Project Description: Stock Market Monitoring System

**Project Overview:**

The Stock Market Monitoring System is designed to provide real-time updates to investors about significant changes in stock prices. Investors subscribe to receive real-time updates about specific stocks they are interested in, and when there is a significant change in the stock prices they will be notified so that they can change their investment Strategy at once. This is one of the practical scenarios for applying the Observer pattern. The system utilizes the Observer pattern to establish a one-to-many relationship between stocks (**subjects**) and investors (**observers**).

**Project Components:**

1. Stock (**Subject**):
   * Interface outlines the operations a Stock operation including:
   * **Add** and **remove** investors from list.
   * **Notifies** investors when there is a significant change in the stock price: For the sake of simplicity, notify method provide a new stock price (100, 120…) instead of the difference in value (+10, +20, -30...).
2. ConcreteStock (**Concrete Subject**):
   * Represents a stock with a unique symbol: e.g., AAPL for Apple
   * Maintains the current stock price: Market value.
   * ConcreteStock is the concrete subject implementing the “Subject" interface.
   * It maintains a list of observers (“Investor") and provides methods to manage this list
3. Investor (**Observer**):
   * An interface or abstract class representing the observer: Contains a method (e.g., **update**) that is called by the stock when there is a significant price change.
4. Strategy (enum):
   * A simple enumeration presents the common Strategy that investor may adapt: MonitorOnly, QuickTransactions, HoldPositions, HedgingStrategies.
5. SimpleInvestor (**Concrete Observers**):
   * A SimpleInvestor class that implements the observer interface.
   * SimpleInvestor has a name and keeps their current Strategy.
   * SimpleInvestor default behavior is “MonitorOnly”.
   * SimpleInvestor prefers adapting “HedgingStrategies” when stock price is lower than 50 but they are likely to adopt “QuickTransactions” when the price go higher than 200. (implement this behavior by simply **printing out** name, stock symbol, current price and current Strategy).
6. StockMarket (Client Code):
   * Write client code to demonstrate the usage of the Observer pattern.
   * Create instances of stocks, investors, and the stock market.
   * Register investors for specific stocks.
   * Simulate changes in stock prices and observe how investors are notified.

A diagram of a stock market

Description automatically generated

Practice Assignment

# **Implement stock market monitoring system using Observer pattern (80%)**

# **2. Extend the system to support an Advanced Investor (20%)**

* Advanced Investors default behavior is “HoldPositions”.
* Advanced Investors only change their strategy to “HedgingStrategies” if there are three consecutive price drops.