TASK 2: Stock Trading Platform

This Java program simulates a basic stock trading environment. It allows users to view market data, buy/sell stocks, and track portfolio performance. It uses Object-Oriented Programming (OOP) to manage users, stocks, and transactions. Data persistence can be optionally added using file I/O or a database.

```
import java.util.*;

class Stock {
    String symbol;
    double price;

    Stock(String symbol, double price) {
        this.symbol = symbol;
        this.price = price;
    }
}

class Portfolio {
    String userName;
    Map<String, Integer> holdings;

    Portfolio(String userName) {
        this.userName = userName;
        holdings = new HashMap<>();
    }

    void buy(String symbol, int quantity) {
        holdings.put(symbol, holdings.getOrDefault(symbol, 0) + quantity);
        System.out.println("Bought " + quantity + " of " + symbol);
    }

    void sell(String symbol, int quantity) {
        if (holdings.containsKey(symbol) && holdings.get(symbol) >= quantity) {
            holdings.put(symbol, holdings.get(symbol) - quantity);
            System.out.println("Sold " + quantity + " of " + symbol);
        } else {
            System.out.println("Not enough shares to sell.");
        }
    }
```

```
    void viewPortfolio() {
        System.out.println("Portfolio for " + userName + ": " + holdings);
    }
}

public class StockTradingPlatform {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Map<String, Stock> market = new HashMap<>();
        market.put("AAPL", new Stock("AAPL", 150.0));
        market.put("GOOG", new Stock("GOOG", 2800.0));
        market.put("TSLA", new Stock("TSLA", 700.0));

        System.out.print("Enter your name: ");
        String name = sc.nextLine();
        Portfolio portfolio = new Portfolio(name);

        while (true) {
            System.out.println("\n1. View Market\n2. Buy Stock\n3. Sell Stock\n4. View Portfolio\n5. Exit");
            System.out.print("Choose an option: ");
            int choice = sc.nextInt();
            sc.nextLine(); // consume newline

            switch (choice) {
                case 1:
                    System.out.println("Market Data:");
                    for (Stock s : market.values()) {
                        System.out.println(s.symbol + " - $" + s.price);
                    }
                    break;
                case 2:
                    System.out.print("Enter stock symbol to buy: ");
                    String buySymbol = sc.nextLine().toUpperCase();
                    System.out.print("Enter quantity: ");
                    int buyQty = sc.nextInt();
                    portfolio.buy(buySymbol, buyQty);
                    break;
                case 3:
                    System.out.print("Enter stock symbol to sell: ");
                    String sellSymbol = sc.nextLine().toUpperCase();
                    System.out.print("Enter quantity: ");
                    int sellQty = sc.nextInt();
```

```
                portfolio.sell(sellSymbol, sellQty);
                break;
            case 4:
                portfolio.viewPortfolio();
                break;
            case 5:
                System.out.println("Exiting...");
                sc.close();
                return;
            default:
                System.out.println("Invalid option.");
        }
    }
  }
}
```