, male | SP.POP.65UP.MA.IN | India | IND | 32534945 | 34019075 | 35603764 | 3 |

```
df_t_0=pd.read_excel('/content/0-14 total.xlsx')
df_t_0.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | [YI |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 0-14, total | SP.POP.0014.TO | India | IND | 386290188 | 383971739 | 381480646 | 378 |

```
df_t_15=pd.read_excel('/content/15-64 total.xlsx')
df_t_15.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | [YI |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 15-64, total | SP.POP.1564.TO | India | IND | 856342424 | 871420269 | 886641930 | 901 |

```
df_t_65=pd.read_excel('/content/Age 65 & above total.xlsx')
df_t_65.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | [YR2 |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 65 and above, total | SP.POP.65UP.TO | India | IND | 69644579 | 72632490 | 75821720 | 791 |

```
#concat all loaded dataset
df = pd.concat([df_f0, df_m0, df_f15, df_m15, df_f65, df_m65, df_t_0, df_t_15, df_t_65], axis=0)
df.head()
```

| | Series Name | Series Code | Country Name | Country Code | 2014 [YR2014] | 2015 [YR2015] | 2016 [YR2016] | |
|---|---|---|---|---|---|---|---|---|
| 0 | Population ages 0-14, female | SP.POP.0014.FE.IN | India | IND | 183993964 | 182951247 | 181831596 | 1 |
| 0 | Population ages 0-14, male | SP.POP.0014.MA.IN | India | IND | 202296224 | 201020492 | 199649050 | 1 |
| 0 | Population ages 15-64, female | SP.POP.1564.FE.IN | India | IND | 412907483 | 420161099 | 427477653 | 4 |
| 0 | Population ages 15-64, male | SP.POP.1564.MA.IN | India | IND | 443434941 | 451259170 | 459164278 | 4 |
| 0 | Population ages 65 and above, female | SP.POP.65UP.FE.IN | India | IND | 37109634 | 38613415 | 40217956 | |

Next steps: Generate code with df View recommended plots New interactive sheet

```
df.columns
```

```
Index(['Series Name', 'Series Code', 'Country Name', 'Country Code',
       '2014 [YR2014]', '2015 [YR2015]', '2016 [YR2016]', '2017 [YR2017]',
       '2018 [YR2018]', '2019 [YR2019]', '2020 [YR2020]', '2021 [YR2021]',
       '2022 [YR2022]', '2023 [YR2023]'],
      dtype='object')
```

```
df.columns=['Series Name', 'Series Code', 'Country Name', 'Country Code',
       '2014', '2015', '2016', '2017','2018', '2019', '2020', '2021',
       '2022', '2023']
```

```
#drop first 4 columns
df=df.drop(df.columns[0:4],axis=1)
df.head()
```

| | 2014 | 2015 | 2016 | 2017 | 2018 | 2019 | 2020 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 183993964 | 182951247 | 181831596 | 180541801 | 179200843 | 177886985 | 176475236 | 1750 |
| 0 | 202296224 | 201020492 | 199649050 | 198093854 | 196462625 | 194822123 | 193049643 | 1912 |
| 0 | 412907483 | 420161099 | 427477653 | 434781814 | 441709500 | 448232319 | 454575427 | 4604 |
| 0 | 443434941 | 451259170 | 459164278 | 467043895 | 474541800 | 481642958 | 488513474 | 4948 |
| 0 | 37109634 | 38613415 | 40217956 | 41914690 | 43698562 | 45559944 | 47391566 | 487 |

Next steps: Generate code with df View recommended plots New interactive sheet

```
df=df.transpose()
df
```

| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
|---|---|---|---|---|---|---|---|---|
| 2014 | 183993964 | 202296224 | 412907483 | 443434941 | 37109634 | 32534945 | 386290188 | 856 |
| 2015 | 182951247 | 201020492 | 420161099 | 451259170 | 38613415 | 34019075 | 383971739 | 871 |
| 2016 | 181831596 | 199649050 | 427477653 | 459164278 | 40217956 | 35603764 | 381480646 | 886 |
| 2017 | 180541801 | 198093854 | 434781814 | 467043895 | 41914690 | 37281347 | 378635655 | 901 |
| 2018 | 179200843 | 196462625 | 441709500 | 474541800 | 43698562 | 39045735 | 375663468 | 916 |
| 2019 | 177886985 | 194822123 | 448232319 | 481642958 | 45559944 | 40885983 | 372709109 | 929 |
| 2020 | 176475236 | 193049643 | 454575427 | 488513474 | 47391566 | 42612350 | 369524878 | 943 |
| 2021 | 175075695 | 191276108 | 460474670 | 494806597 | 48729429 | 43841398 | 366351802 | 955 |
| 2022 | 173742919 | 189564729 | 465941614 | 500684854 | 50207224 | 45281873 | 363307647 | 966 |
| 2023 | 172455965 | 187882029 | 471471101 | 506719577 | 52259267 | 47281658 | 360337993 | 978 |

Next steps: Generate code with df View recommended plots New interactive sheet

```
print(df.columns)
```

```
Index([0, 0, 0, 0, 0, 0, 0, 0, 0], dtype='int64')
```

```
#to change column name
df.columns = ['0-14, Female', '0-14, Male', '15-64, Female', 'T15-64, male','65 and above,Female',
              '65 and above, male','0-14, total','15-64, Total','65 and above, Total']
df.head()
```

| | 0-14, Female | 0-14, Male | 15-64, Female | T15-64, male | 65 and above,Female | 65 and above, male | 0-14, total |
|---|---|---|---|---|---|---|---|
| **2014** | 183993964 | 202296224 | 412907483 | 443434941 | 37109634 | 32534945 | 386290188 |
| **2015** | 182951247 | 201020492 | 420161099 | 451259170 | 38613415 | 34019075 | 383971739 |
| **2016** | 181831596 | 199649050 | 427477653 | 459164278 | 40217956 | 35603764 | 381480646 |
| **2017** | 180541801 | 198093854 | 434781814 | 467043895 | 41914690 | 37281347 | 378635655 |
| **2018** | 179200843 | 196462625 | 441709500 | 474541800 | 43698562 | 39045735 | 375663468 |

Next steps: Generate code with df | View recommended plots | New interactive sheet

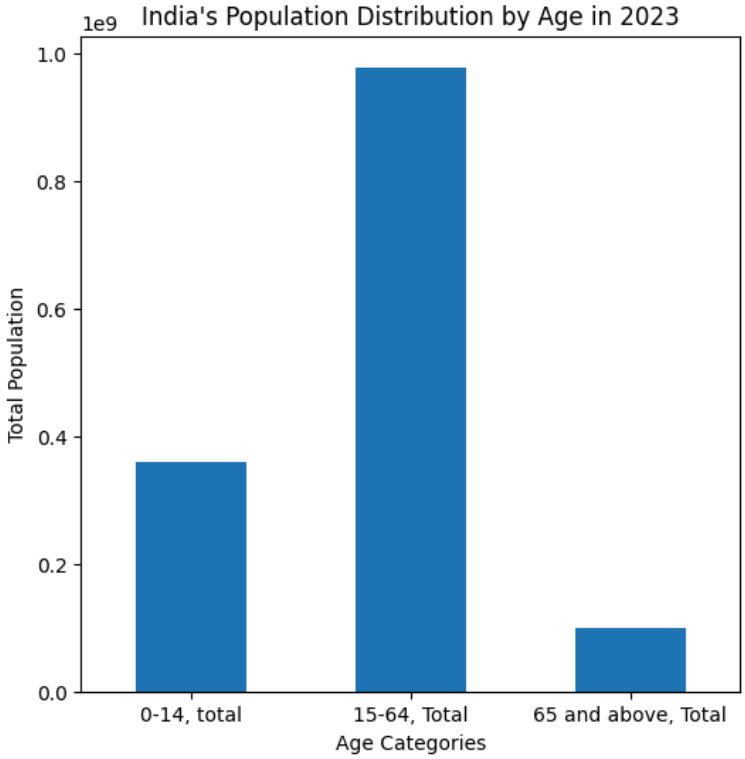
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 10 entries, 2014 to 2023
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   0-14, Female         10 non-null     int64
 1   0-14, Male           10 non-null     int64
 2   15-64, Female        10 non-null     int64
 3   T15-64, male         10 non-null     int64
 4   65 and above,Female  10 non-null     int64
 5   65 and above, male   10 non-null     int64
 6   0-14, total          10 non-null     int64
 7   15-64, Total         10 non-null     int64
 8   65 and above, Total  10 non-null     int64
dtypes: int64(9)
memory usage: 1.1+ KB
```

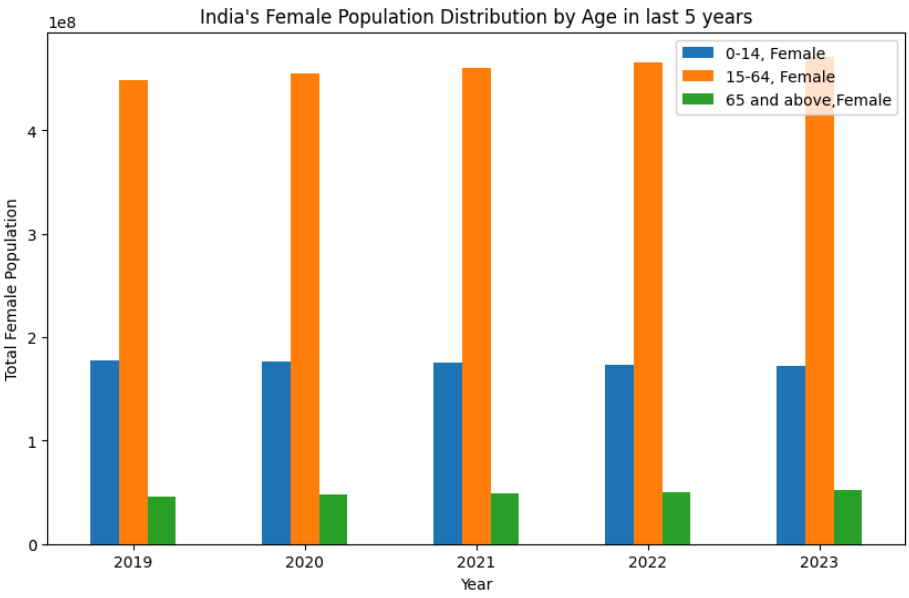
```
#bar chart to compare total of 3 categories
import pandas as pd
import matplotlib.pyplot as plt

#extract total of 2023 for 3 categories
plot_2023= df.loc['2023', ['0-14, total', '15-64, Total', '65 and above, Total']]

#create bar chart
plot_2023.plot(kind='bar', figsize=(6, 6))
plt.title("India's Population Distribution by Age in 2023")
plt.xlabel('Age Categories')
plt.ylabel('Total Population')
plt.xticks(rotation=0)
plt.show()
```




```
#plot bar chart to compare no of female in last 5 years
import pandas as pd
import matplotlib.pyplot as plt
plot_22=df.loc[['2019','2020','2021','2022','2023'], ['0-14, Female','15-64, Female','65 and above,Female']]
plot_22.plot(kind='bar', figsize=(10, 6))
plt.title("India's Female Population Distribution by Age in last 5 years")
plt.xlabel('Year')
plt.ylabel('Total Female Population')
plt.xticks(rotation=0)
plt.show()
```



```
#plot bar chart to compare no of male in last 5 years
import pandas as pd
import matplotlib.pyplot as plt
plot_22=df.loc[['2019','2020','2021','2022','2023'], ['0-14, Male', 'T15-64, male',
              '65 and above, male']]
plot_22.plot(kind='bar', figsize=(10, 6))
plt.title("India's male Population Distribution by Age in last 5 years")
plt.xlabel('Year')
plt.ylabel('Total Male Population')
plt.xticks(rotation=0)
plt.show()
```