Exercise 1: Implementing the Singleton Pattern

1. **class** SingletonDemo {

**private** **static** SingletonDemo *instance*;

**private** SingletonDemo() {

System.***out***.println("Singleton instance created");

}

**public** **static** SingletonDemo getInstance() {

**if**(*instance* == **null**) {

*instance* = **new** SingletonDemo();

}

**return** *instance*;

}

}

**public** **class** Singleton {

**public** **static** **void** main(String[] args) {

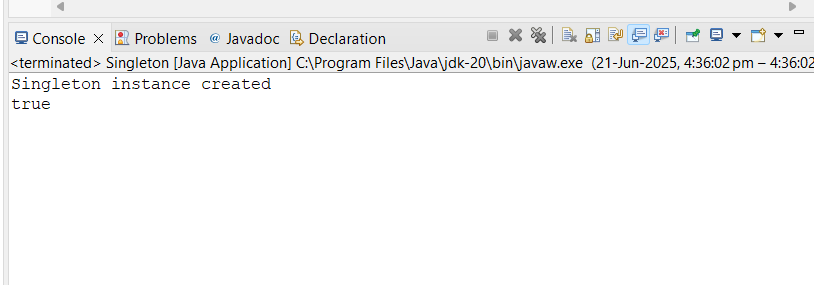
SingletonDemo obj1 = SingletonDemo.*getInstance*();

SingletonDemo obj2 = SingletonDemo.*getInstance*();

System.***out***.println(obj1 == obj2);

}

}



2) Exercise 2: Implementing the Factory Method Pattern

**interface** Room {

**void** book();

}

**class** DeluxeRoom **implements** Room {

**public** **void** book() {

System.***out***.println("Deluxe Room has been booked.");

}

}

**class** StandardRoom **implements** Room {

**public** **void** book() {

System.***out***.println("Standard Room has been booked.");

}

}

**abstract** **class** Hotel {

**public** **abstract** Room createRoom();

**public** **void** bookRoom() {

Room room = createRoom();

room.book();

}

}

**class** LuxuryHotel **extends** Hotel {

**public** Room createRoom() {

**return** **new** DeluxeRoom();

}

}

**class** BudgetHotel **extends** Hotel {

**public** Room createRoom() {

**return** **new** StandardRoom();

}

}

**public** **class** Factorymethod {

**public** **static** **void** main(String[] args) {

Hotel luxury = **new** LuxuryHotel();

luxury.bookRoom();

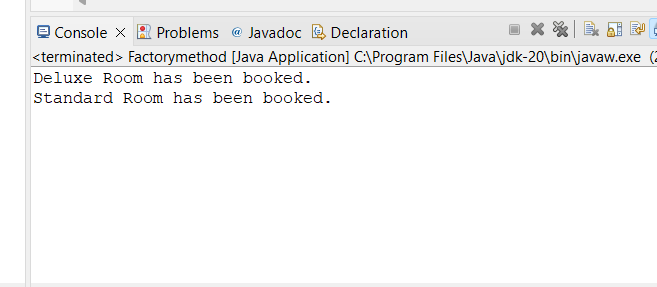
// Book room from Budget Hotel

Hotel budget = **new** BudgetHotel();

budget.bookRoom();

}

}



3) Exercise 2: E-commerce Platform Search Function

**import** java.util.Scanner;

**public** **class** Ecommerce {

**static** **class** Shoe {

String name;

**double** price;

Shoe(String name, **double** price) {

**this**.name = name;

**this**.price = price;

}

**void** display() {

System.***out***.println("Shoe: " + name + ", Price: " + price);

}

}

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

Shoe[] shoes = **new** Shoe[5];

shoes[0] = **new** Shoe("Nike Running Shoes", 4999);

shoes[1] = **new** Shoe("Adidas Sports Shoes", 4599);

shoes[2] = **new** Shoe("Puma Casual Shoes", 3999);

shoes[3] = **new** Shoe("Woodland Boots", 5999);

shoes[4] = **new** Shoe("Campus Walking Shoes", 2999);

System.***out***.print("Enter a word to search for shoes: ");

String keyword = sc.nextLine().toLowerCase();

**boolean** found = **false**;

System.***out***.println("\nSearch Results:");

**for** (**int** i = 0; i < shoes.length; i++) {

**if** (shoes[i].name.toLowerCase().contains(keyword)) {

shoes[i].display();

found = **true**;

}

}

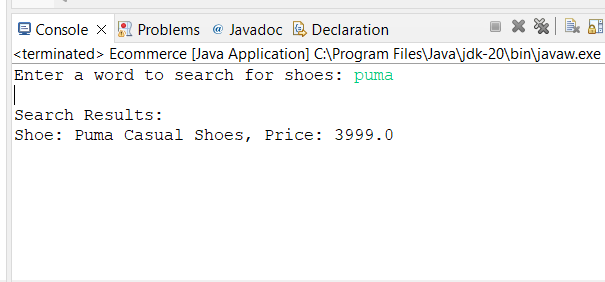
**if** (!found) {

System.***out***.println("No matching shoes found.");

}

}

}



4) Exercise 7: Financial Forecasting

**import** java.util.Scanner;

**public** **class** Financial {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.print("Enter your monthly income : ");

**double** income = sc.nextDouble();

System.***out***.print("Enter your monthly expenses : ");

**double** expenses = sc.nextDouble();

System.***out***.print("Enter number of months to forecast: ");

**int** months = sc.nextInt();

System.***out***.print("Enter expected monthly growth rate : ");

**double** growthRate = sc.nextDouble();

// Start forecasting

**double** savings = 0;

System.***out***.println("Financial Forecast:");

**for** (**int** i = 1; i <= months; i++) {

**double** monthlySavings = income - expenses;

savings += monthlySavings;

savings += (savings \* growthRate / 100);

System.***out***.printf("Month %d: Total Savings = %.2f\n", i, savings);

}

System.***out***.printf("\nAfter %d months, your projected savings: %.2f\n", months, savings);

sc.close();

}

}

