

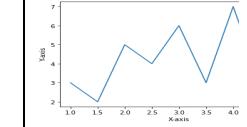
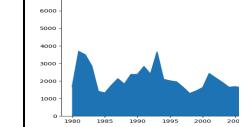
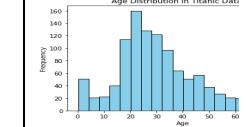
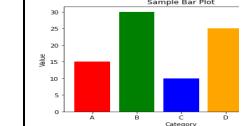
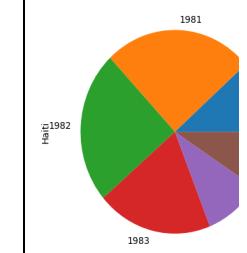
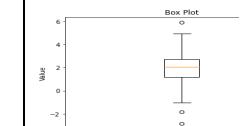
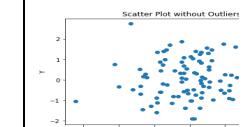
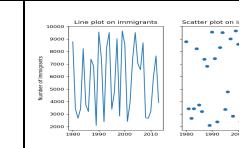
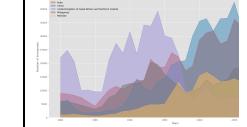


Data Visualization with Python

Cheat Sheet : Plotting with Matplotlib using Pandas

Plot Type	Description	Pandas Function	Example	Visual
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>	A line plot showing sales figures over time. The x-axis represents years from 1980 to 2000, and the y-axis represents sales from 0 to 6000. The plot shows a general upward trend with some fluctuations.
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>	An area plot showing multiple data series as filled regions. The x-axis represents years from 1980 to 2000, and the y-axis represents values from 0 to 6000. The plot displays several overlapping areas in different colors.
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>	A histogram showing the frequency of age data. The x-axis represents age groups from 2000 to 4000, and the y-axis represents frequency from 0 to 12. The distribution is skewed to the left with a peak around 2000.
Bar Chart	Displays data using rectangular bars	<code>DataFrame.plot.bar()</code> <code>DataFrame.plot(kind = 'bar')</code>	<code>df.plot(kind='bar')</code>	A bar chart showing data using rectangular bars. The x-axis represents categories from 1980 to 1983, and the y-axis represents values from 0 to 6000. The bars are colored blue, green, red, and orange respectively.
Pie Chart	Displays data as a circular plot divided into slices, representing proportions or percentages of a whole	<code>Series.plot.pie()</code> <code>Series.plot(kind = 'pie')</code> <code>DataFrame.plot.pie(y, labels)</code> <code>DataFrame.plot(kind = 'pie')</code>	<code>s.plot(kind='pie', autopct='%1.1f%%')</code> <code>df.plot(x='Category', y='Percentage', kind='pie')</code>	A pie chart showing proportions or percentages of a whole. The slices are labeled 1981, 1982, 1983, and 1984, representing different categories.
Box Plot	Displays the distribution of a dataset along with key statistical measures	<code>DataFrame.plot.box()</code> <code>DataFrame.plot(kind = 'box')</code>	<code>df_can.plot(kind='box')</code>	A box plot showing the distribution of height data. The x-axis is labeled 'Height' and the y-axis ranges from 0 to 6000. The plot includes whiskers, a box, and individual data points (outliers).
Scatter Plot	Uses Cartesian coordinates to display values for two variables	<code>DataFrame.plot.scatter()</code> <code>DataFrame.plot(x, y, kind = 'scatter')</code>	<code>df.plot(x='Height', y='Weight', kind='scatter')</code>	A scatter plot showing a positive correlation between height and weight. The x-axis is labeled 'Height' and the y-axis is labeled 'Weight'. The data points form a clear positive linear trend.

Cheat Sheet : Plotting directly with Matplotlib

Plot Type	Description	Matplotlib Function	Example	Visual
Line Plot	Shows trends and changes over time	plt.plot()	plt.plot(x, y, color='red', linewidth=2)	
Area Plot	Display data series as filled areas	plt.fill_between()	plt.fill_between(x, y1, y2, color='blue', alpha=0.5)	
Histogram	Displays bars representing the data count in each interval/bin	plt.hist()	plt.hist(data, bins=10, color='orange', edgecolor='black')	
Bar Chart	Displays data using rectangular bars	plt.bar()	plt.bar(x, height, color='green', width=0.5)	
Pie Chart	Displays data as a circular plot divided into slices, representing proportions or percentages of a whole	plt.pie()	plt.pie(sizes, labels=labels, colors=colors, explode=explode)	
Box Plot	Displays the distribution of a dataset along with key statistical measures	plt.boxplot()	plt.boxplot(data, notch=True)	
Scatter Plot	Uses Cartesian coordinates to display values for two variables	plt.scatter()	plt.scatter(x, y, color='purple', marker='o', s=50)	
Subplotting	Creating multiple plots on one figure	plt.subplots()	fig, axes = plt.subplots(nrows=2, ncols=2)	
Customization	Customizing plot: adding labels, title, legend, grid	Various customization	plt.title('Title') plt.xlabel('X Label') plt.ylabel('Y Label') plt.legend() plt.grid(True)	

Author(s)

Dr. Pooja