A Simple Task for Visualization/Graph - SeaBorn:

Dataset under discussion - Sample URL:

https://github.com/ShahzadSarwar10/AI-ML-Explorer/blob/main/USOpen-DataSet/Real_Estate_Sales_2001-2022_GL-Short.csv

It is REAL ESTATE – US data.

TASK:

1. Load above CVS file above, into DataFrame variable , with Pandas, following columns
   With “Serial Number” as Index column.
   Print DataFrame.
2. Call following method/properties of DataFrame, print output and analyze the output.
   .info()
   .dtypes
   .describe()
   .shape
   .
3. Draw - Line Plot, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.lineplot.html
   Study and Analyze the output graph.

4. Draw - categorical plots, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.catplot.html
   Study and Analyze the output graph.

5. Draw - Plot univariate or bivariate distributions using kernel density estimation, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.kdeplot.html
   Study and Analyze the output graph.

6. Draw - a scatter plot, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.scatterplot.html
   Study and Analyze the output graph.

7. Draw bar plot, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.barplot.html
   Study and Analyze the output graph.

8. Draw Plot rectangular data as a color-encoded matrix, with X parameter – as “Town“ and y parameter as “Assessed Value”
   https://seaborn.pydata.org/generated/seaborn.heatmap.html

9. Draw - Line Plot, with X parameter – as “Property Type“ and y parameter as “Sale Amount”
   https://seaborn.pydata.org/generated/seaborn.lineplot.html
   Study and Analyze the output graph.

10. Draw - categorical plots, with X parameter – as “Property Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.catplot.html
    Study and Analyze the output graph.

11. Draw - Plot univariate or bivariate distributions using kernel density estimation, with X parameter – as “Property Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.kdeplot.html
    Study and Analyze the output graph.

12. Draw - a scatter plot, with X parameter – as “Property Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.scatterplot.html
    Study and Analyze the output graph.

13. Draw bar plot, with X parameter – as “Property Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.barplot.html
    Study and Analyze the output graph.

14. Draw Plot rectangular data as a color-encoded matrix, with X parameter – as “Property Type“ and y parameter as “Assessed Value”
    https://seaborn.pydata.org/generated/seaborn.heatmap.html

15. Draw - Line Plot, with X parameter – as “Residential Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.lineplot.html
    Study and Analyze the output graph.

16. Draw - categorical plots, with X parameter – as “Residential Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.catplot.html
    Study and Analyze the output graph.

17. Draw - Plot univariate or bivariate distributions using kernel density estimation, with X parameter – as “Residential Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.kdeplot.html
    Study and Analyze the output graph.

18. Draw - a scatter plot, with X parameter – as “Residential Type“ and y parameter as “Sale Amount”
    https://seaborn.pydata.org/generated/seaborn.scatterplot.html
    Study and Analyze the output graph.

19. Draw bar plot, with X parameter – as “Residential Type“ and y parameter as “Sale Amount”
https://seaborn.pydata.org/generated/seaborn.barplot.html
Study and Analyze the output graph.

20. Draw Plot rectangular data as a color-encoded matrix, with X parameter – as “Residential Type“ and y parameter as “Assessed Value”
https://seaborn.pydata.org/generated/seaborn.heatmap.html
Study and Analyze the output graph.

21. Draw - Line Plot, “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Assessed Value”
https://seaborn.pydata.org/generated/seaborn.lineplot.html
Study and Analyze the output graph.

22. Draw - categorical plots, “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Sale Amount”
https://seaborn.pydata.org/generated/seaborn.catplot.html
Study and Analyze the output graph.

23. Draw - Plot univariate or bivariate distributions using kernel density estimation, “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Sale Amount”
https://seaborn.pydata.org/generated/seaborn.kdeplot.html
Study and Analyze the output graph.

24. Draw - a scatter plot, with “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Sale Amount”
https://seaborn.pydata.org/generated/seaborn.scatterplot.html
Study and Analyze the output graph.

25. Draw bar plot, with “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Sale Amount”
https://seaborn.pydata.org/generated/seaborn.barplot.html
Study and Analyze the output graph.

26. Draw Plot rectangular data as a color-encoded matrix, with “Date Recorded” extract “year”, to create as with X parameter – as “year“ and y parameter as “Assessed Value”
https://seaborn.pydata.org/generated/seaborn.heatmap.html
Study and Analyze the output graph.

SeaBorn – Theme

```
https://seaborn.pydata.org/generated/seaborn.set_theme.html
https://seaborn.pydata.org/tutorial/aesthetics.html
https://python-charts.com/seaborn/themes/
```

27. Create 5 - line plot, set following 5 theme one by one. [sns.set_theme( )]

```
darkgrid: Adds a gray background with white gridlines. It is the default
theme.
whitegrid: Adds gray gridlines on a white background.
dark: Similar to darkgrid but without the gridlines.
white: Similar to whitegrid but without the gridlines.
ticks: Adds ticks to the axes and uses a white background.
```

Study and Analyze the output 5 graph.

28. Create 5 - Bar plot, set following 5 theme one by one. [sns.set_style( )]

```
darkgrid: Adds a gray background with white gridlines. It is the default
theme.
whitegrid: Adds gray gridlines on a white background.
dark: Similar to darkgrid but without the gridlines.
white: Similar to whitegrid but without the gridlines.
ticks: Adds ticks to the axes and uses a white background.
```

Study and Analyze the output 5 graph.

29. Custom theme , for 5 graph.

Create custom theme, by using following theme property.

Study and Analyze the output 5 graph.

```
Customizing Themes
It is possible to customize the themes further by passing a dictionary of
parameters to the rc argument of seaborn.set_theme() or seaborn.set_style().
This allows for fine-grained control over the appearance of plots."""
```

```
axes.facecolor: Background color of the plotting area (e.g., 'white',
'#EAEAF2').
axes.edgecolor: Color of the axes lines (e.g., 'black', 'gray').
axes.linewidth: Width of the axes lines in points.
axes.grid: Whether to show the grid ('True' or 'False').
axes.grid.axis: Which axes to show the grid lines on ('x', 'y', or 'both').
axes.grid.which: Which grid lines to draw ('major', 'minor', or 'both').
axes.labelcolor: Color of the axis labels.
axes.labelsize: Size of the axis labels in points or as a relative string
(e.g., 'large', 'small').
axes.titlesize: Size of the plot title.
xtick.color: Color of the x-axis tick marks and labels.
ytick.color: Color of the y-axis tick marks and labels.
xtick.labelsize: Size of the x-axis tick labels.
ytick.labelsize: Size of the y-axis tick labels.
grid.color: Color of the grid lines.
grid.linewidth: Width of the grid lines.
font.family: Font family to use (e.g., 'sans-serif', 'serif', 'monospace').
font.size: Default font size for text elements.
lines.linewidth: Width of lines in plots.
lines.linestyle: Style of lines (e.g., '-', '--', '-.', ':').
patch.edgecolor: Color of patch edges (e.g., in histograms, bar plots).
patch.linewidth: Width of patch edges.
legend.frameon: Whether to display a frame around the legend ('True' or
'False').
legend.fontsize: Size of the legend text.
figure.figsize: Size of the figure (width, height) in inches.
figure.facecolor: Background color of the entire figure
```

Reference code: https://github.com/ShahzadSarwar10/AI-ML-Explorer/blob/main/Week4/Case4-18-Seaborn-Zameencom-property-data-by-Kaggle.py

Ask questions, if you have confusions. ASK me, Call me on whatsapp.

Let’s put best efforts.

Thanks