**Exercise 7: Implementing the Observer Pattern**

public class ObserverPatternExample {  
  
 interface Stock {  
 void register(Observer observer);  
 void deregister(Observer observer);  
 void notifyObservers(double price);  
 }  
  
 interface Observer {  
 void update(double price);  
 }  
  
 static class StockMarket implements Stock {  
 private java.util.List<Observer> observers = new java.util.ArrayList<>();  
  
 // ... rest of the code  
  
 private double stockPrice;  
  
 public void setPrice(double price) {  
 this.stockPrice = price;  
 notifyObservers(price);  
 }  
  
 public void register(Observer observer) {  
 observers.add(observer);  
 }  
  
 public void deregister(Observer observer) {  
 observers.remove(observer);  
 }  
  
 public void notifyObservers(double price) {  
 for (Observer o : observers) {  
 o.update(price);  
 }  
 }  
 }  
  
 static class MobileApp implements Observer {  
 private String name;  
  
 public MobileApp(String name) {  
 this.name = name;  
 }  
  
 public void update(double price) {  
 System.*out*.println("MobileApp [" + name + "] - Stock Price Updated: " + price);  
 }  
 }  
  
 static class WebApp implements Observer {  
 private String name;  
  
 public WebApp(String name) {  
 this.name = name;  
 }  
  
 public void update(double price) {  
 System.*out*.println("WebApp [" + name + "] - Stock Price Updated: " + price);  
 }  
 }  
  
 public static void main(String[] args) {  
 StockMarket stockMarket = new StockMarket();  
  
 Observer mobileUser1 = new MobileApp("Alice");  
 Observer mobileUser2 = new MobileApp("Bob");  
 Observer webUser = new WebApp("Dashboard");  
  
 stockMarket.register(mobileUser1);  
 stockMarket.register(mobileUser2);  
 stockMarket.register(webUser);  
  
 System.*out*.println("Stock price update to 100.50:");  
 stockMarket.setPrice(100.50);  
  
 System.*out*.println("\nDeregistering Bob...");  
 stockMarket.deregister(mobileUser2);  
  
 System.*out*.println("\nStock price update to 105.75:");  
 stockMarket.setPrice(105.75);  
 }  
}

OUTPUT:

A screenshot of a computer

AI-generated content may be incorrect.