Kingdom of Saudi Arabia Ministry of Education
Imam Abdulrahman bin Faisal University
College of Science and Humanities
Computer Science Department

Salon management system

Wasan alshehri

Lama Abdulaziz

Layan Alanazi

Computer Science

1443هـ

Table of content

Table of content …….
Introduction…….
The problem statement……
The target…….
UML……
Full code…….
Conclusion.....

Introduction:

Salon design and implementation of the latest application ideas that many people need, everyone makes you more connected to his phone, and who is looking for more than that, his mobile, clarification because most salons are required to book before coming to the salon because of the COVID-19 pandemic, so they prefer to use more than that. Website and find out some prices, events, here is some information about each other. Browse to see available services, offers, and discounts.

The problem statement:

Many customers are present in the salon, not knowing whether the service they want is available or not because there is no prior reservation. In addition, this causes crowding, which may reduce the salon's productivity in terms of (customers delay and waste their time, work pressure) and then lead to loss salon for customers.

The target:

-Targeting a Gender

Although salons are open to people of all genders, the marketing efforts may be best spent targeting the gender that drops the most money on salon services: women. That said, you can certainly have more than one target market.

-Targeting the age categories

Salons receive people of all age groups, but it is better to target marketing for age group, the most visitors to salons are between 16 to 24 years old, so it is always recommended to market to this age group more than other groups.

-Targeting the financial side

It is better to diversify and target all financial groups, low, medium, high cost to suit everyone, so we will make sure to diversify the salons in terms of the financial aspect to win the largest amount of customers.

UML:

Customer
-Username : String -Password : String -phoneNumber: String -address: String -customerName:String -userManager: userManager
+ Copy() +getUsername() +getPassword() +findUser() +updateInfo() +getCustomerByUserName() +toString()

CustomerStatus
LOGIN: enum *LOGOUT:enum*

BookingInfo
reservationNumber: Integer Customer: customer Services: List totalPrice: double DayAndDate : dayAndDate;
getTotalPrice() getPrice() toString()

UserManger
customers: List **status: List UserManger** ***instance***
getInstance() **getCustomerByUserName()** **stream().filter()** **findUser()** **validateUser()** **addUser()** **login()** **getUsername()** **getPassword()** **logout()** **updateInfo()**

App
+main(args:Strinh[]):void

UserInterface
in:BufferReader currentuser:currentuser cart:List
userMenu() creatBooking() userInterface()

DayAndDate
date:String day:String
getDay() getDate() setDay() setDate() toString ()

Paying
totalPrice:double payOnArrivl:Boolean nameOfPaymentmethod:String cardName:String cardNumber:int securityCode:int cardExpiryDay:int yearCardExpiry:int
Paying() registers() toString() updateInfo()

Service
name:String price:double serviceCatalog: ServiceCatalog
Service() getName() getPrice() registerService() removeService() updatePrice()

Interface
in: BufferedReader
+IntroMenu() **+registerSalon()** **+updateSalonInfo** **+removeSalon()** **+addService()** **+showAllService()** **+registerUser()** **+loginUser()**

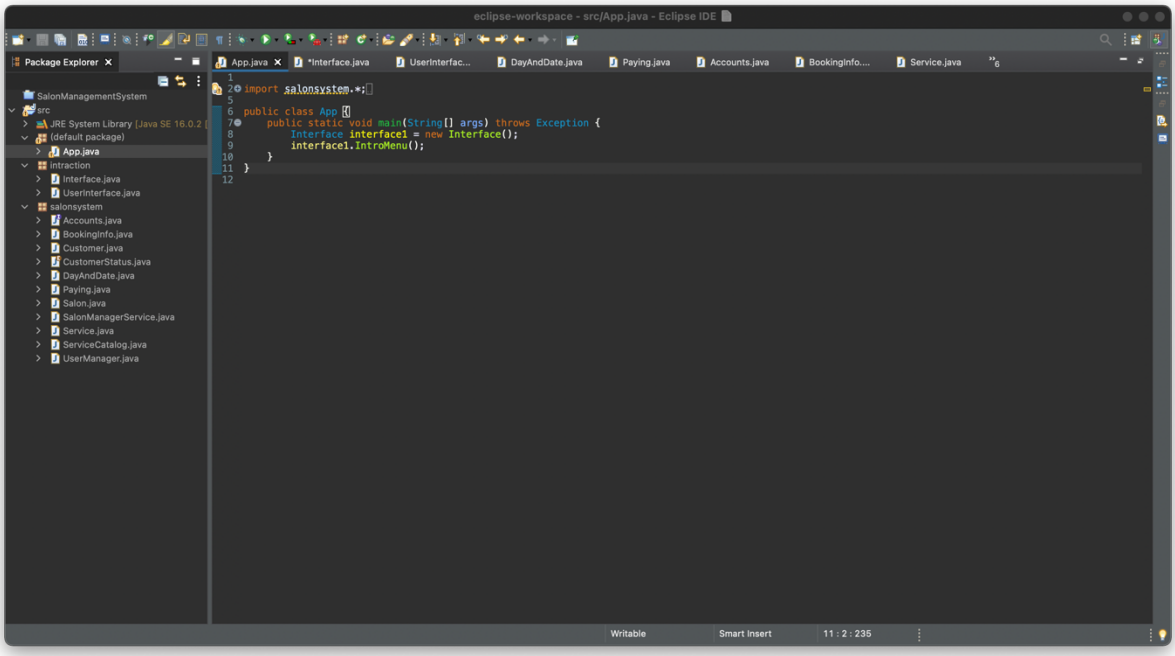
Accounts
getUsername() **getPassword()** **register()** **login()** **logout()** **updateInfo()**

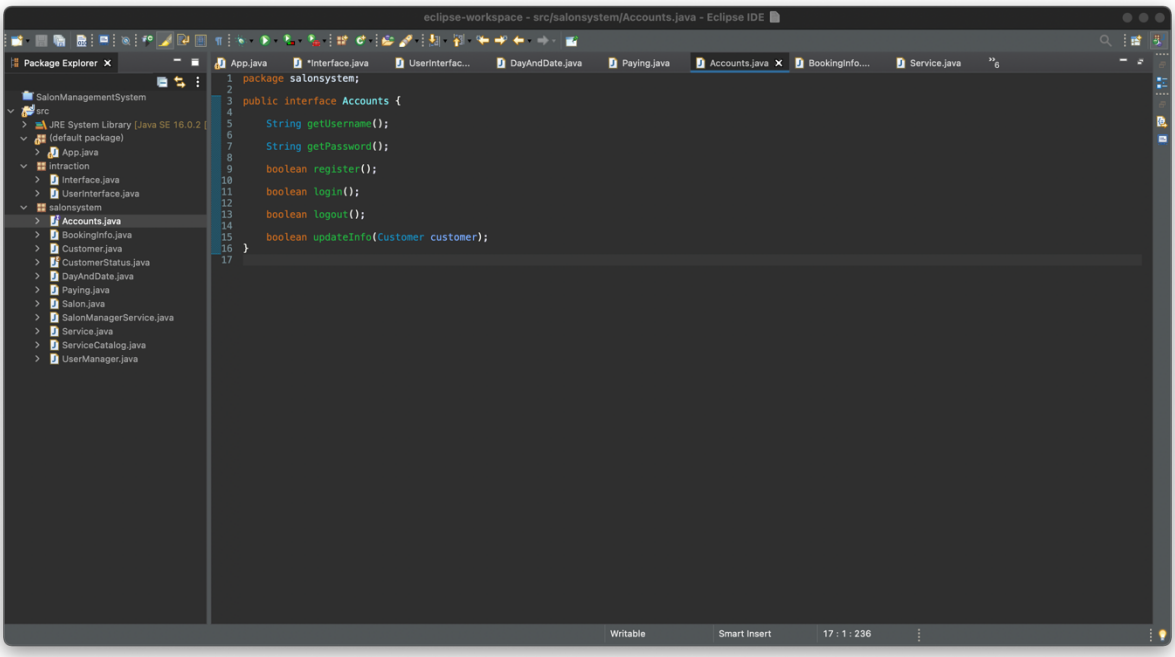
Salon
name:string **address:string** **salonManagerService:salonManager**
getNsme() **setName()** **getAddress()** **setAddress()** **registerSalon()** **removeSalon()** **findSalon()**

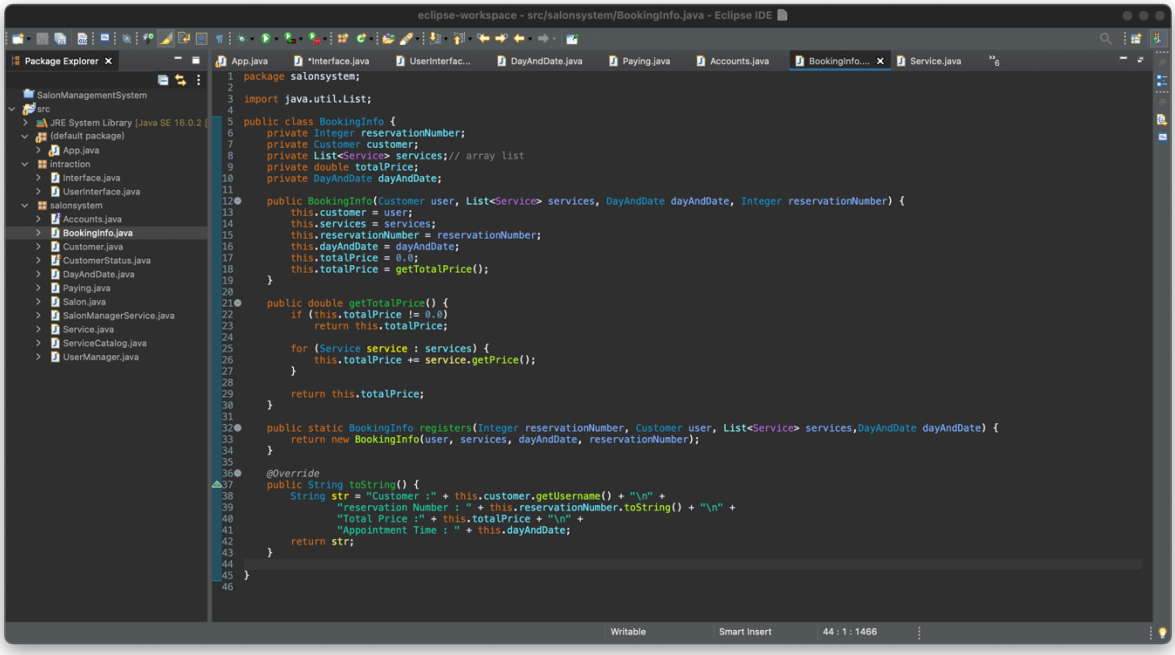
SalonMangerService
salon:Map<string,salon> instance:salonManagerService
getInstance() getID() registerSalon() removeSalon() findByName()

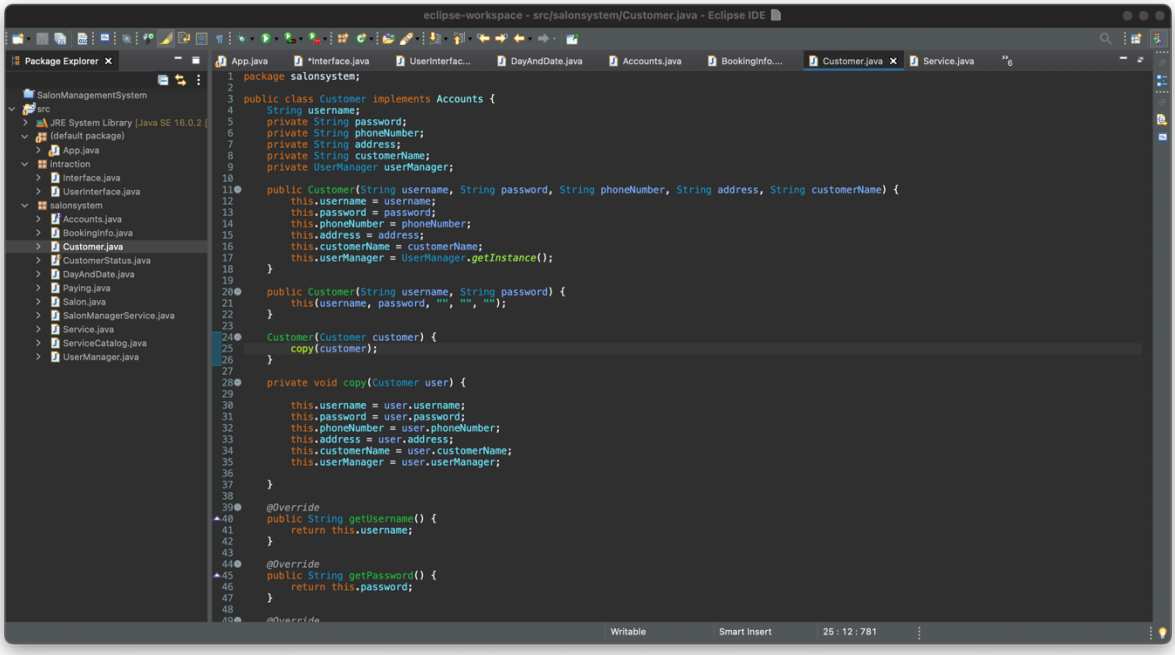
ServiceCatalog
service :Map<string,salon> instance:ServiceCatalog
getInstance() getAllService() find() register() remove() update()

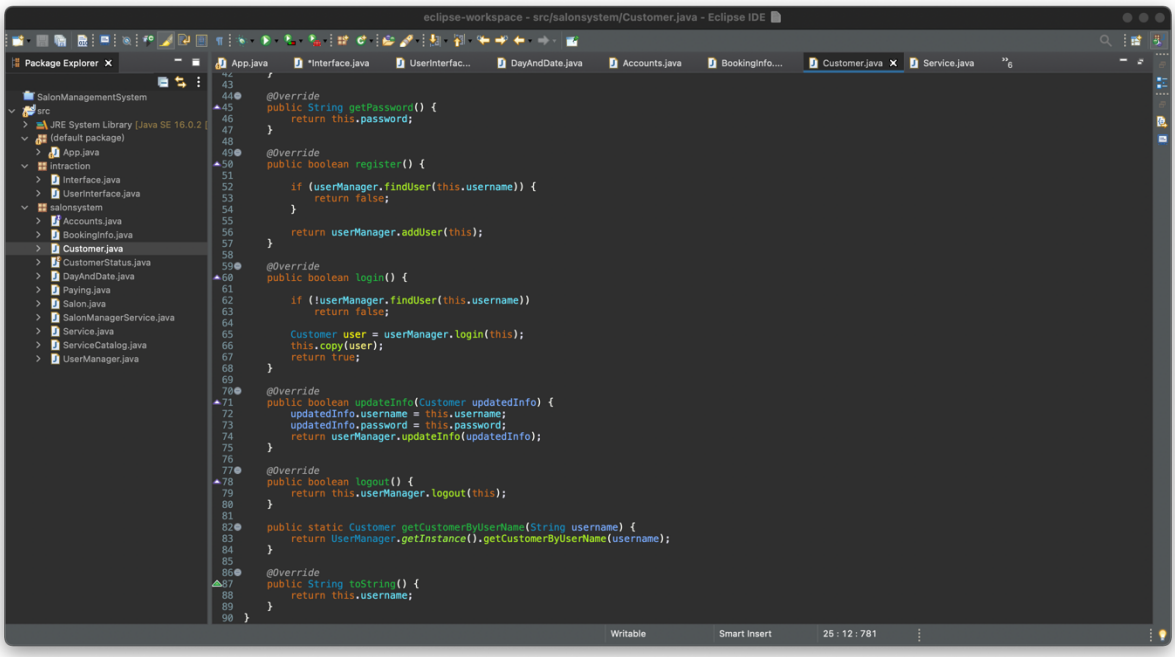
Full code :

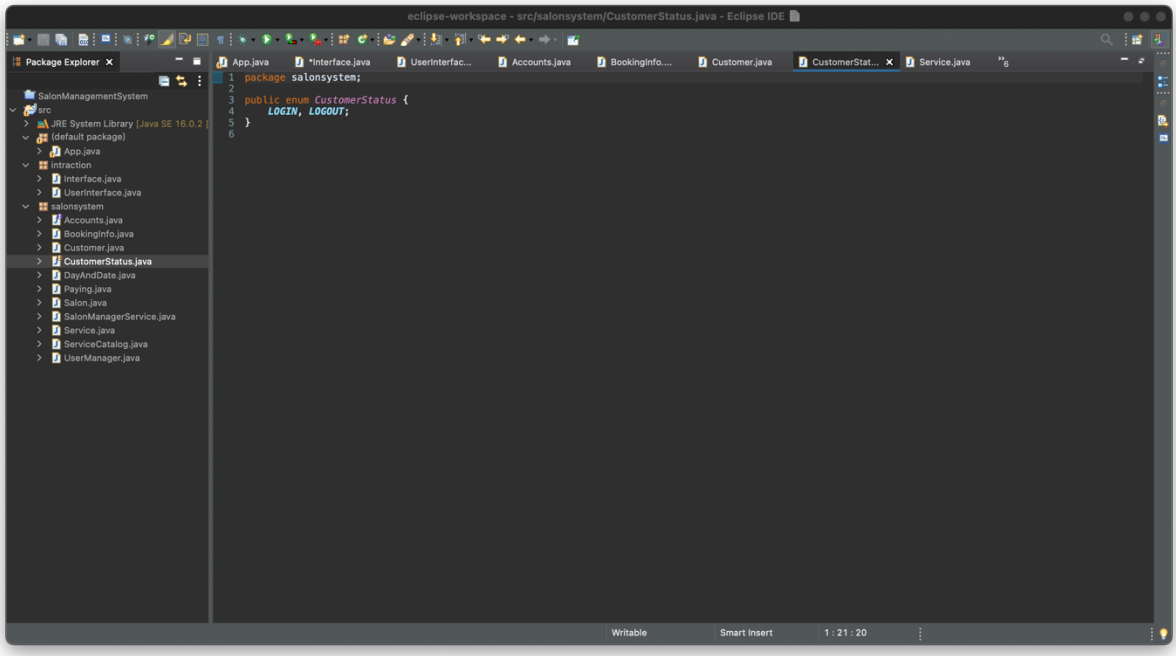

eclipse-workspace - src/App.java - Eclipse IDE
import salonsystem.*;
public class App {
public static void main(String[] args) throws Exception {
Interface interface1 = new Interface();
interface1.IntroMenu();
}
}

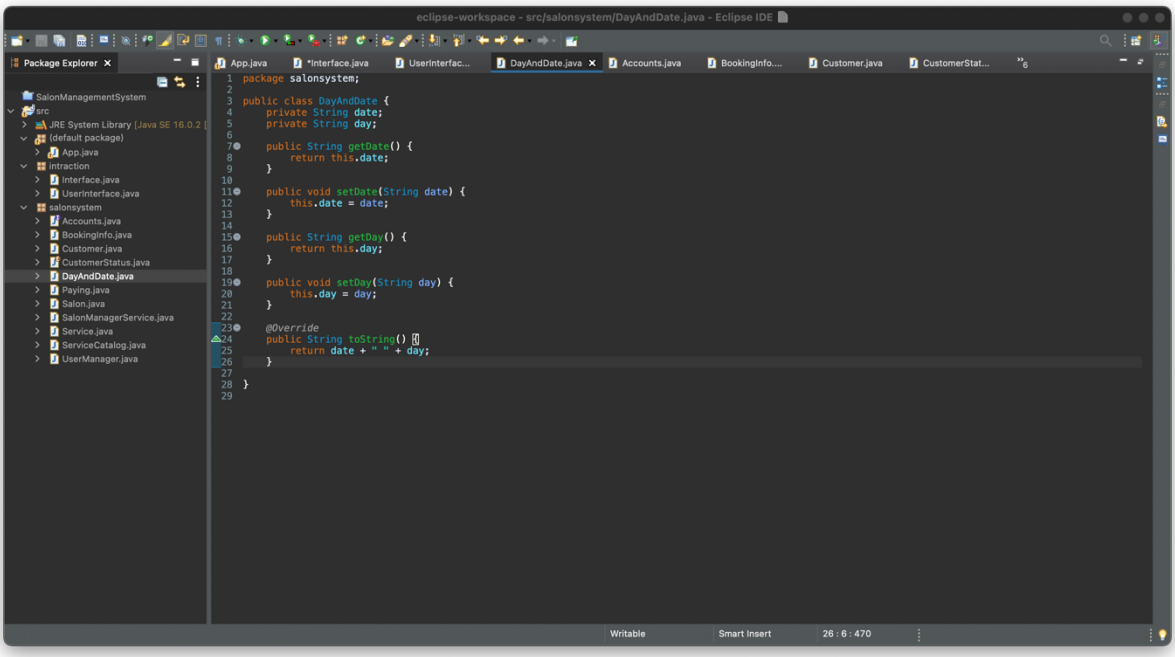

eclipse-workspace - src/salonsystem/Accounts.java - Eclipse IDE
package salonsystem;
public interface Accounts {
String getUsername();
String getPassword();
boolean register();
boolean login();
boolean logout();
boolean updateInfo(Customer customer);
}

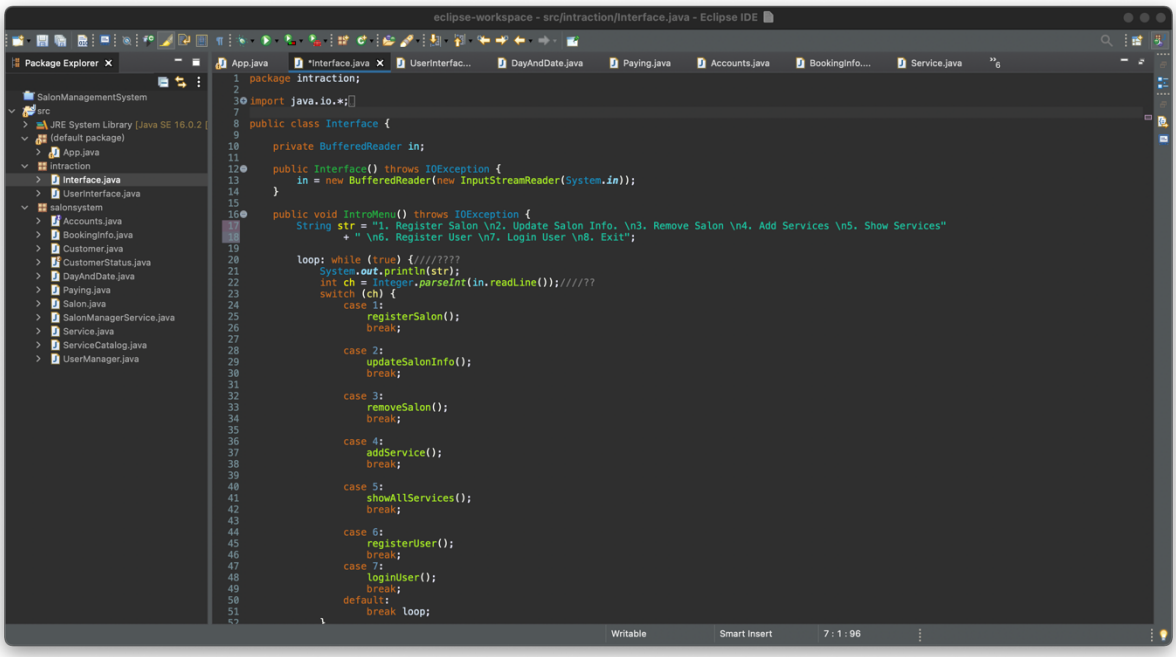

eclipse-workspace - src/salonsystem/BookingInfo.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package salonsystem;
import java.util.List;
public class BookingInfo {
private Integer reservationNumber;
private Customer customer;
private List<Service> services;// array list
private double totalPrice;
private DayAndDate dayAndDate;
public BookingInfo(Customer user, List<Service> services, DayAndDate dayAndDate, Integer reservationNumber) {
this.customer = user;
this.services = services;
this.reservationNumber = reservationNumber;
this.dayAndDate = dayAndDate;
this.totalPrice = 0.0;
this.totalPrice = getTotalPrice();
}
public double getTotalPrice() {
if (this.totalPrice != 0.0)
return this.totalPrice;
for (Service service : services) {
this.totalPrice += service.getPrice();
}
return this.totalPrice;
}
public static BookingInfo registers(Integer reservationNumber, Customer user, List<Service> services,DayAndDate dayAndDate) {
return new BookingInfo(user, services, dayAndDate, reservationNumber);
}
@Override
public String toString() {
String str = "Customer :" + this.customer.getUsername() + "\n" +
"reservation Number : " + this.reservationNumber.toString() + "\n" +
"Total Price :" + this.totalPrice + "\n" +
"Appointment Time : " + this.dayAndDate;
return str;
}
}
Writable
Smart Insert
44 : 1 : 1466


eclipse-workspace - src/salonsystem/Customer.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Accounts.java
BookingInfo....
Customer.java
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package salonsystem;
public class Customer implements Accounts {
String username;
private String password;
private String phoneNumber;
private String address;
private String customerName;
private UserManager userManager;
public Customer(String username, String password, String phoneNumber, String address, String customerName) {
this.username = username;
this.password = password;
this.phoneNumber = phoneNumber;
this.address = address;
this.customerName = customerName;
this.userManager = UserManager.getInstance();
}
public Customer(String username, String password) {
this(username, password, "", "", "");
}
Customer(Customer customer) {
copy(customer);
}
private void copy(Customer user) {
this.username = user.username;
this.password = user.password;
this.phoneNumber = user.phoneNumber;
this.address = user.address;
this.customerName = user.customerName;
this.userManager = user.userManager;
}
@Override
public String getUsername() {
return this.username;
}
@Override
public String getPassword() {
return this.password;
}
Writable
Smart Insert
25 : 12 : 781

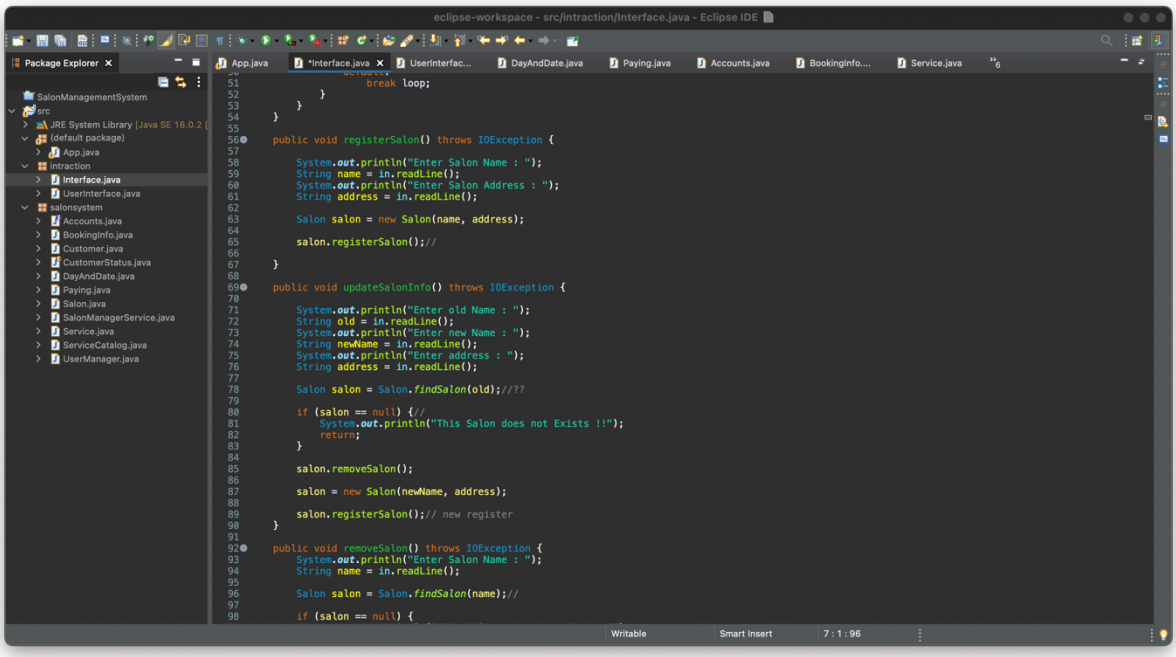

eclipse-workspace - src/salonsystem/Customer.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Accounts.java
BookingInfo....
Customer.java
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
@Override
public String getPassword() {
return this.password;
}
@Override
public boolean register() {
if (userManager.findUser(this.username)) {
return false;
}
return userManager.addUser(this);
}
@Override
public boolean login() {
if (!userManager.findUser(this.username))
return false;
Customer user = userManager.login(this);
this.copy(user);
return true;
}
@Override
public boolean updateInfo(Customer updatedInfo) {
updatedInfo.username = this.username;
updatedInfo.password = this.password;
return userManager.updateInfo(updatedInfo);
}
@Override
public boolean logout() {
return this.userManager.logout(this);
}
public static Customer getCustomerByUserName(String username) {
return UserManager.getInstance().getCustomerByUserName(username);
}
@Override
public String toString() {
return this.username;
}
}
Writable
Smart Insert
25 : 12 : 781

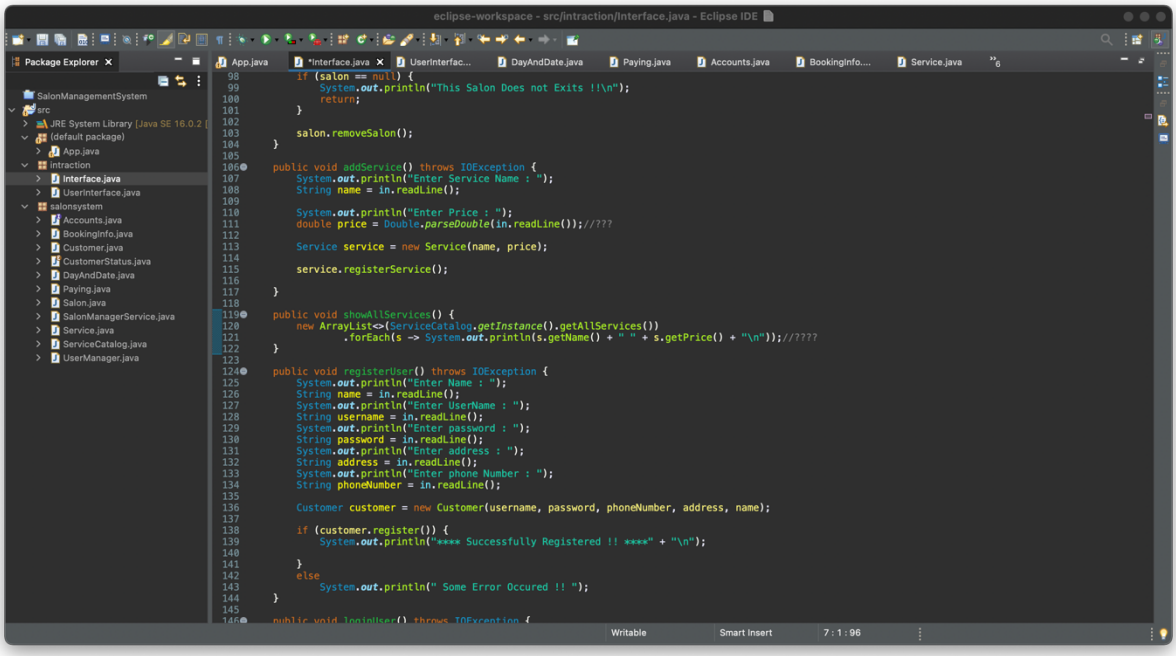

eclipse-workspace - src/salonsystem/CustomerStatus.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
Accounts.java
BookingInfo....
Customer.java
CustomerStat...
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package salonsystem;
public enum CustomerStatus {
LOGIN, LOGOUT;
}
Writable
Smart Insert
1 : 21 : 20

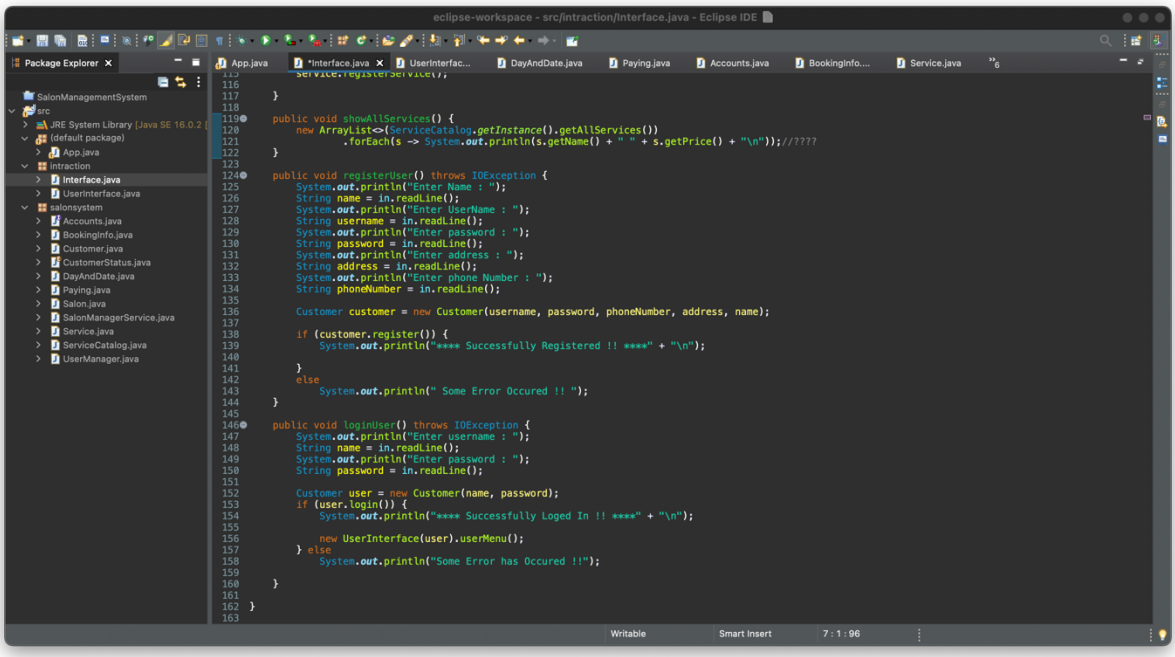

eclipse-workspace - src/salonsystem/DayAndDate.java - Eclipse IDE
Package Explorer
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
Intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Accounts.java
BookingInfo....
Customer.java
CustomerStat...
package salonsystem;
public class DayAndDate {
private String date;
private String day;
public String getDate() {
return this.date;
}
public void setDate(String date) {
this.date = date;
}
public String getDay() {
return this.day;
}
public void setDay(String day) {
this.day = day;
}
@Override
public String toString() {
return date + " " + day;
}
}
Writable
Smart Insert
26 : 6 : 470

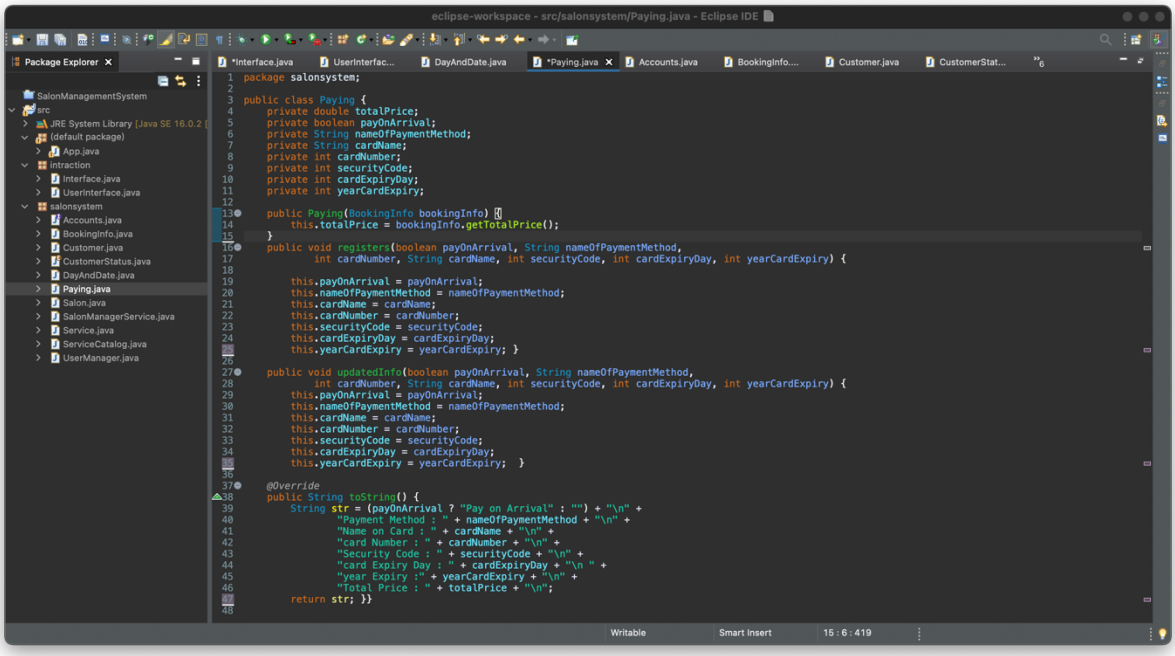


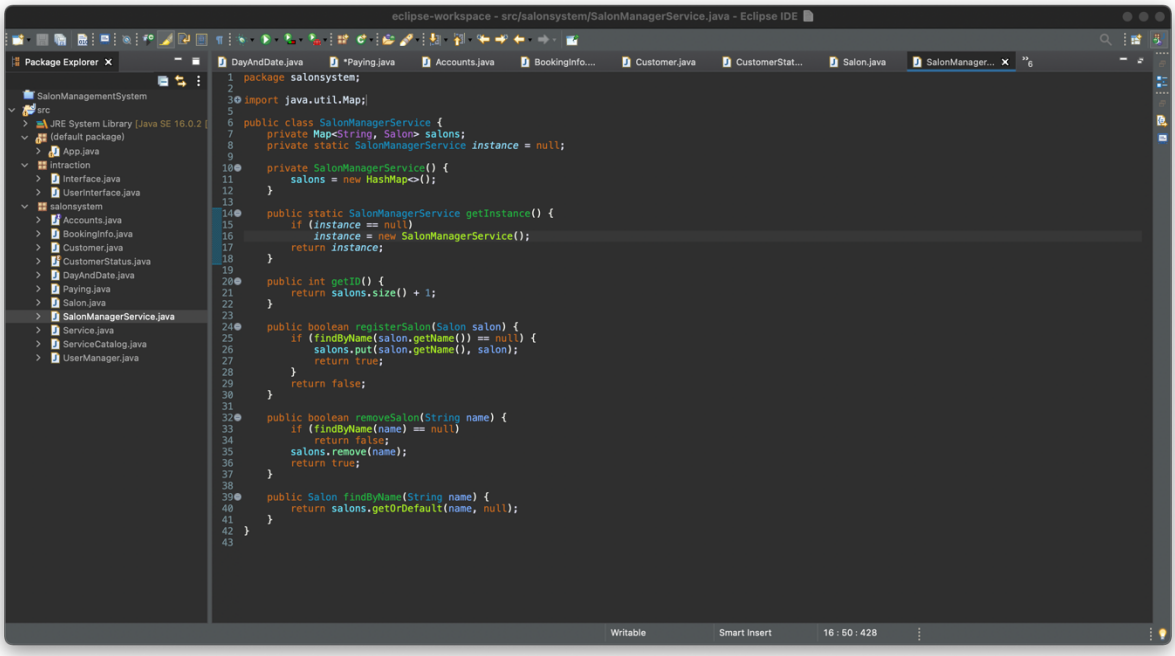

```
package intraction;

import java.io.*;

public class Interface {

    private BufferedReader in;

    public Interface() throws IOException {
        in = new BufferedReader(new InputStreamReader(System.in));
    }

    public void IntroMenu() throws IOException {
        String str = "1. Register Salon \n2. Update Salon Info. \n3. Remove Salon \n4. Add Services \n5. Show Services"
                + " \n6. Register User \n7. Login User \n8. Exit";

        loop: while (true) {////????
            System.out.println(str);
            int ch = Integer.parseInt(in.readLine());////??
            switch (ch) {
                case 1:
                    registerSalon();
                    break;

                case 2:
                    updateSalonInfo();
                    break;

                case 3:
                    removeSalon();
                    break;

                case 4:
                    addService();
                    break;

                case 5:
                    showAllServices();
                    break;

                case 6:
                    registerUser();
                    break;
                case 7:
                    loginUser();
                    break;
                default:
                    break loop;
            }
```

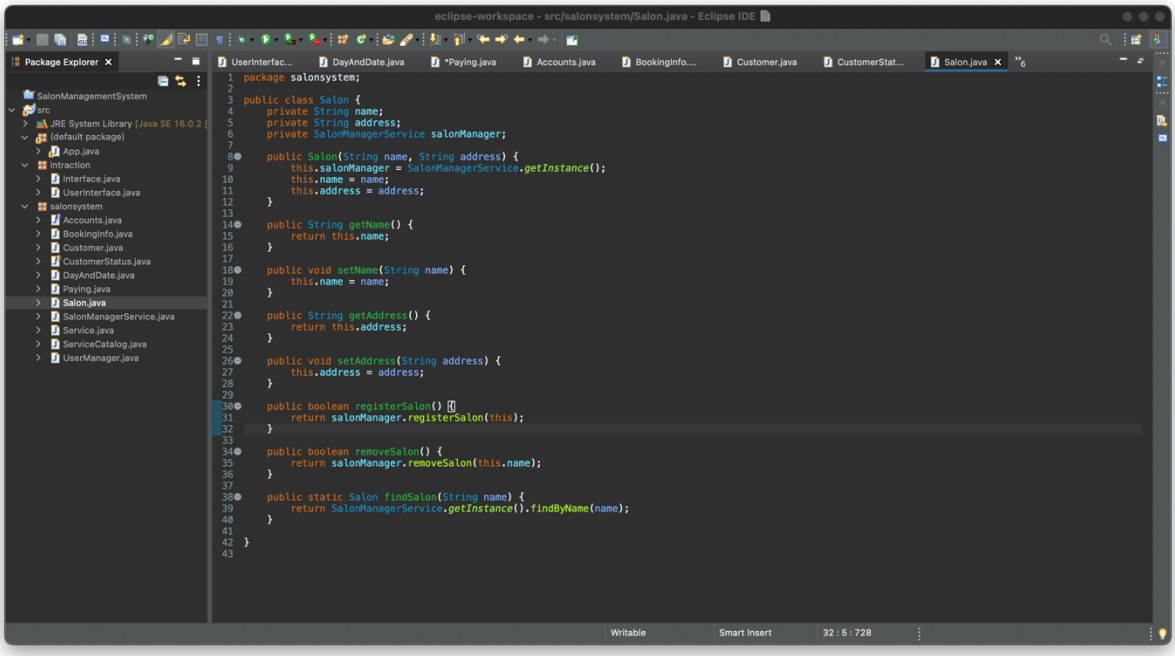

eclipse-workspace - src/intraction/Interface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
break loop;
}
}
public void registerSalon() throws IOException {
System.out.println("Enter Salon Name : ");
String name = in.readLine();
System.out.println("Enter Salon Address : ");
String address = in.readLine();
Salon salon = new Salon(name, address);
salon.registerSalon();//
}
public void updateSalonInfo() throws IOException {
System.out.println("Enter old Name : ");
String old = in.readLine();
System.out.println("Enter new Name : ");
String newName = in.readLine();
System.out.println("Enter address : ");
String address = in.readLine();
Salon salon = Salon.findSalon(old);//??
if (salon == null) {//
System.out.println("This Salon does not Exists !!");
return;
}
salon.removeSalon();
salon = new Salon(newName, address);
salon.registerSalon();// new register
}
public void removeSalon() throws IOException {
System.out.println("Enter Salon Name : ");
String name = in.readLine();
Salon salon = Salon.findSalon(name);//
if (salon == null) {
Writable
Smart Insert
7 : 1 : 96

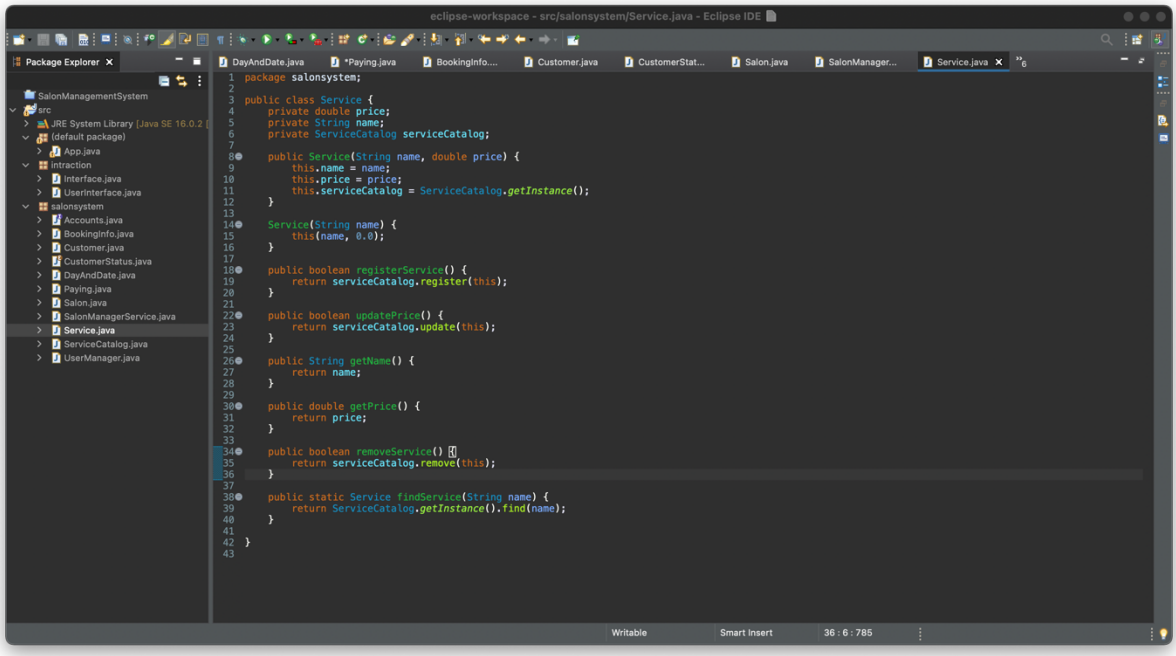

eclipse-workspace - src/intraction/Interface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
if (salon == null) {
System.out.println("This Salon Does not Exits !!\n");
return;
}
salon.removeSalon();
}
public void addService() throws IOException {
System.out.println("Enter Service Name : ");
String name = in.readLine();
System.out.println("Enter Price : ");
double price = Double.parseDouble(in.readLine());//???
Service service = new Service(name, price);
service.registerService();
}
public void showAllServices() {
new ArrayList<>(ServiceCatalog.getInstance().getAllServices())
.forEach(s -> System.out.println(s.getName() + " " + s.getPrice() + "\n"));//????
}
public void registerUser() throws IOException {
System.out.println("Enter Name : ");
String name = in.readLine();
System.out.println("Enter UserName : ");
String username = in.readLine();
System.out.println("Enter password : ");
String password = in.readLine();
System.out.println("Enter address : ");
String address = in.readLine();
System.out.println("Enter phone Number : ");
String phoneNumber = in.readLine();
Customer customer = new Customer(username, password, phoneNumber, address, name);
if (customer.register()) {
System.out.println("**** Successfully Registered !! ****" + "\n");
}
else
System.out.println(" Some Error Occured !! ");
}
Writable
Smart Insert
7 : 1 : 96

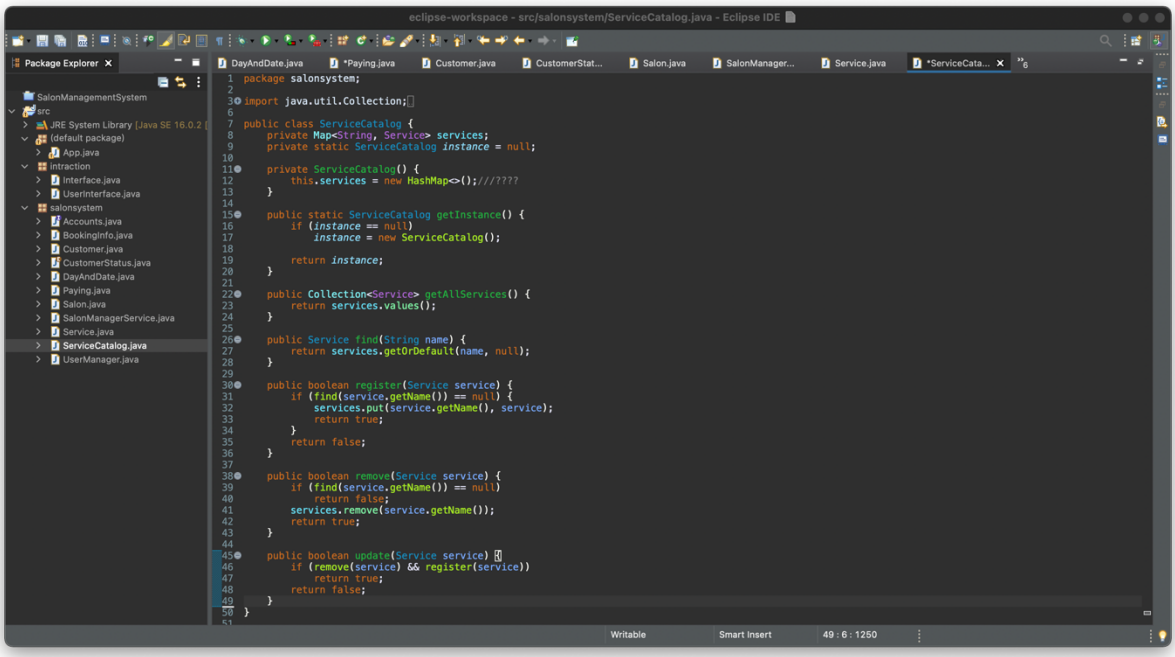

eclipse-workspace - src/intraction/Interface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2]
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
}
public void showAllServices() {
new ArrayList<>(ServiceCatalog.getInstance().getAllServices())
.forEach(s -> System.out.println(s.getName() + " " + s.getPrice() + "\n"));//????
}
public void registerUser() throws IOException {
System.out.println("Enter Name : ");
String name = in.readLine();
System.out.println("Enter UserName : ");
String username = in.readLine();
System.out.println("Enter password : ");
String password = in.readLine();
System.out.println("Enter address : ");
String address = in.readLine();
System.out.println("Enter phone Number : ");
String phoneNumber = in.readLine();
Customer customer = new Customer(username, password, phoneNumber, address, name);
if (customer.register()) {
System.out.println("**** Successfully Registered !! ****" + "\n");
}
else
System.out.println(" Some Error Occured !! ");
}
public void loginUser() throws IOException {
System.out.println("Enter username : ");
String name = in.readLine();
System.out.println("Enter password : ");
String password = in.readLine();
Customer user = new Customer(name, password);
if (user.login()) {
System.out.println("**** Successfully Loged In !! ****" + "\n");
new UserInterface(user).userMenu();
} else
System.out.println("Some Error has Occured !!");
}
}
Writable
Smart Insert
7 : 1 : 96

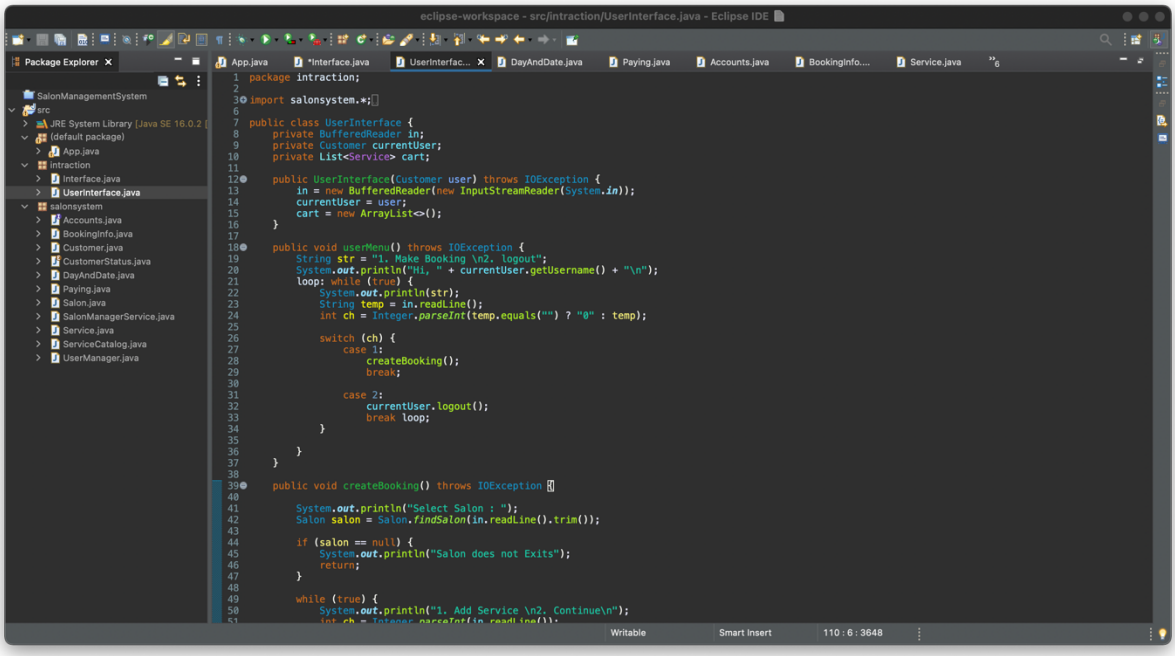

eclipse-workspace - src/salonsystem/Paying.java - Eclipse IDE
Package Explorer
*Interface.java
UserInterfac...
DayAndDate.java
*Paying.java
Accounts.java
BookingInfo....
Customer.java
CustomerStat...
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2]
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package salonsystem;
public class Paying {
private double totalPrice;
private boolean payOnArrival;
private String nameOfPaymentMethod;
private String cardName;
private int cardNumber;
private int securityCode;
private int cardExpiryDay;
private int yearCardExpiry;
public Paying(BookingInfo bookingInfo) {
this.totalPrice = bookingInfo.getTotalPrice();
}
public void registers(boolean payOnArrival, String nameOfPaymentMethod,
int cardNumber, String cardName, int securityCode, int cardExpiryDay, int yearCardExpiry) {
this.payOnArrival = payOnArrival;
this.nameOfPaymentMethod = nameOfPaymentMethod;
this.cardName = cardName;
this.cardNumber = cardNumber;
this.securityCode = securityCode;
this.cardExpiryDay = cardExpiryDay;
this.yearCardExpiry = yearCardExpiry; }
public void updatedInfo(boolean payOnArrival, String nameOfPaymentMethod,
int cardNumber, String cardName, int securityCode, int cardExpiryDay, int yearCardExpiry) {
this.payOnArrival = payOnArrival;
this.nameOfPaymentMethod = nameOfPaymentMethod;
this.cardName = cardName;
this.cardNumber = cardNumber;
this.securityCode = securityCode;
this.cardExpiryDay = cardExpiryDay;
this.yearCardExpiry = yearCardExpiry; }
@Override
public String toString() {
String str = (payOnArrival ? "Pay on Arrival" : "") + "\n" +
"Payment Method : " + nameOfPaymentMethod + "\n" +
"Name on Card : " + cardName + "\n" +
"Card Number : " + cardNumber + "\n" +
"Security Code : " + securityCode + "\n" +
"card Expiry Day : " + cardExpiryDay + "\n " +
"year Expiry :" + yearCardExpiry + "\n" +
"Total Price : " + totalPrice + "\n";
return str; }}
Writable
Smart Insert
15 : 6 : 419

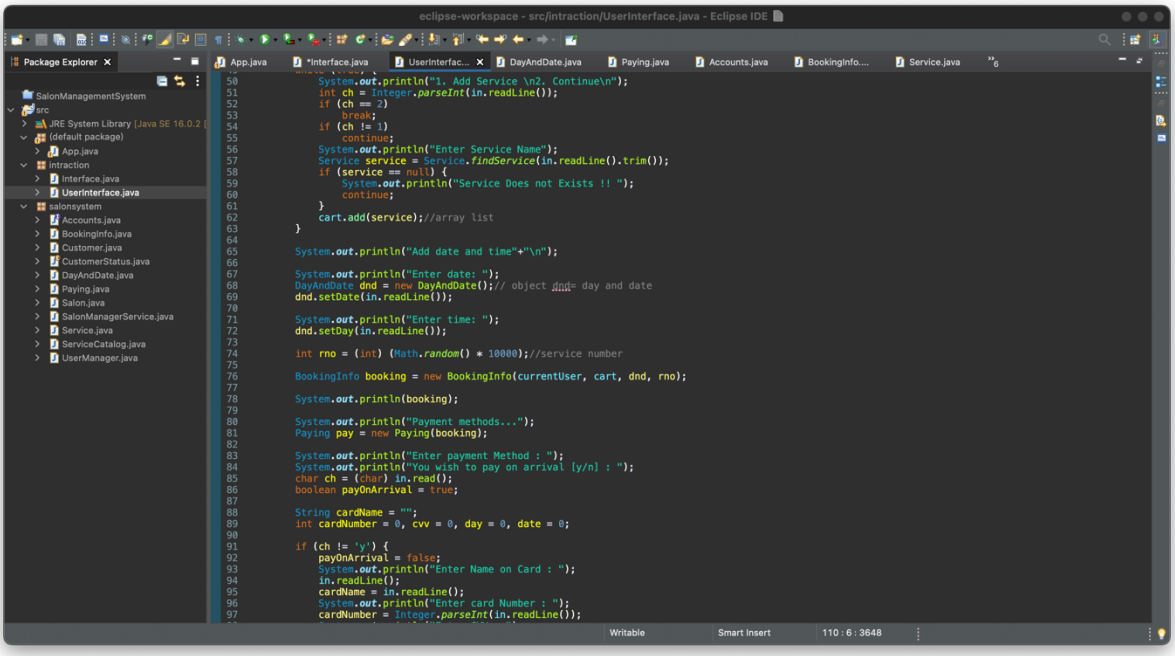

eclipse-workspace - src/salonsystem/SalonManagerService.java - Eclipse IDE
package salonsystem;
import java.util.Map;
public class SalonManagerService {
private Map<String, Salon> salons;
private static SalonManagerService instance = null;
private SalonManagerService() {
salons = new HashMap<>();
}
public static SalonManagerService getInstance() {
if (instance == null)
instance = new SalonManagerService();
return instance;
}
public int getID() {
return salons.size() + 1;
}
public boolean registerSalon(Salon salon) {
if (findByName(salon.getName()) == null) {
salons.put(salon.getName(), salon);
return true;
}
return false;
}
public boolean removeSalon(String name) {
if (findByName(name) == null)
return false;
salons.remove(name);
return true;
}
public Salon findByName(String name) {
return salons.getOrDefault(name, null);
}
}

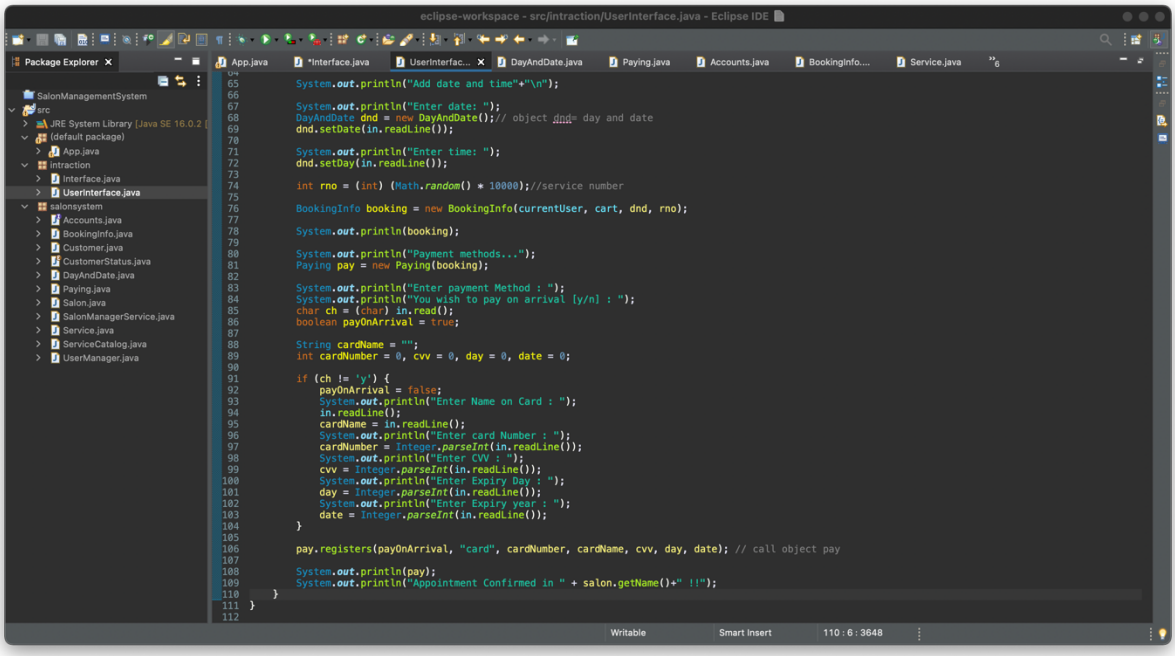

eclipse-workspace - src/salonsystem/Salon.java - Eclipse IDE
package salonsystem;
public class Salon {
private String name;
private String address;
private SalonManagerService salonManager;
public Salon(String name, String address) {
this.salonManager = SalonManagerService.getInstance();
this.name = name;
this.address = address;
}
public String getName() {
return this.name;
}
public void setName(String name) {
this.name = name;
}
public String getAddress() {
return this.address;
}
public void setAddress(String address) {
this.address = address;
}
public boolean registerSalon() {
return salonManager.registerSalon(this);
}
public boolean removeSalon() {
return salonManager.removeSalon(this.name);
}
public static Salon findSalon(String name) {
return SalonManagerService.getInstance().findByName(name);
}
}

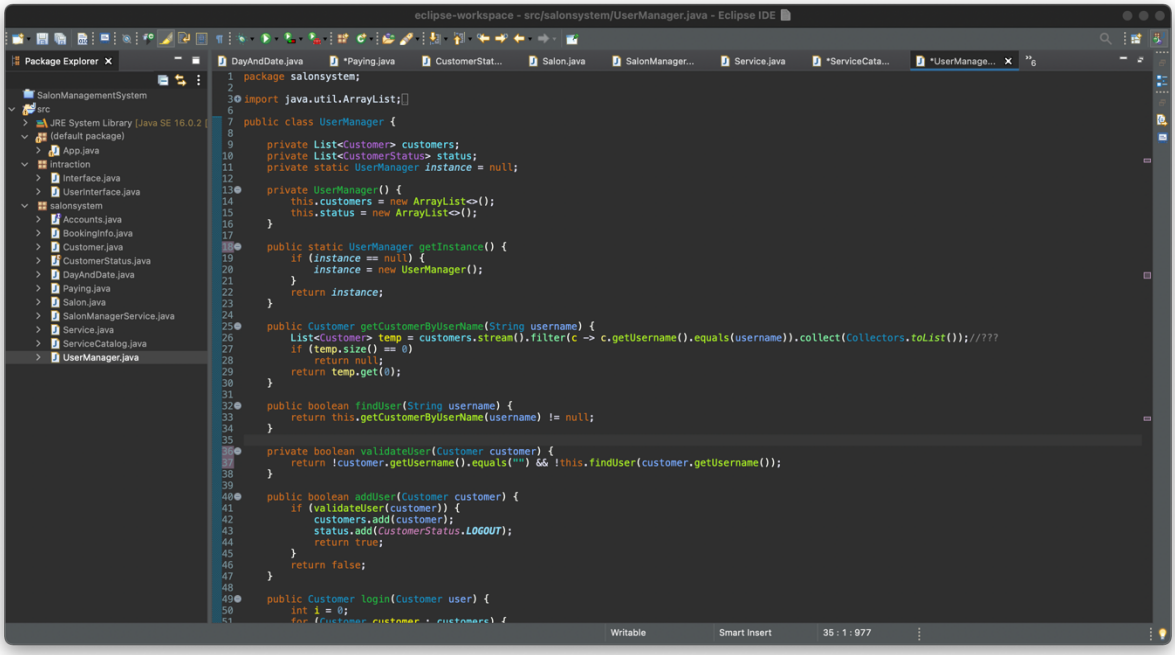

eclipse-workspace - src/salonsystem/Service.java - Eclipse IDE
package salonsystem;
public class Service {
private double price;
private String name;
private ServiceCatalog serviceCatalog;
public Service(String name, double price) {
this.name = name;
this.price = price;
this.serviceCatalog = ServiceCatalog.getInstance();
}
Service(String name) {
this(name, 0.0);
}
public boolean registerService() {
return serviceCatalog.register(this);
}
public boolean updatePrice() {
return serviceCatalog.update(this);
}
public String getName() {
return name;
}
public double getPrice() {
return price;
}
public boolean removeService() {
return serviceCatalog.remove(this);
}
public static Service findService(String name) {
return ServiceCatalog.getInstance().find(name);
}
}

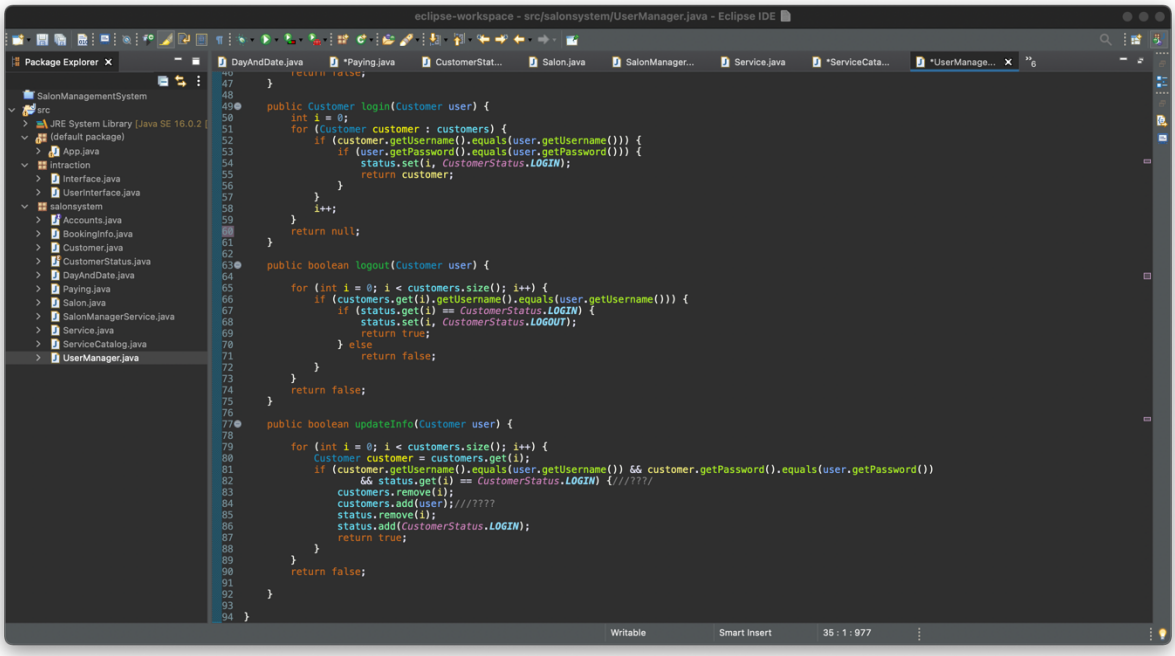

eclipse-workspace - src/salonsystem/ServiceCatalog.java - Eclipse IDE
package salonsystem;
import java.util.Collection;
public class ServiceCatalog {
private Map<String, Service> services;
private static ServiceCatalog instance = null;
private ServiceCatalog() {
this.services = new HashMap<>();////??
}
public static ServiceCatalog getInstance() {
if (instance == null)
instance = new ServiceCatalog();
return instance;
}
public Collection<Service> getAllServices() {
return services.values();
}
public Service find(String name) {
return services.getOrDefault(name, null);
}
public boolean register(Service service) {
if (find(service.getName()) == null) {
services.put(service.getName(), service);
return true;
}
return false;
}
public boolean remove(Service service) {
if (find(service.getName()) == null)
return false;
services.remove(service.getName());
return true;
}
public boolean update(Service service) {
if (remove(service) && register(service))
return true;
return false;
}
}


eclipse-workspace - src/intraction/UserInterface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package intraction;
import salonsystem.*;
public class UserInterface {
private BufferedReader in;
private Customer currentUser;
private List<Service> cart;
public UserInterface(Customer user) throws IOException {
in = new BufferedReader(new InputStreamReader(System.in));
currentUser = user;
cart = new ArrayList<>();
}
public void userMenu() throws IOException {
String str = "1. Make Booking \n2. logout";
System.out.println("Hi, " + currentUser.getUsername() + "\n");
loop: while (true) {
System.out.println(str);
String temp = in.readLine();
int ch = Integer.parseInt(temp.equals("") ? "0" : temp);
switch (ch) {
case 1:
createBooking();
break;
case 2:
currentUser.logout();
break loop;
}
}
}
public void createBooking() throws IOException {
System.out.println("Select Salon : ");
Salon salon = Salon.findSalon(in.readLine().trim());
if (salon == null) {
System.out.println("Salon does not Exits");
return;
}
while (true) {
System.out.println("1. Add Service \n2. Continue\n");
Writable
Smart Insert
110 : 6 : 3648


eclipse-workspace - src/intraction/UserInterface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
System.out.println("1. Add Service \n2. Continue\n");
int ch = Integer.parseInt(in.readLine());
if (ch == 2)
break;
if (ch != 1)
continue;
System.out.println("Enter Service Name");
Service service = Service.findService(in.readLine().trim());
if (service == null) {
System.out.println("Service Does not Exists !! ");
continue;
}
cart.add(service);//array list
}
System.out.println("Add date and time"+"\n");
System.out.println("Enter date: ");
DayAndDate dnd = new DayAndDate();// object dnd= day and date
dnd.setDate(in.readLine());
System.out.println("Enter time: ");
dnd.setDay(in.readLine());
int rno = (int) (Math.random() * 10000);//service number
BookingInfo booking = new BookingInfo(currentUser, cart, dnd, rno);
System.out.println(booking);
System.out.println("Payment methods...");
Paying pay = new Paying(booking);
System.out.println("Enter payment Method : ");
System.out.println("You wish to pay on arrival [y/n] : ");
char ch = (char) in.read();
boolean payOnArrival = true;
String cardName = "";
int cardNumber = 0, cvv = 0, day = 0, date = 0;
if (ch != 'y') {
payOnArrival = false;
System.out.println("Enter Name on Card : ");
in.readLine();
cardName = in.readLine();
System.out.println("Enter card Number : ");
cardNumber = Integer.parseInt(in.readLine());
Writable
Smart Insert
110 : 6 : 3648


eclipse-workspace - src/intraction/UserInterface.java - Eclipse IDE
Package Explorer
App.java
*Interface.java
UserInterfac...
DayAndDate.java
Paying.java
Accounts.java
BookingInfo....
Service.java
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2]
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
System.out.println("Add date and time"+"\n");
System.out.println("Enter date: ");
DayAndDate dnd = new DayAndDate();// object dnd= day and date
dnd.setDate(in.readLine());
System.out.println("Enter time: ");
dnd.setDay(in.readLine());
int rno = (int) (Math.random() * 10000);//service number
BookingInfo booking = new BookingInfo(currentUser, cart, dnd, rno);
System.out.println(booking);
System.out.println("Payment methods...");
Paying pay = new Paying(booking);
System.out.println("Enter payment Method : ");
System.out.println("You wish to pay on arrival [y/n] : ");
char ch = (char) in.read();
boolean payOnArrival = true;
String cardName = "";
int cardNumber = 0, cvv = 0, day = 0, date = 0;
if (ch != 'y') {
payOnArrival = false;
System.out.println("Enter Name on Card : ");
in.readLine();
cardName = in.readLine();
System.out.println("Enter card Number : ");
cardNumber = Integer.parseInt(in.readLine());
System.out.println("Enter CVV : ");
cvv = Integer.parseInt(in.readLine());
System.out.println("Enter Expiry Day : ");
day = Integer.parseInt(in.readLine());
System.out.println("Enter Expiry year : ");
date = Integer.parseInt(in.readLine());
}
pay.registers(payOnArrival, "card", cardNumber, cardName, cvv, day, date); // call object pay
System.out.println(pay);
System.out.println("Appointment Confirmed in " + salon.getName()+" !!");
}
}
Writable
Smart Insert
110 : 6 : 3648


eclipse-workspace - src/salonsystem/UserManager.java - Eclipse IDE
Package Explorer
DayAndDate.java
*Paying.java
CustomerStat...
Salon.java
SalonManager...
Service.java
*ServiceCata...
*UserManage...
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2]
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
package salonsystem;
import java.util.ArrayList;
public class UserManager {
private List<Customer> customers;
private List<CustomerStatus> status;
private static UserManager instance = null;
private UserManager() {
this.customers = new ArrayList<>();
this.status = new ArrayList<>();
}
public static UserManager getInstance() {
if (instance == null) {
instance = new UserManager();
}
return instance;
}
public Customer getCustomerByUserName(String username) {
List<Customer> temp = customers.stream().filter(c -> c.getUsername().equals(username)).collect(Collectors.toList());//???
if (temp.size() == 0)
return null;
return temp.get(0);
}
public boolean findUser(String username) {
return this.getCustomerByUserName(username) != null;
}
private boolean validateUser(Customer customer) {
return !customer.getUsername().equals("") && !this.findUser(customer.getUsername());
}
public boolean addUser(Customer customer) {
if (validateUser(customer)) {
customers.add(customer);
status.add(CustomerStatus.LOGOUT);
return true;
}
return false;
}
public Customer login(Customer user) {
int i = 0;
Writable
Smart Insert
35 : 1 : 977


eclipse-workspace - src/salonsystem/UserManager.java - Eclipse IDE
Package Explorer
DayAndDate.java
*Paying.java
CustomerStat...
Salon.java
SalonManager...
Service.java
*ServiceCata...
*UserManage...
SalonManagementSystem
src
JRE System Library [Java SE 16.0.2]
(default package)
App.java
intraction
Interface.java
UserInterface.java
salonsystem
Accounts.java
BookingInfo.java
Customer.java
CustomerStatus.java
DayAndDate.java
Paying.java
Salon.java
SalonManagerService.java
Service.java
ServiceCatalog.java
UserManager.java
}
public Customer login(Customer user) {
int i = 0;
for (Customer customer : customers) {
if (customer.getUsername().equals(user.getUsername())) {
if (user.getPassword().equals(user.getPassword())) {
status.set(i, CustomerStatus.LOGIN);
return customer;
}
}
i++;
}
return null;
}
public boolean logout(Customer user) {
for (int i = 0; i < customers.size(); i++) {
if (customers.get(i).getUsername().equals(user.getUsername())) {
if (status.get(i) == CustomerStatus.LOGIN) {
status.set(i, CustomerStatus.LOGOUT);
return true;
} else
return false;
}
}
return false;
}
public boolean updateInfo(Customer user) {
for (int i = 0; i < customers.size(); i++) {
Customer customer = customers.get(i);
if (customer.getUsername().equals(user.getUsername()) && customer.getPassword().equals(user.getPassword())
&& status.get(i) == CustomerStatus.LOGIN) {///???/
customers.remove(i);
customers.add(user);///????
status.remove(i);
status.add(CustomerStatus.LOGIN);
return true;
}
}
return false;
}
}
Writable
Smart Insert
35 : 1 : 977

Conclusion:

in conclusion, the problem that we tried to solve is now solved successfully by this program. this program can help to reduce all the problems that salons can face during the workday. customers now can book an appointment while they are at home in a very easy way. We hope that this program will be used.