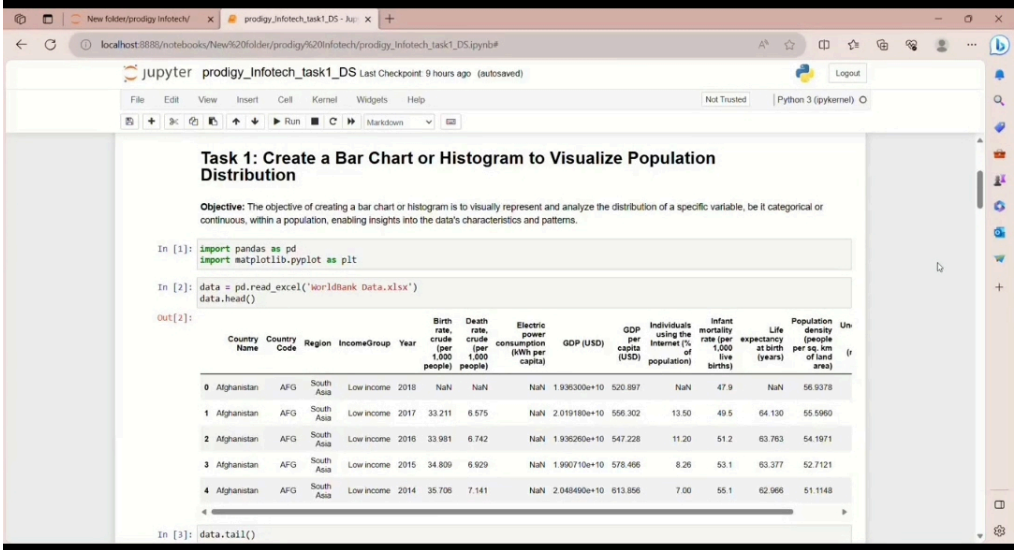


Task 1: Create a Bar Chart or Histogram to Visualize Population Distribution

Objective: The objective of creating a bar chart or histogram is to visually represent and analyze the distribution of a specific variable, be it categorical or continuous, within a population, enabling insights into the data's characteristics and patterns.

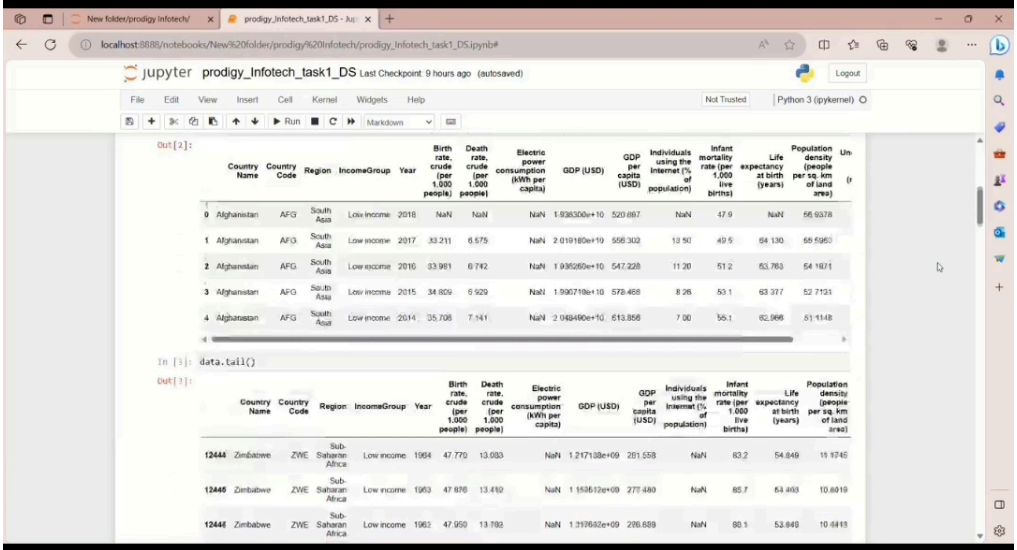
In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

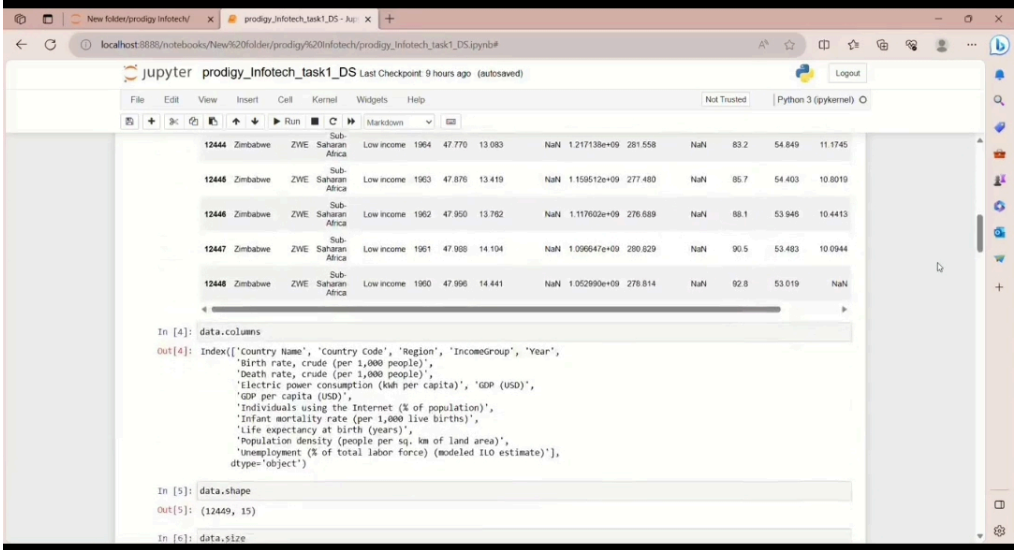
In [2]:
```
data = pd.read_excel('WorldBank Data.xlsx')
data.head()
```

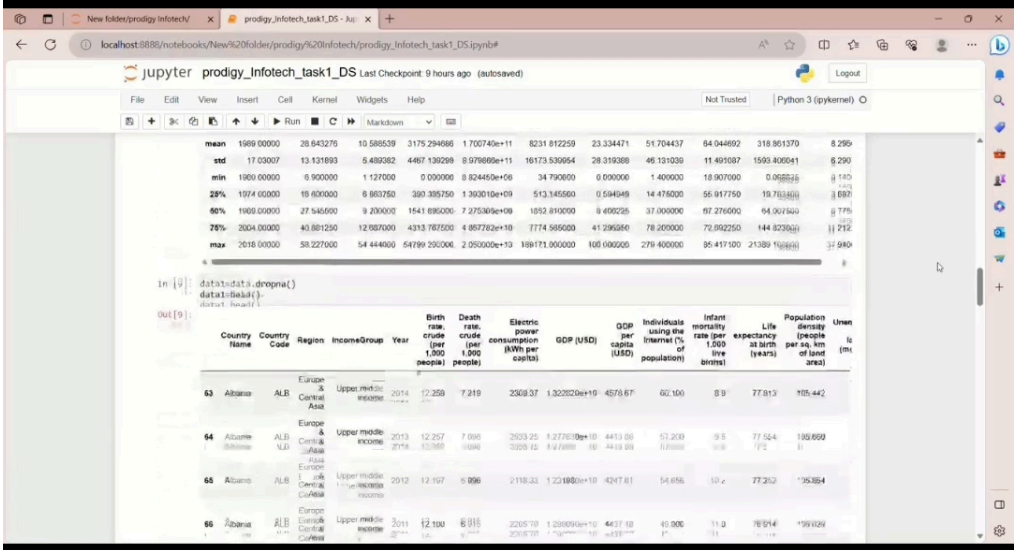
Out[2]:

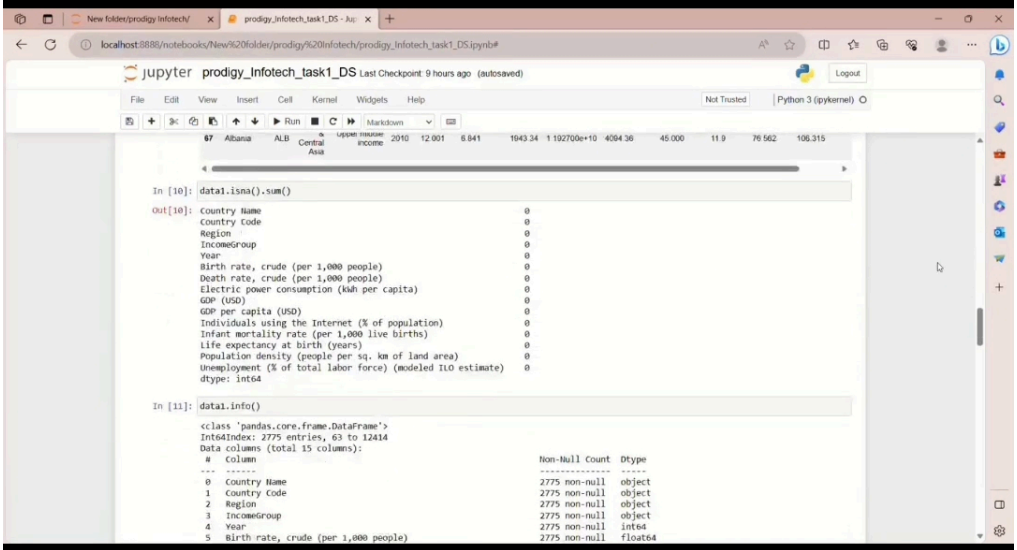
	Country Name	Country Code	Region	IncomeGroup	Year	Birth rate, crude (per 1,000 people)	Death rate, crude (per 1,000 people)	Electric power consumption (kWh per capita)	GDP (USD)	GDP per capita (USD)	Individuals using the Internet (% of population)	Infant mortality rate (per 1,000 live births)	Life expectancy at birth (years)	Population density (people per sq. km of land area)	Un (r
0	Afghanistan	AFG	South Asia	Low income	2018	NaN	NaN	NaN	1.936300e+10	520.897	NaN	47.9	NaN	56.9378	
1	Afghanistan	AFG	South Asia	Low income	2017	33.211	6.575	NaN	2.019180e+10	556.302	13.50	49.5	64.130	55.5960	
2	Afghanistan	AFG	South Asia	Low income	2016	33.981	6.742	NaN	1.936260e+10	547.228	11.20	51.2	63.763	54.1971	
3	Afghanistan	AFG	South Asia	Low income	2015	34.809	6.929	NaN	1.990710e+10	578.466	8.26	53.1	63.377	52.7121	
4	Afghanistan	AFG	South Asia	Low income	2014	35.706	7.141	NaN	2.048490e+10	613.856	7.00	55.1	62.966	51.1148	

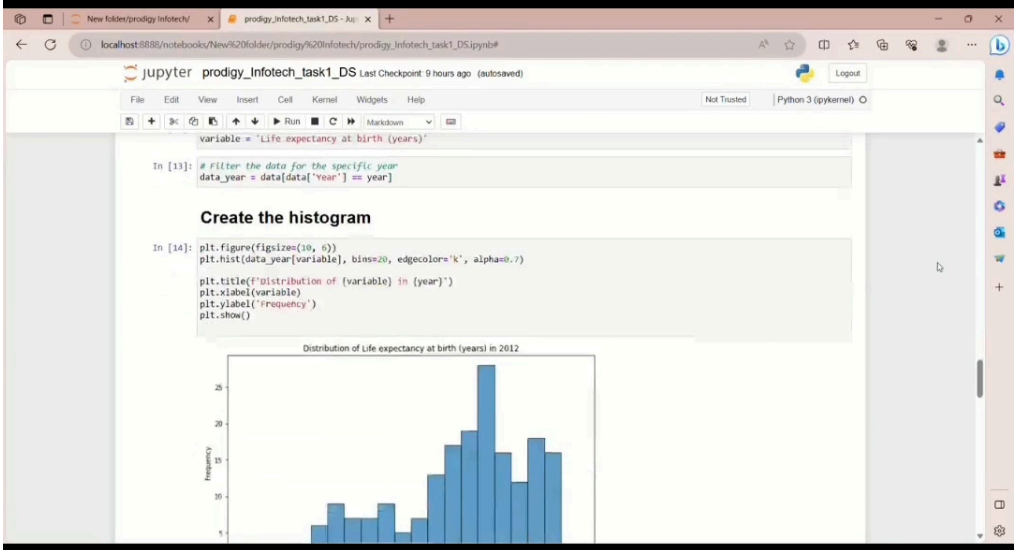
In [3]:
```
data.tail()
```

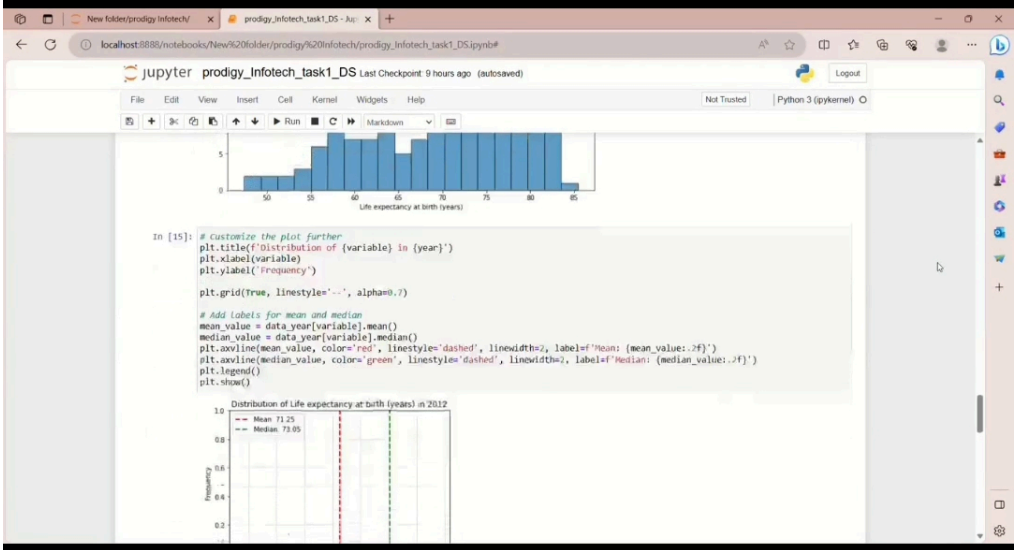

jupyter prodigy_Infotech_task1_DS Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
Out[2]:
Country Name Country Code Region IncomeGroup Year Birth rate, crude (per 1,000 people) Death rate, crude (per 1,000 people) Electric power consumption (kWh per capita) GDP (USD) GDP per capita (USD) Individuals using the Internet (% of population) Infant mortality rate (per 1,000 live births) Life expectancy at birth (years) Population density (people per sq. km of land area)
0 Afghanistan AFG South Asia Low income 2016 NaN NaN NaN 1.938300e+10 520.887 NaN 47.9 NaN 56.9378
1 Afghanistan AFG South Asia Low income 2017 33.211 6.575 NaN 2.019180e+10 556.302 13.50 49.5 64.130 55.5960
2 Afghanistan AFG South Asia Low income 2016 33.981 6.742 NaN 1.935260e+10 547.228 11.20 51.2 63.763 54.1671
3 Afghanistan AFG South Asia Low income 2015 34.809 6.929 NaN 1.990710e+10 578.468 8.26 53.1 63.377 52.7131
4 Afghanistan AFG South Asia Low income 2014 35.708 7.141 NaN 2.048490e+10 613.856 7.00 55.1 62.966 51.1148
In [3]: data.tail()
Out[3]:
Country Name Country Code Region IncomeGroup Year Birth rate, crude (per 1,000 people) Death rate, crude (per 1,000 people) Electric power consumption (kWh per capita) GDP (USD) GDP per capita (USD) Individuals using the Internet (% of population) Infant mortality rate (per 1,000 live births) Life expectancy at birth (years) Population density (people per sq. km of land area)
12444 Zimbabwe ZWE Sub-Saharan Africa Low income 1964 47.770 13.083 NaN 1.217138e+09 281.558 NaN 83.2 54.849 15.9746
12445 Zimbabwe ZWE Sub-Saharan Africa Low income 1963 47.876 13.419 NaN 1.153612e+09 277.480 NaN 85.7 54.493 15.8019
12446 Zimbabwe ZWE Sub-Saharan Africa Low income 1962 47.950 13.783 NaN 1.117682e+09 286.689 NaN 88.1 53.849 15.6413

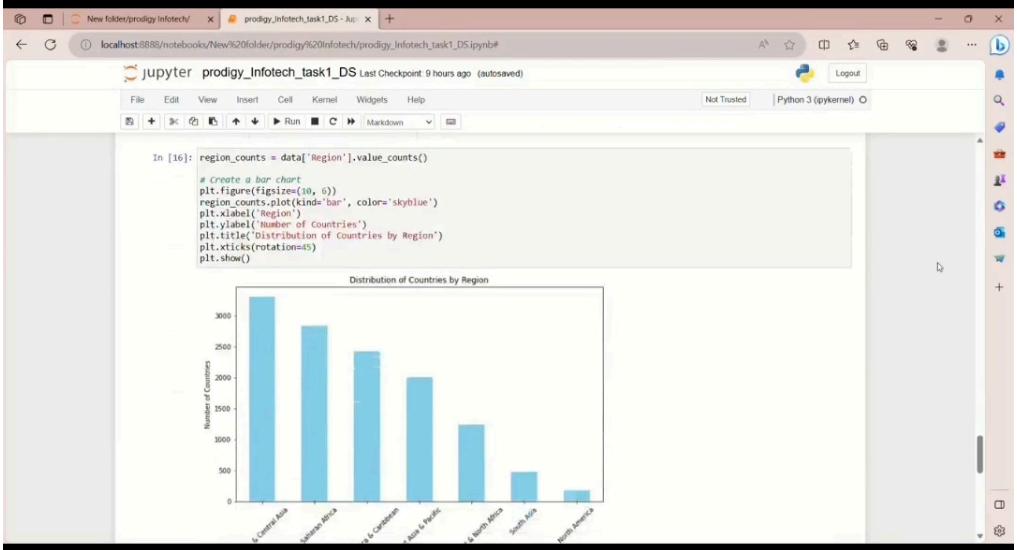

New folder/prodigy Infotech/
prodigy_Infotech_task1_DS - Jup
localhost:8888/notebooks/New%20folder/prodigy%20Infotech/prodigy_Infotech_task1_DS.ipynb#
jupyter prodigy_Infotech_task1_DS Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
12444 Zimbabwe ZWE Sub-Saharan Africa Low income 1964 47.770 13.083 NaN 1.217138e+09 281.558 NaN 83.2 54.849 11.1745
12445 Zimbabwe ZWE Sub-Saharan Africa Low income 1963 47.876 13.419 NaN 1.159512e+09 277.480 NaN 85.7 54.403 10.8019
12446 Zimbabwe ZWE Sub-Saharan Africa Low income 1962 47.950 13.762 NaN 1.117602e+09 276.689 NaN 88.1 53.946 10.4413
12447 Zimbabwe ZWE Sub-Saharan Africa Low income 1961 47.988 14.104 NaN 1.096647e+09 280.829 NaN 90.5 53.483 10.0944
12448 Zimbabwe ZWE Sub-Saharan Africa Low income 1960 47.996 14.441 NaN 1.052890e+09 276.814 NaN 92.8 53.019 NaN
In [4]: data.columns
Out[4]: Index(['Country Name', 'Country Code', 'Region', 'IncomeGroup', 'Year',
'Birth rate, crude (per 1,000 people)',
'Death rate, crude (per 1,000 people)',
'Electric power consumption (kWh per capita)', 'GDP (USD)',
'GDP per capita (USD)',
'Individuals using the Internet (% of population)',
'Infant mortality rate (per 1,000 live births)',
'Life expectancy at birth (years)',
'Population density (people per sq. km of land area)',
'Unemployment (% of total labor force) (modeled ILO estimate)'],
dtype='object')
In [5]: data.shape
Out[5]: (12449, 15)
In [6]: data.size

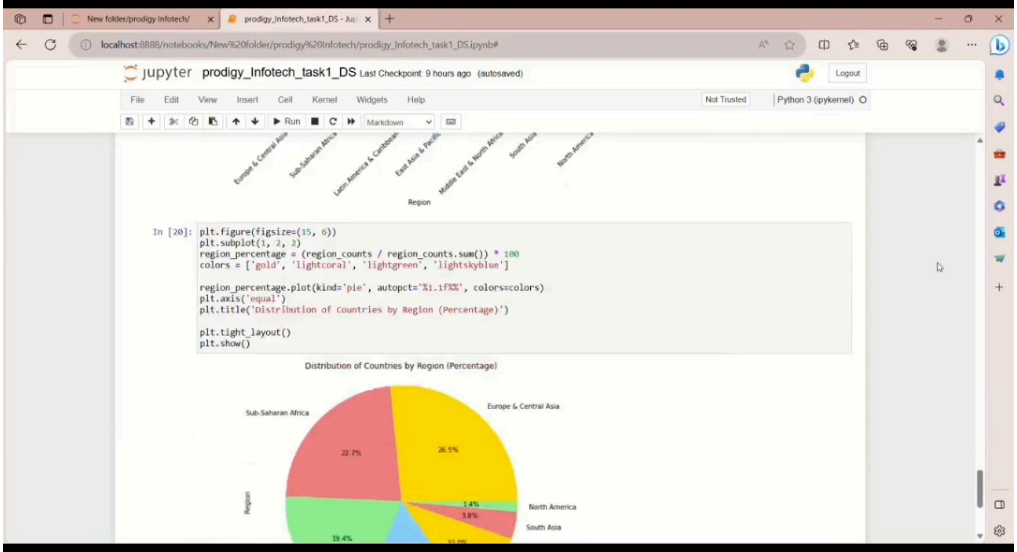

jupyter prodigy_Infotech_task1_DS Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
mean 1989.00000 28.643276 10.588539 3175.294686 1.700740e+11 8231.812259 23.334471 51.704437 64.044692 318.861370 8.295
std 17.03007 13.131893 5.489382 4467.139298 8.979868e+11 16173.539954 28.319388 46.131039 11.491087 1593.406041 6.290
min 1960.00000 0.900000 1.127000 0.000000 8.824450e+08 34.790600 0.000000 1.400000 18.907000 0.068535 0.145
25% 1974.00000 15.600000 6.863750 390.385750 1.393010e+09 513.145560 0.594949 14.476000 56.917750 19.783400 3.887
50% 1989.00000 27.545500 9.200000 1541.695000 7.275309e+09 1852.810000 9.468285 37.000000 67.276000 64.007500 8.775
75% 2004.00000 40.881250 12.687000 4313.787500 4.857782e+10 7774.565000 41.296980 78.200000 72.892250 144.823000 11.212
max 2018.00000 58.227000 54.444000 54799.290006 2.050000e+13 189171.000000 100.000000 279.400000 85.417100 21389.100000 37.980
In [9]: data=data.dropna()
data1=data().
data1.head()
Out[9]:
Country Name Country Code Region IncomeGroup Year Birth rate, crude (per 1,000 people) Death rate, crude (per 1,000 people) Electric power consumption (kWh per capita) GDP (USD) GDP per capita (USD) Individuals using the Internet (% of population) Infant mortality rate (per 1,000 live births) Life expectancy at birth (years) Population density (people per sq. km of land area)
63 Albania ALB Europe & Central Asia Upper middle income 2014 12.258 7.219 2309.37 1.322820e+10 4578.67 60.100 8.9 77.813 105.442
64 Albania ALB Europe & Central Asia Upper middle income 2013 12.257 7.096 2503.25 1.277630e+10 4413.09 57.200 9.5 77.554 105.660
65 Albania ALB Europe & Central Asia Upper middle income 2012 12.197 6.996 2118.33 1.231980e+10 4247.81 54.656 10.2 77.252 105.954
66 Albania ALB Europe & Central Asia Upper middle income 2011 12.100 6.915 2205.78 1.208959e+10 4237.18 49.906 11.0 76.914 106.029

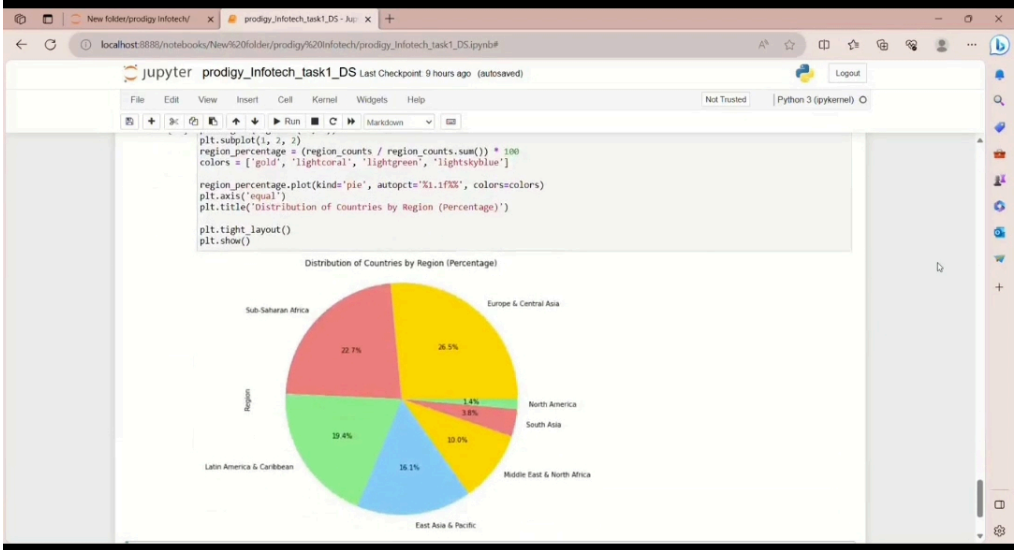

New folder/prodigy Infotech/
prodigy_Infotech_task1_DS - Jup
localhost:8888/notebooks/New%20folder/prodigy%20Infotech/prodigy_Infotech_task1_DS.ipynb#
jupyter prodigy_Infotech_task1_DS Last Checkpoint: 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
67 Albania ALB Europe & Central Asia Upper middle income 2010 12.001 6.841 1643.34 1.192700e+10 4094.36 45.000 11.9 76.562 106.315
In [10]: data1.isna().sum()
Out[10]:
Country Name 0
Country Code 0
Region 0
IncomeGroup 0
Year 0
Birth rate, crude (per 1,000 people) 0
Death rate, crude (per 1,000 people) 0
Electric power consumption (kWh per capita) 0
GDP (USD) 0
GDP per capita (USD) 0
Individuals using the Internet (% of population) 0
Infant mortality rate (per 1,000 live births) 0
Life expectancy at birth (years) 0
Population density (people per sq. km of land area) 0
Unemployment (% of total labor force) (modeled ILO estimate) 0
dtype: int64
In [11]: data1.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2775 entries, 63 to 12414
Data columns (total 15 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Country Name 2775 non-null object
1 Country Code 2775 non-null object
2 Region 2775 non-null object
3 IncomeGroup 2775 non-null object
4 Year 2775 non-null int64
5 Birth rate, crude (per 1,000 people) 2775 non-null float64

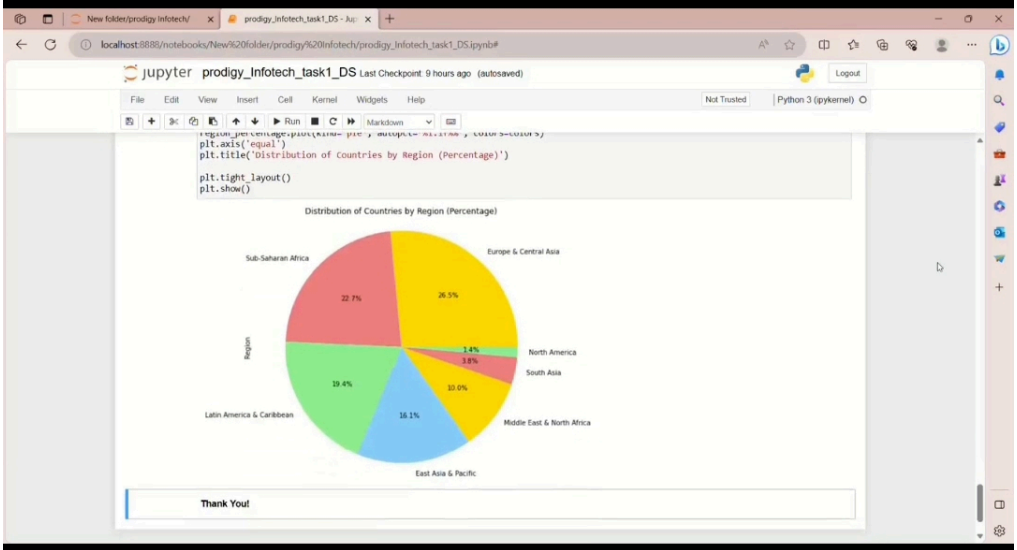

jupyter prodigy_Infotech_task1_DS Last Checkpoint: 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
variable = 'Life expectancy at birth (years)'
In [13]:
# Filter the data for the specific year
data_year = data[data['Year'] == year]
Create the histogram
In [14]:
plt.figure(figsize=(10, 6))
plt.hist(data_year[variable], bins=20, edgecolor='k', alpha=0.7)
plt.title(f'Distribution of {variable} in {year}')
plt.xlabel(variable)
plt.ylabel('Frequency')
plt.show()
Distribution of Life expectancy at birth (years) in 2012
Frequency


jupyter prodigy_Infotech_task1_DS Last Checkpoint: 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
Life expectancy at birth (years)
In [15]:
# Customize the plot further
plt.title(f'Distribution of {variable} in {year}')
plt.xlabel(variable)
plt.ylabel('Frequency')
plt.grid(True, linestyle='--', alpha=0.7)
# Add labels for mean and median
mean_value = data_year[variable].mean()
median_value = data_year[variable].median()
plt.axvline(mean_value, color='red', linestyle='dashed', linewidth=2, label=f'Mean: {mean_value:.2f}')
plt.axvline(median_value, color='green', linestyle='dashed', linewidth=2, label=f'Median: {median_value:.2f}')
plt.legend()
plt.show()
Distribution of Life expectancy at birth (years) in 2012
Mean: 71.25
Median: 73.05
Frequency


prodigy_Infotech_task1_DS
Last Checkpoint: 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
In [16]:
region_counts = data['Region'].value_counts()
# Create a bar chart
plt.figure(figsize=(10, 6))
region_counts.plot(kind='bar', color='skyblue')
plt.xlabel('Region')
plt.ylabel('Number of Countries')
plt.title('Distribution of Countries by Region')
plt.xticks(rotation=45)
plt.show()
Distribution of Countries by Region
Number of Countries
0
500
1000
1500
2000
2500
3000
South Asia
North America


prodigy_Infotech_task1_DS
Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
Europe & Central Asia
Sub-Saharan Africa
Latin America & Caribbean
East Asia & Pacific
Middle East & North Africa
South Asia
North America
Region
In [20]:
plt.figure(figsize=(15, 6))
plt.subplot(1, 2, 2)
region_percentage = (region_counts / region_counts.sum()) * 100
colors = ['gold', 'lightcoral', 'lightgreen', 'lightskyblue']
region_percentage.plot(kind='pie', autopct='%1.1f%%', colors=colors)
plt.axis('equal')
plt.title('Distribution of Countries by Region (Percentage)')
plt.tight_layout()
plt.show()
Distribution of Countries by Region (Percentage)
Sub-Saharan Africa
Europe & Central Asia
22.7%
26.5%
1.4%
3.8%
North America
South Asia
Region
19.4%


prodigy_Infotech_task1_DS - Ju
localhost:8888/notebooks/New%20folder/prodigy%20Infotech/prodigy_Infotech_task1_DS.ipynb#
jupyter prodigy_Infotech_task1_DS Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
plt.subplot(1, 2, 2)
region_percentage = (region_counts / region_counts.sum()) * 100
colors = ['gold', 'lightcoral', 'lightgreen', 'lightskyblue']
region_percentage.plot(kind='pie', autopct='%1.1f%%', colors=colors)
plt.axis('equal')
plt.title('Distribution of Countries by Region (Percentage)')
plt.tight_layout()
plt.show()
Distribution of Countries by Region (Percentage)
Sub-Saharan Africa
Europe & Central Asia
22.7%
26.5%
Region
1.4%
North America
3.8%
South Asia
19.4%
10.0%
Latin America & Caribbean
16.1%
Middle East & North Africa
East Asia & Pacific


jupyter prodigy_Infotech_task1_DS Last Checkpoint 9 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
plt.axis('equal')
plt.title('Distribution of Countries by Region (Percentage)')
plt.tight_layout()
plt.show()
Distribution of Countries by Region (Percentage)
Sub-Saharan Africa
Europe & Central Asia
22.7%
26.5%
Region
1.4%
North America
3.8%
South Asia
19.4%
10.0%
Latin America & Caribbean
16.1%
Middle East & North Africa
East Asia & Pacific
Thank You!