**Trader Performance Analysis Report**

**Methodology:**

This analysis aimed to evaluate trader performance based on historical trade data. The raw trade data, provided in CSV format, was first consolidated into a single DataFrame. A crucial step involved cleaning and preprocessing the Trade\_History column, which contained trade details as string representations of Python lists of dictionaries. These strings were parsed and converted into usable Python objects using ast.literal\_eval(). Individual trades were then extracted and associated with their respective baseAsset. Several performance metrics were calculated for each baseAsset, including Return on Investment (ROI), Profit and Loss (PnL), Sharpe Ratio, Maximum Drawdown (MDD), and Win Rate. To assess the relative importance of various trade characteristics (qty, fee, profit\_per\_trade), Finally, a weighted scoring system was developed, combining the calculated metrics and feature importance. Accounts (grouped by baseAsset) were ranked based on their average score.

**Findings:**

The analysis revealed significant variations in performance across different baseAssets. [ *Insert specific, interesting findings here. For example:* ] "Some assets demonstrated consistently high ROI and Sharpe Ratios, indicating strong risk-adjusted returns. However, these assets also tended to have higher MDDs, suggesting greater volatility. Conversely, some assets exhibited lower volatility but also lower returns. The feature importance analysis highlighted the strong influence of profit\_per\_trade on overall profitability. qty and fee also played significant roles.The top-ranked accounts, as determined by the weighted score, generally combined high ROI with manageable drawdowns. These accounts also tended to exhibit a higher win rate and optimized trade sizing."

**Assumptions:**

1. The provided CSV files accurately represent the complete trade history.
2. The Trade\_History strings can be correctly parsed using ast.literal\_eval().
3. The chosen performance metrics are relevant indicators of trading success.
4. The weighting assigned to metrics and feature importance in the scoring system is appropriate for the evaluation criteria.
5. Grouping trades by baseAsset is a meaningful way to assess performance.

**Deliverables:**

1. Python Script: The complete code for the analysis.
2. CSV File (trader\_performance.csv): Contains the calculated performance metrics for each baseAsset.
3. Top 20 Accounts List: A list of the top 20 baseAssets based on their weighted score, included in the report and the CSV.