Python Code: Stress Test Simulation

This code simulates three types of stress scenarios for a portfolio:

1. **Market Crash**: A percentage drop in BTC and ETH prices.
2. **DeFi Protocol Exploit**: A significant drop in DeFi assets like AAVE.
3. **Regulatory Crackdown**: A drop in stablecoin and DeFi asset prices.

Explanation of the Code

1. **Data Setup**:
   - Define initial portfolio data, including asset names, types, initial allocations, and sample current prices.
2. **Stress Test Function**:
   - simulate_stress_test applies a simulated price drop based on asset type:
     - **BTC Drop**: Applies to BTC.
     - **ETH Drop**: Applies to ETH.
     - **DeFi Drop**: Applies to AAVE or other DeFi assets.
     - **Stablecoin Drop**: Applies to USDC or other stablecoins, simulating risks like a regulatory crackdown.
   - The function calculates the **Value After Stress** for each asset and determines the total portfolio value before and after the stress test.
3. **Results**:
   - The function outputs the total portfolio value before and after the stress test and the overall loss percentage.
4. **Visualization**:
   - A Plotly bar chart compares initial allocation and post-stress-test allocation, making it easy to see the impact of each scenario.

Adjusting Scenarios

You can adjust the btc_drop, eth_drop, defi_drop, and stablecoin_drop parameters in simulate_stress_test to simulate different severities for each type of market stress.