import java.util.Scanner;  
class Product {  
 private String productName;  
 private int productQuantity;  
 private double productPrice;  
 public Product(String productName, int productQuantity, double productPrice) {  
 this.productName = productName;  
 this.productQuantity = productQuantity;  
 this.productPrice = productPrice;  
 }  
 public void displayProductInfo() {  
 System.out.println("Product Name: " + productName);  
 System.out.println("Quantity: " + productQuantity);  
 System.out.println("Price per unit: $" + productPrice);  
 }  
 public int getQuantity() {  
 return productQuantity;  
 }  
 public double getPrice() {  
 return productPrice;  
 }  
 public boolean reduceStock(int productQuantity) {  
 if (this.productQuantity >= productQuantity) {  
 this.productQuantity -= productQuantity;  
 System.out.println("Stock is down and remaining stock: " + this.productQuantity + " (10 - " + (productQuantity) + ")");  
 return true;  
 } else {  
 System.out.println("There is not enough stock to fulfill your order. \n" + "But if you want, there are 15 units in stock of Smart Watch-OHD/2, which is the higher model of the smart watch you want to buy. \n" + "There is a $12.45 difference between the prices. So, OHD/2 is more expensive.\n" + "But, if your budget is not enough, we have cheaper and higher quality smart watches. These are OHD/3, OHD/4 and more.");  
 return false;  
 }  
 }  
}  
class ShoppingBasket {  
 private double totalPrice;  
 public ShoppingBasket() {  
 this.totalPrice = 0;  
 }  
 public boolean addToBasket(Product product, int productQuantity) {  
 if (product.getQuantity() >= productQuantity) {  
 totalPrice += product.getPrice() \* productQuantity;  
 product.reduceStock(productQuantity);  
 return true;  
 } else {  
 product.reduceStock(productQuantity);  
 return false;  
 }  
 }  
 public double getTotalPrice() {  
 return totalPrice;  
 }  
 public void completePayment(String IBAN, String address, String fullName) {  
 System.out.println("Payment of $" + totalPrice + " completed.");  
 System.out.println("Recipient: " + fullName);  
 }  
}  
public class Main {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.in);  
  
 Product smartWatch = new Product("Smart Watch - OHD/1", 10, 87.55);  
  
 smartWatch.displayProductInfo();

ShoppingBasket basket = new ShoppingBasket();   
  
 System.out.print("Enter the quantity you want to buy: ");  
 int productQuantity = scanner.nextInt();  
  
 boolean addedToBasket = basket.addToBasket(smartWatch, productQuantity);   
  
 if (addedToBasket) {  
 System.out.println("Total amount: $" + basket.getTotalPrice());   
  
 String approval = "";  
 while (!approval.equalsIgnoreCase("yes") && !approval.equalsIgnoreCase("no")) {  
 System.out.print("Do you approve the final price? (yes/no): ");  
 approval = scanner.next();  
 if (!approval.equalsIgnoreCase("yes") && !approval.equalsIgnoreCase("no")) {  
 System.out.println("Invalid input. Please answer with 'yes' or 'no'.");  
 }  
 }  
  
 if (approval.equalsIgnoreCase("yes")) {  
 String IBAN;  
 while (true) { System.out.print("Enter your IBAN (16 digits): ");  
 IBAN = scanner.next();  
 if (IBAN.length() == 16) {  
 break;  
 } else {  
 System.out.println("IBAN must be 16 characters.");  
 }  
 }  
  
 System.out.print("Enter recipient's full name: ");  
 scanner.nextLine();  
 String fullName = scanner.nextLine();  
  
 String address;  
 while (true) { System.out.print("Enter your address (max 25 characters): ");  
 address = scanner.nextLine();  
 if (address.length() <= 25) {  
 break;  
 } else {  
 System.out.println("Address must be at most 25 characters.");  
 }  
 }  
  
 basket.completePayment(IBAN, address, fullName); } else {  
 System.out.println("If your budget is not enough, we have cheaper and higher quality smart watches. These are OHD/3, OHD/4 and more.");  
 }  
 }  
 scanner.close();  
 }  
}

**OUTPUT:**

**Product Name: Smart Watch - OHD/1**

**Quantity: 10**

**Price per unit: $87.55**

**Enter the quantity you want to buy: 6**

**Stock is down and remaining stock: 4 (10 - 6)**

**Total amount: $525.3**

**Do you approve the final price? (yes/no): yes**

**Enter your IBAN (16 digits): 1234567891234567**

**Enter recipient's full name: Ömer Halis Demir**

**Enter your address (max 25 characters): Abcd efgh efaf agag ...**

**Payment of $525.3 completed.**

**Recipient: Ömer Halis Demir**

**metin, ekran görüntüsü, yazı tipi, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu**