**ADITYA AMIN**

**ASSIGN : 25**

Q1. What is the distinction between a numpy array and a pandas data frame? Is there a way to convert between the two if there is?

Data Structure:

NumPy Array: A NumPy array is a homogeneous, multidimensional array of fixed size. It consists of elements of the same data type and is organized in a contiguous block of memory.

Pandas DataFrame: A Pandas DataFrame is a two-dimensional, labeled data structure. It can be thought of as a table with columns and rows, where each column can have a different data type. Internally, a DataFrame is composed of multiple NumPy arrays or Series objects, allowing flexibility in handling structured data.

Functionality:

NumPy Array: NumPy provides efficient numerical operations and mathematical functions on arrays. It is widely used for numerical computations, scientific computing, and data manipulation. However, it has limited support for data manipulation beyond basic array operations.

Pandas DataFrame: Pandas is built on top of NumPy and extends its functionality by providing high-level data manipulation and analysis tools. DataFrames offer features like labeled indexing, column-wise operations, data alignment, data cleaning, reshaping, grouping, merging, and powerful data querying capabilities.

Conversion between NumPy arrays and Pandas DataFrames is straightforward, and both libraries provide methods to facilitate this conversion:

NumPy Array to Pandas DataFrame: You can convert a NumPy array to a Pandas DataFrame using the pd.DataFrame() constructor. Simply pass the NumPy array as an argument, and Pandas will create a DataFrame with default column labels and index values.

Pandas DataFrame to NumPy Array: To convert a Pandas DataFrame to a NumPy array, you can use the values attribute of the DataFrame, which returns the underlying data as a NumPy array.

Q2. What can go wrong when an user enters in a stock-ticker symbol, and how do you handle it?

When a user enters a stock ticker symbol, several issues can arise, and it's important to handle them appropriately to ensure a smooth user experience and avoid potential errors. Some common problems and their handling methods include:

Invalid Ticker Symbol: The user may enter an invalid or non-existent ticker symbol that does not correspond to any stock. To handle this, you can implement error checking by verifying the ticker symbol against a known list of valid symbols or utilizing an API or data source that provides validation. If an invalid symbol is detected, you can display an error message to the user, prompt them to enter a valid symbol, or suggest similar symbols if a close match is found.

Q3. Identify some of the plotting techniques that are used to produce a stock-market chart.

Several plotting techniques are commonly used to produce stock market charts. Some of them include:

Line Chart: The line chart is a basic and widely used technique to plot stock market data. It represents the closing prices of stocks over a specific time period using connected line segments. Line charts provide a visual representation of price trends and patterns.

Candlestick Chart: Candlestick charts are popular in stock market analysis. They display the opening, closing, high, and low prices of a stock for a given time period. Each data point is represented by a "candlestick" consisting of a rectangular body (representing the opening and closing prices) and vertical lines (representing the high and low prices).

OHLC Chart: OHLC (Open, High, Low, Close) charts are similar to candlestick charts but use simpler graphical representations. Each data point is represented by a vertical line segment with a small horizontal line on each end, indicating the opening, closing, high, and low prices.

Q4. Why is it essential to print a legend on a stock market chart?

Printing a legend on a stock market chart is essential because it provides key information about the data being displayed. Here are a few reasons why it is important to include a legend:

Data Interpretation: A stock market chart may contain multiple lines, bars, or other graphical elements representing different stocks, indices, or indicators. The legend helps viewers understand which data series each element represents.

Data Comparison: Stock market charts often compare different stocks or indices. The legend allows viewers to identify and differentiate between the various securities or data series being compared.

Contextual Information: A legend provides contextual information about the units of measurement, scaling, or time period represented on the chart. For example, it may indicate that the vertical axis represents stock prices in dollars or that the horizontal axis represents time in months or years.

Chart Customization: Including a legend allows viewers to customize the chart display according to their preferences. They can toggle the visibility of specific data series using the legend.

Presentation and Communication: When sharing or presenting stock market charts, including a legend enhances the clarity and professionalism of the visual representation.

Q5. What is the best way to limit the length of a pandas data frame to less than a year?

To limit the length of a Pandas DataFrame to less than a year, you can filter the DataFrame based on the date or time range. Here's an example of how you can achieve this:

Assuming your DataFrame has a column named 'date' that represents the date of each data point, you can follow these steps:

Convert the 'date' column to a Pandas datetime format if it is not already in that format. This allows for convenient date-based operations.

Determine the desired time range by specifying the start and end dates for the one-year period. For example, if you want to limit the DataFrame to the past year from the latest date in the dataset.

Adjust the start and end dates according to your specific requirements.

Filter the DataFrame using the determined time range

Q6. What is the definition of a 180-day moving average?

A 180-day moving average is a technical analysis indicator that calculates the average value of a data series over a 180-day period. It is also known as a 6-month moving average because it considers the data of the past six months.

Q7. Did the chapter's final example use "indirect" importing? If so, how exactly do you do it?

Yes, the final example in the chapter used "indirect" importing to import modules. Indirect importing refers to importing modules indirectly through a variable or string. This technique allows for dynamic module loading based on runtime conditions or user input.