

Portfolio Performance Tracker with Benchmark Comparison

Project: Portfolio Performance Tracker with Benchmark Comparison

Duration: X weeks/months

Technologies: Python, Pandas, NumPy, Matplotlib

Overview:

This project tracks the performance of a portfolio comprising various assets and compares it with a benchmark index (such as NIFTY 50).

The portfolio is simulated using asset price fluctuations, and its performance is visualized with time-series graphs.

The project allows for real-time price updates and portfolio value adjustments, while calculating key financial metrics such as dividend income and asset value.

Key Features:

- Portfolio management tool that simulates asset price changes using a random walk model.
- Real-time portfolio value adjustment and optimization to match a target value.
- Performance comparison against a benchmark index like NIFTY 50.
- Data visualization of portfolio vs. benchmark index using Matplotlib.
- Uses Pandas for data manipulation and NumPy for numerical simulations.

Installation Instructions:

1. Clone this repository or download the project files.
2. Ensure that you have Python 3.x installed on your system.

3. Install the required libraries using pip:

```
`pip install pandas numpy matplotlib`
```

How to Use:

1. Add assets to the portfolio using the ``add_asset()`` method, specifying the name, type, units, price, and dividend yield.
2. Update asset prices using the ``update_price()`` method to reflect real-time data.
3. Adjust the portfolio to a target value using ``adjust_portfolio_value()``.
4. Simulate benchmark index data (e.g., NIFTY) and compare portfolio performance using ``compare_performance()``.
5. Generate the portfolio report to analyze asset allocation and dividend income.

Example usage:

```
...
```

```
portfolio = Portfolio()
```

```
portfolio.add_asset("IRCTC", "Stock", 50, 700, 0.5)
```

```
portfolio.add_asset("Zomato", "Stock", 100, 75, 0.0)
```

```
portfolio.add_asset("Reliance", "Stock", 20, 2300, 0.8)
```

```
portfolio.adjust_portfolio_value(21000)
```

```
portfolio.generate_report()
```

```
benchmark_data = pd.DataFrame({
```

```
    "Date": pd.date_range(start="2023-01-01", periods=10, freq="D"),
```

```
    "Index": portfolio.simulate_nifty(25200, 25100, 25000, 10)
```

```
})
```

portfolio.compare_performance(benchmark_data)

...

License:

This project is licensed under the MIT License - see the LICENSE file for details.

License

This project is licensed under the MIT License. You are free to use, modify, and distribute the code, provided that you include the appropriate copyright notice and disclaimers.

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files

(the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge,

publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is provided to do so,

subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF
CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.