OOPS PROJECT FLAT RENTAL SYSTEM

import java.util.ArrayList;  
import java.util.List;  
import java.util.Scanner;  
  
class Flat {  
 private int Flat\_num;  
 private int Floor;  
 public int BHK;  
 private double Area;  
 private double basePricePermonths;  
 private boolean isAvailable;  
  
 public Flat(int Flat\_num, int floor, int BHK, double basePricePermonths, double Area) {  
 this.Flat\_num = Flat\_num;  
 this.Floor = floor;  
 this.BHK = BHK;  
 this.Area = Area;  
 this.basePricePermonths = basePricePermonths;  
 this.isAvailable = true; // by default...because initially all flats are available  
 }  
 public int getFlat\_num() {  
  
 return Flat\_num;  
 }  
  
 public int getFloor() {  
  
 return Floor;  
 }  
  
 public int getBHK() {  
 return BHK;  
 }  
 public double getArea(){  
 return Area;  
 }  
  
 public double getBasePricePermonths(){  
 return basePricePermonths;  
 }  
  
 public double calculatePrice(int rentalmonths) {  
 return basePricePermonths\* rentalmonths;  
 }  
  
 public boolean isAvailable() {  
 return isAvailable;  
 }  
  
 public void rent() {  
 isAvailable = false;  
 }  
  
 public void returnFlat() {  
 isAvailable = true;  
 }  
}  
  
class Customer {  
 private String customerID;  
 private String name;  
 // private int phone\_number;  
  
 public Customer(String customerID , String name ) {  
 this.customerID = customerID;  
 this.name = name;  
// this.phone\_number=phone\_number;  
  
 }  
 public String getCustomerID(){  
 return customerID;  
 }  
  
 public String getName() {  
 return name;  
 }  
// public int getPhone\_number() {  
// return phone\_number;  
// }  
  
}  
class Rental {  
 private Flat flat;  
 private Customer customer;  
 private int months;  
  
 public Rental(Flat flat, Customer customer, int months) {  
 this.flat= flat;  
 this.customer = customer;  
 this.months = months;  
 }  
  
 public Flat getFlat() {  
 return flat;  
 }  
  
 public Customer getCustomer() {  
  
 return customer;  
 }  
  
 public int getMonths() {  
 return months;  
 }  
}  
class FlatRentalSystem {  
 private List<Flat> flats; // only store flat related details  
 private List<Customer> customers; // only store customers related details  
 private List<Rental> rentals; // store which customers rented which flat  
  
 public FlatRentalSystem() { //constructor  
 flats = new ArrayList<>(); // memory allocation  
 customers = new ArrayList<>();  
 rentals = new ArrayList<>();  
 }  
  
 public void addflat(Flat flat) {  
  
 flats.add(flat);  
 }  
  
 public void addCustomer(Customer customer) {  
  
 customers.add(customer);  
 }  
  
 public void rentFlat(Flat flat, Customer customer, int months) { //method  
 if (flat.isAvailable()) { // if it is returned true ,then it calls flat.rent  
 flat.rent();  
 rentals.add(new Rental(flat, customer, months)); // show details  
  
 } else {  
 System.*out*.println("Oops you are late!! Flat is not available for rent.");  
 }  
 }  
  
 public void returnFlat(Flat flat) {  
 flat.returnFlat();  
 Rental rentalToRemove = null;  
 for (Rental rental : rentals) {  
 if (rental.getFlat() == flat) {  
 rentalToRemove = rental;  
 break;  
 }  
 }  
 if (rentalToRemove != null) {  
 rentals.remove(rentalToRemove);  
 System.*out*.println("Flat return successfully");  
 } else {  
 System.*out*.println("Flat was not rented.");  
 }  
 }  
  
 public void menu() {  
 Scanner scanner = new Scanner(System.*in*);  
  
 while (true) {  
 System.*out*.println("===== FlatRental System =====");  
 System.*out*.println("1. Rent a Flat");  
 System.*out*.println("2. Return a Flat");  
 System.*out*.println("3. Exit");  
 System.*out*.print("Enter your choice: ");  
  
 int choice = scanner.nextInt();  
 scanner.nextLine(); // Consume newline  
  
 if (choice == 1) {  
 System.*out*.println("\n== Rent a Flat ==\n");  
 System.*out*.print("Enter your name: ");  
 String customerName = scanner.nextLine();  
  
  
 System.*out*.println("\nAvailable Flats:");  
 //System.out.println("Flat\_number - floor - BHK - AREA");  
 for (Flat flat : flats) {  
 if (flat.isAvailable()) {  
 System.*out*.println("Flat Number :"+flat.getFlat\_num() + " - " +"Floor Number :"+flat.getFloor() + " - " + "BHK :"+flat.getBHK() + " - " +"Base Price per month of the flat :"+flat.getBasePricePermonths() +" - " + "Area of flat:"+flat.getArea());  
 }  
 }  
  
 System.*out*.println("\nEnter the FLAT number you want to rent: ");  
 int Flat\_num= scanner.nextInt();  
  
 System.*out*.print("Enter the number of months for rental: ");  
 int rentalMonths = scanner.nextInt();  
 scanner.nextLine(); // Consume newline  
  
 Customer newCustomer = new Customer("CUSTOMER" + (customers.size() + 1), customerName);  
 addCustomer(newCustomer);  
  
 Flat selectedFlat = null;  
 for (Flat flat : flats) {  
 int primitiveInt = flat.getFlat\_num();  
 Integer wrappedInt = Integer.*valueOf*(primitiveInt);  
 if (wrappedInt.equals(Flat\_num) && flat.isAvailable()) {  
 selectedFlat = flat;  
 break;  
 }  
 }  
  
 if (selectedFlat != null) {  
 double totalPrice = selectedFlat.calculatePrice(rentalMonths);  
 System.*out*.println("\n== Rental Information ==\n");  
 System.*out*.println("Customer ID: " + newCustomer.getCustomerID());  
 System.*out*.println("Customer Name: " + newCustomer.getName());  
 System.*out*.println("Flat Number: " + selectedFlat.getFlat\_num() );  
 System.*out*.println("BHK " + selectedFlat.getBHK());  
 System.*out*.println("Rental Months: " + rentalMonths);  
 System.*out*.printf("Total Price: $%.2f%n", totalPrice);  
  
 System.*out*.print("\nConfirm rental (Y/N): ");  
 String confirm = scanner.nextLine();  
  
 if (confirm.equalsIgnoreCase("Y")) {  
 rentFlat(selectedFlat, newCustomer, rentalMonths);  
 System.*out*.println("\nFlat rented successfully.");  
 } else {  
 System.*out*.println("\nRental canceled.");  
 }  
 } else {  
 System.*out*.println("\nInvalid Flat selection or car not available for rent.");  
 }  
 } else if (choice == 2) {  
 System.*out*.println("\n== Return a flat ==\n");  
 System.*out*.print("Enter the flat number you want to return: ");  
 int Flat\_num = scanner.nextInt();  
  
 Flat flatToReturn = null;  
  
 for (Flat flat : flats) {  
 int primitiveInt = flat.getFlat\_num();  
 Integer wrappedInt = Integer.*valueOf*(primitiveInt);  
 if (wrappedInt.equals(Flat\_num) && !flat.isAvailable()) {  
 flatToReturn = flat;  
 break;  
 }  
 }  
  
 if (flatToReturn != null) {  
 Customer customer = null;  
 for (Rental rental : rentals) {  
 if (rental.getFlat() == flatToReturn) {  
 customer = rental.getCustomer();  
 break;  
 }  
 }  
  
 if (customer != null) {  
 returnFlat(flatToReturn);  
 System.*out*.println("Flat returned successfully by " + customer.getName());  
 } else {  
 System.*out*.println("Flat was not rented or rental information is missing.");  
 }  
 } else {  
 System.*out*.println("Invalid Flat ID or Flat is not rented.");  
 }  
 } else if (choice == 3) {  
 break;  
 } else {  
 System.*out*.println("Invalid choice. Please enter a valid option.");  
 }  
 }  
  
 System.*out*.println("\nThank you for using the Flat Rental System!");  
 }  
  
}  
 public class Main{  
 public static void main(String[] args) {  
 FlatRentalSystem rentalSystem = new FlatRentalSystem();  
  
  
 Flat flat1 = new Flat(1,0,2,10000,500);  
 Flat flat2= new Flat( 2,0,3,20000,650);  
 Flat flat3 = new Flat(4,1 , 2 , 20000, 700);  
 Flat flat4 = new Flat(5,1 , 3 , 20000, 700);  
 Flat flat5 = new Flat(6,2 , 4 , 40000, 900);  
 Flat flat6 = new Flat(7,2 , 3 , 30000, 650);  
 Flat flat7 = new Flat(8,3 , 3 , 25000, 650);  
 Flat flat8 = new Flat(9,3 , 2 , 20000, 700);  
 Flat flat9 = new Flat(10,4 , 2 , 20000, 500);  
  
 rentalSystem.addflat(flat1);  
 rentalSystem.addflat(flat2);  
 rentalSystem.addflat(flat3);  
 rentalSystem.addflat(flat4);  
 rentalSystem.addflat(flat5);  
 rentalSystem.addflat(flat6);  
 rentalSystem.addflat(flat7);  
 rentalSystem.addflat(flat8);  
 rentalSystem.addflat(flat9);  
  
  
  
 rentalSystem.menu();  
 }  
 }