```java
import java.util.*;


public class Main {
  static class Stock {
      String symbol;
      String name;
      double price;


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


  static class Portfolio {
      Map<String, Integer> holdings = new HashMap<>();
      double cash = 10000.0;


      void buy(String symbol, int quantity, double price) {
          double total = quantity * price;
          if (cash >= total) {
              holdings.put(symbol, holdings.getOrDefault(symbol, 0) + quantity);
              cash -= total;
              System.out.println("Bought " + quantity + " shares of " + symbol);
```

```
    } else {
        System.out.println("Not enough cash!");
    }
}


void sell(String symbol, int quantity, double price) {
    int owned = holdings.getOrDefault(symbol, 0);
    if (owned >= quantity) {
        holdings.put(symbol, owned - quantity);
        cash += quantity * price;
        System.out.println("Sold " + quantity + " shares of " + symbol);
    } else {
        System.out.println("Not enough shares to sell!");
    }
}


void showPortfolio(Map<String, Stock> market) {
    System.out.println("\n--- Portfolio ---");
    for (var entry : holdings.entrySet()) {
        String symbol = entry.getKey();
        int qty = entry.getValue();
        double currentPrice = market.get(symbol).price;
        System.out.printf("%s: %d shares (Current Value: $%.2f)\n", symbol, qty, qty * currentPrice);
    }
    System.out.printf("Cash: $%.2f\n", cash);
    System.out.println("-----------------\n");
}
```

```
}


static class Market {
    Map<String, Stock> stocks = new HashMap<>();


    void generateStocks() {
        stocks.put("AAPL", new Stock("AAPL", "Apple Inc.", 150));
        stocks.put("GOOG", new Stock("GOOG", "Alphabet Inc.", 2800));
        stocks.put("TSLA", new Stock("TSLA", "Tesla Inc.", 700));
        stocks.put("AMZN", new Stock("AMZN", "Amazon.com Inc.", 3300));
    }


    void showMarket() {
        System.out.println("\n--- Market Data ---");
        for (Stock stock : stocks.values()) {
            stock.price = stock.price * (0.95 + Math.random() * 0.1); // simulate price change
            System.out.printf("%s (%s): $%.2f\n", stock.name, stock.symbol, stock.price);
        }
        System.out.println("-------------------\n");
    }


    Stock getStock(String symbol) {
        return stocks.get(symbol);
    }
}
```

```java
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    Market market = new Market();
    Portfolio portfolio = new Portfolio();
    market.generateStocks();


    while (true) {
        System.out.println("1. View Market");
        System.out.println("2. Buy Stock");
        System.out.println("3. Sell Stock");
        System.out.println("4. View Portfolio");
        System.out.println("5. Exit");
        System.out.print("Choose an option: ");


        int choice = scanner.nextInt();
        scanner.nextLine(); // consume newline


        switch (choice) {
            case 1:
                market.showMarket();
                break;
            case 2:
                System.out.print("Enter stock symbol to buy: ");
                String buySymbol = scanner.nextLine().toUpperCase();
                Stock buyStock = market.getStock(buySymbol);
                if (buyStock != null) {
                    System.out.print("Enter quantity: ");
                    int qty = scanner.nextInt();
                    portfolio.buy(buySymbol, qty, buyStock.price);
                } else {
                    System.out.println("Stock not found!");
                }
                break;
            case 3:
                System.out.print("Enter stock symbol to sell: ");
                String sellSymbol = scanner.nextLine().toUpperCase();
                Stock sellStock = market.getStock(sellSymbol);
                if (sellStock != null) {
                    System.out.print("Enter quantity: ");
                    int qty = scanner.nextInt();
                    portfolio.sell(sellSymbol, qty, sellStock.price);
                } else {
                    System.out.println("Stock not found!");
                }
                break;
            case 4:
                portfolio.showPortfolio(market.stocks);
                break;
            case 5:
                System.out.println("Exiting. Goodbye!");
                return;
            default:
                System.out.println("Invalid option.");
        }
```

```
   }
 }
}
```