

TDI PYTHON FOR DATA ANALYTICS WEEK 4: Data Analysis with Pandas

Welcome to Week 4 of Python for Data Analytics!

This week, you will focus on working with data using the Pandas library. Pandas is essential for data manipulation and analysis, providing powerful and flexible tools for handling structured data.

Topics Covered:

Data analysis with Pandas

- Introduction to Pandas
- Understanding Pandas and its importance
- Series and DataFrames
- Creating and manipulating DataFrames

Dataset: Titanic Analysis

We will use the Titanic dataset you uploaded. This dataset contains information about passengers, such as age, class, fare, and survival status. You'll be working with this data to understand basic file operations and analysis using NumPy.

Learning Resources:

[Pandas Dataframe](#)

[Plotting With Pandas](#)

Questions:

1. Introduction to Pandas

- Explain the two primary data structures in Pandas: "Series" and "DataFrame".

- Discuss the advantages of Pandas compared to other libraries for handling structured data.

2. Understanding Pandas and Its Importance

- Demonstrate how to import Pandas and explain its basic functionality.
- Load the Titanic dataset into a Pandas DataFrame.

3. Series and DataFrames

- Create a Pandas “Series” by extracting a column from the Titanic dataset.
- Perform basic operations on the “Series”, such as checking for missing values and calculating summary statistics like the mean and median.
- Explore the first few rows of the DataFrame, determine its shape, and view summary statistics for the Titanic dataset.

4. Creating and Manipulating DataFrames

- Add new columns to the DataFrame using existing data (e.g., combining 'SibSp' and 'Parch' to create a 'FamilySize' column).
- Modify existing columns or create new categorical columns based on existing data (e.g., binning numerical data into categories).
- Identify missing values in the dataset and decide whether to fill them or drop them, depending on the situation.
- Filter the dataset based on multiple conditions (e.g., find passengers who survived and were in 1st class).
- Sort the data by various criteria (e.g., sort passengers by age in descending order).

5. Data Aggregation and Grouping

- Group data by specific categories (e.g., passenger class) and perform aggregation functions such as calculating the average fare and the survival rate for each group.

6. Data Visualization with Pandas

- Use Pandas' built-in plotting functions to create basic visualizations.
- Visualize the distribution of ages in the dataset or the survival rate across different passenger classes.

Submission Instructions:

- Submit your Python scripts or Jupyter notebook containing your solutions to the tasks outlined.
- Make sure your code is clean, well-documented, and follows best practices for readability and performance.