**jsahgdhgsaAdvanced Data Visualization using Seaborn**

**Duration**: 4 Hours  
**Prerequisites**: Basic Python, Pandas, Matplotlib fundamentals

**Outcomes:**

By the end of this course, learners will be able to:

* Understand Seaborn’s functionality and aesthetics for statistical plotting
* Create advanced and multi-dimensional visualizations with ease
* Customize Seaborn plots for clarity and presentation
* Perform exploratory data analysis using visual tools
* Integrate Seaborn with Pandas DataFrames and Matplotlib for flexible visuals

**Hour 1: Introduction to Seaborn and Basic Statistical Plots**

**Goal**: Understand Seaborn’s philosophy and create foundational plots

**Topics**:

* What is Seaborn? Advantages over Matplotlib for statistical visualization
* Importing Seaborn and setting styles (import seaborn as sns)
* Understanding the default dataset structure for Seaborn
* Plotting univariate distributions:
  + Histograms and KDE plots (sns.histplot(), sns.kdeplot())
  + Rug plots (sns.rugplot())
* Plotting bivariate relationships:
  + Scatter plots with regression lines (sns.regplot(), sns.lmplot())

**Exercise**:

* Visualize distribution of numerical features in a dataset (e.g., Iris or Titanic)
* Create scatter plots with regression lines between variables

**Hour 2: Categorical Data Visualization**

**Goal**: Explore techniques for visualizing categorical and grouped data

**Topics**:

* Bar plots (sns.barplot()) and count plots (sns.countplot())
* Box plots (sns.boxplot()) and violin plots (sns.violinplot())
* Swarm plots (sns.swarmplot()) and strip plots (sns.stripplot())
* Combining categorical plots with hue for subgroup analysis
* Customizing plot aesthetics: palettes, color codes, markers

**Exercise**:

* Compare distributions of categories using box and violin plots
* Visualize counts and groupings with bar and swarm plots

**Hour 3: Multi-Variable and Matrix Plots**

**Goal**: Handle complex datasets with multi-dimensional visualizations

**Topics**:

* Pair plots and scatter matrix (sns.pairplot())
* Heatmaps (sns.heatmap()) for correlation matrices and tabular data
* Cluster maps (sns.clustermap()) for hierarchical clustering visualization
* Facet grids (sns.FacetGrid()) for plotting subsets of data
* Joint plots (sns.jointplot()) combining scatter and histograms

**Exercise**:

* Create pair plots and heatmaps for a multi-variable dataset
* Use FacetGrid to visualize data subsets across categories

**Hour 4: Customization, Integration & Advanced Features**

**Goal**: Fine-tune plots and combine Seaborn with Matplotlib for enhanced visuals

**Topics**:

* Fine-tuning axes, labels, titles, and legends in Seaborn plots
* Adjusting plot size, aspect ratio, and figure aesthetics
* Integrating Seaborn with Matplotlib commands (plt.figure(), plt.subplots())
* Saving figures and exporting with custom DPI and formats
* Best practices for storytelling with data visualizations

**Mini Project**:

* Load a real-world dataset (e.g., Titanic, Flights, or Tips dataset)
* Conduct an exploratory data visualization analysis using multiple Seaborn plots
* Customize plots for clarity and presentation
* Export final visualizations for reporting