

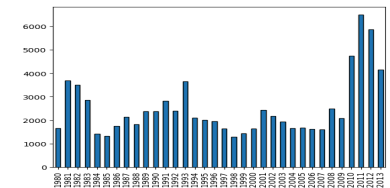
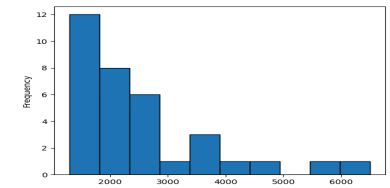
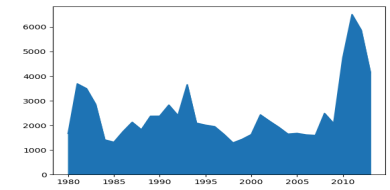
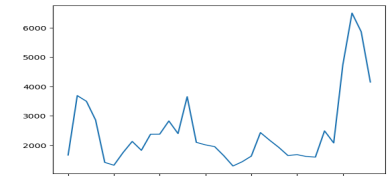


Data Visualization with Python

Cheat Sheet : Plotting with Matplotlib using Pandas

Plot Type	Description	Pandas Function	Example
Line Plot	Shows trends and changes over time	<code>DataFrame.plot.line()</code> <code>DataFrame.plot(kind = 'line')</code>	<code>df.plot(x='year', y='sales', kind='line')</code>
Area Plot	Displays data series as filled areas, showing the relationship between them	<code>DataFrame.plot.area()</code> <code>DataFrame.plot(kind = 'area')</code>	<code>df.plot(kind='area')</code>
Histogram	Displays bars representing the data count in each interval/bin	<code>Series.plot.hist()</code> <code>Series.plot(kind = 'hist', bins = n)</code>	<code>s.plot(kind='hist', bins=10)</code> <code>df['age'].plot(kind='hist', bins=10)</code>
Bar Chart	Displays data using rectangular bars	<code>DataFrame.plot.bar()</code> <code>DataFrame.plot(kind = 'bar')</code>	<code>df.plot(kind='bar')</code>

Visual



Plot Type Description

Pandas Function

Example

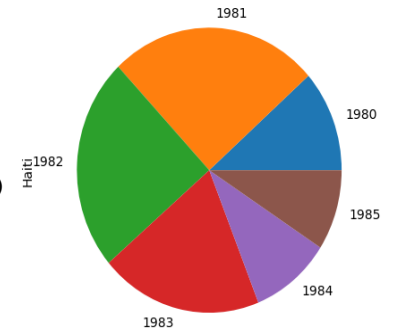
Visual

Pie Chart

Displays data as a circular plot divided into slices, representing proportions or percentages of a whole

```
Series.plot.pie()  
Series.plot(kind = 'pie')  
DataFrame.plot.pie(y, labels)  
DataFrame.plot(kind = 'pie')
```

```
s.plot(kind='pie', autopct='%1.1f%%')  
df.plot(x='Category', y='Percentage', kind='pie')
```

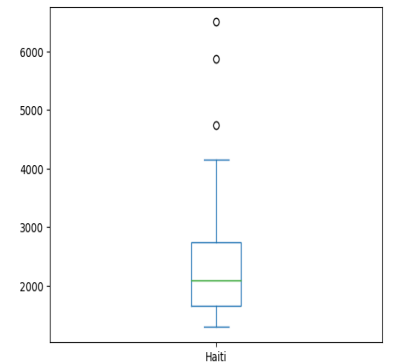


Box Plot

Displays the distribution of a dataset along with key statistical measures

```
DataFrame.plot.box()  
DataFrame.plot(kind = 'box')
```

```
df_can.plot(kind='box')
```

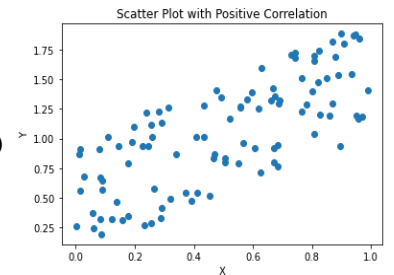


Scatter Plot

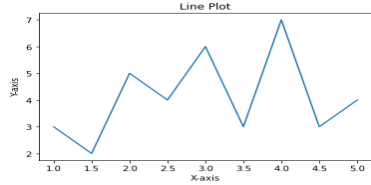
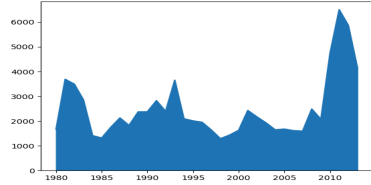
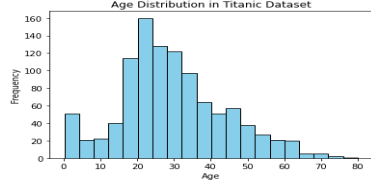
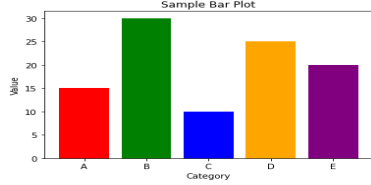
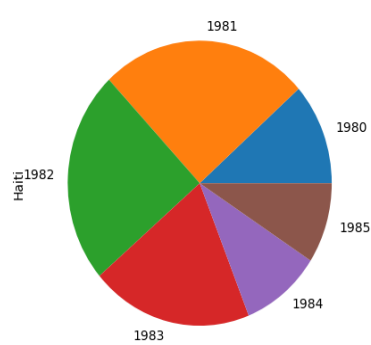
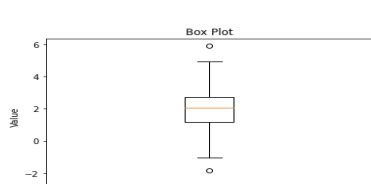
Uses Cartesian coordinates to display values for two variables

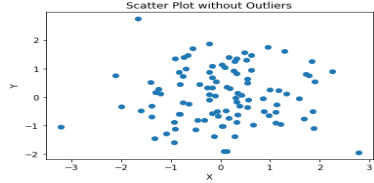
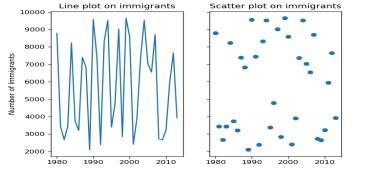
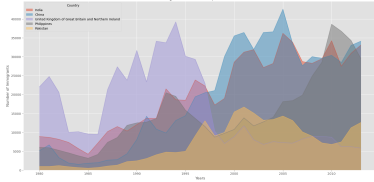
```
DataFrame.plot.scatter()  
DataFrame.plot(x, y, kind = 'scatter')
```

```
df.plot(x='Height', y='Weight', kind='scatter')
```



Cheat Sheet : Plotting directly with Matplotlib

Plot Type	Description	Matplotlib Function	Example	Visual
Line Plot	Shows trends and changes over time	<code>plt.plot()</code>	<code>plt.plot(x, y, color='red', linewidth=2)</code>	
Area Plot	Display data series as filled areas	<code>plt.fill_between()</code>	<code>plt.fill_between(x, y1, y2, color='blue', alpha=0.5)</code>	
Histogram	Displays bars representing the data count in each interval/bin	<code>plt.hist()</code>	<code>plt.hist(data, bins=10, color='orange', edgecolor='black')</code>	
Bar Chart	Displays data using rectangular bars	<code>plt.bar()</code>	<code>plt.bar(x, height, color='green', width=0.5)</code>	
Pie Chart	Displays data as a circular plot divided into slices, representing proportions or percentages of a whole	<code>plt.pie()</code>	<code>plt.pie(sizes, labels=labels, colors=colors, explode=explode)</code>	
Box Plot	Displays the distribution of a dataset along with key statistical measures	<code>plt.boxplot()</code>	<code>plt.boxplot(data, notch=True)</code>	

Plot Type	Description	Matplotlib Function	Example	Visual
Scatter Plot	Uses Cartesian coordinates to display values for two variables	<code>plt.scatter()</code>	<pre>plt.scatter(x, y, color='purple', marker='o', s=50)</pre>	 <p>A scatter plot titled "Scatter Plot without Outliers" showing a distribution of data points in a 2D space. The x-axis ranges from -3 to 3, and the y-axis ranges from -2 to 2. The points are blue circles with a size of 50, scattered across the plot area.</p>
Subplotting	Creating multiple plots on one figure	<code>plt.subplots()</code>	<pre>fig, axes = plt.subplots(nrows=2, ncols=2)</pre>	 <p>Two subplots side-by-side. The left subplot is a line plot titled "Line plot on immigrants" showing the number of immigrants from 1980 to 2010. The y-axis ranges from 2000 to 10000. The right subplot is a scatter plot titled "Scatter plot on immigrants" showing the same data as a scatter plot with blue circles.</p>
Customization	Customizing plot: adding labels, title, legend, grid	Various customization	<pre>plt.title('Title') plt.xlabel('X Label') plt.ylabel('Y Label') plt.legend() plt.grid(True)</pre>	 <p>A stacked area chart titled "Immigration Trend of Top 5 Countries" showing the number of immigrants from 1980 to 2010. The y-axis ranges from 0 to 10000. The chart shows the cumulative immigration of the top 5 countries, with a legend indicating the countries: India, China, USA, UK, and Germany.</p>

Author(s)

Dr. Pooja

Changelog

Date	Version	Changed by	Change Description
2023-06-10	0.1	Dr. Pooja	Initial version created