



## Data Visualization with Python

# Cheat Sheet : Maps, Waffles, WordCloud and Seaborn

Function	Description	Syntax
Folium		
Map	Create a map object with specified center coordinates and zoom level.	<pre>folium.Map(location=[lat, lon], zoom_</pre>
Marker	Add a marker to the map with custom icon, popup, and tiles Tiles as Stamen Toner	<pre>folium.Marker(location=[lat , lon ], popup='Marker Popup'. tiles='Stamen Toner').add_to(map)</pre>
	Tiles as Stamen Terrain	<pre>folium.Marker(location=[lat , lon ], popup='Marker Popup'. tiles='Stamen Terrain').add_to(map)</pre>
Circle	Add a circle to the map with specified radius, color, and fill opacity.	<pre>folium.features.CircleMarker(location radius=n, color='red'. fill_opacity=n).add_to(map)</pre>
Chorpleth	Create a choropleth map based on a GeoJSON file and a specified data column.	<pre>folium.Choropleth(geo_data='path/to/g data=df, columns=['region', 'value co key on='feature.properties.id', fill_ fill_opacity=0.7, line_opacity=0.2, legend_name='Legend').add_to(map)</pre>
PyWaffle		
Waffle	Create a waffle chart based on values and categories.	<pre>plt.figure(FigureClass = Waffle,rows values)  waffle_chart = waffle.Waffle(values=[ rows=n, columns=n)</pre>
Legend	Add a legend to the waffle chart.	<pre>waffle_chart.legend(loc='upper left',</pre>
Title	Add a title to the waffle chart.	<pre>waffle_chart.set_title('Waffle Chart</pre>
Labels	Add labels to the waffle chart.	<pre>waffle_chart.set_labels(['Label 1', '</pre>
WordCloud		

Function	Description	Syntax
WordCloud	Create a word cloud object based on text data.	<pre>wordcloud = WordCloud().generate(text_data)</pre>
Generate	Generate the word cloud based on the text data.	<pre>wordcloud.generate(text_data)</pre>
Display	Display the word cloud using matplotlib or other plotting libraries.	<pre>plt.imshow(wordcloud, interpolation='nearest')</pre>
Options	Set various options for the word cloud, such as font, colors, mask, and stopwords.	<pre>wordcloud = WordCloud(font_path='path_to_font', background_color='white', colormap='Blues', mask=mask_image, stopwords=stopwords).generate(text_data)</pre>
Seaborn		
barplot	Create a bar plot to visualize the relationship between a categorical variable and a numeric variable.	<pre>sns.barplot(x='x_variable', y='y_variable', data=dataframe)</pre>
countplot	Create a count plot to display the frequency of each category in a categorical variable.	<pre>sns.countplot(x='category', data=dataframe)</pre>
regplot	Create a scatter plot with a linear regression line to visualize the relationship between two numeric variables.	<pre>sns.regplot(x='x_variable', y='y_variable', data=dataframe)</pre>

## Author(s)

Dr. Pooja

## Changelog

Date	Version	Changed by	Change Description
2023-06-18	0.1	Dr. Pooja	Initial version created