

Mini Project Report

D. Y. Patil College of Engineering & Technology

(Approved by AICTE, New Delhi, Govt. of Maharashtra and Permanently Affiliated to Shivaji University
Kolhapur)

(An Autonomous Institute)



Student Details

Name	Roll No	Signature
Shravani Phutane	16	
Bhakti Powar	18	
Siddhi Nilkanth	20	
Rujuta Patil	21	
Arpita Patil	25	

Class: SY 'C'

Subject: Object Oriented Programming using C++

Project Title

Hotel Management System using C++

Objective

The objective of this project is to create a simple Hotel Management System that records customer details, calculates bills, applies discounts, and stores data in a file using C++ Object-Oriented Programming concepts.

Concepts Used in the Project

- **Classes and Objects:** Used to represent hotel and customer details.
- **Encapsulation:** Data members are private, accessed through setters and getters.
- **Constructor:** Default and parameterized constructors initialize objects.
- **Function Overloading:** The totalBill() function is overloaded to calculate with and without discount.
- **Operator Overloading:** The == operator is overloaded to compare room numbers.
- **File Handling:** Used ofstream to save data permanently in a text file.
- **Data Formatting:** Used iomanip for proper alignment of output in tabular format.
- **Array of Objects:** Used to store multiple customer records.

Project Description

The Hotel Management System allows the user to input customer details such as room number, customer name, number of days stayed, and rate per day. The program calculates the total bill, applies discounts using function overloading, and saves all records to a file named hotel_data.txt for future reference.

Features

- Enter and display customer details
- Automatically calculate hotel bills
- Apply discounts dynamically
- Save all records to a text file
- Search customer by room number using operator overloading

Code:

```
#include <iostream>

#include <fstream>

#include <iomanip>

#include <string>

using namespace std;

class Hotel
{
private:
    int roomNo;

    string customerName;

    int days;

    float ratePerDay;
public:
    // Default Constructor
    Hotel() : roomNo(0), customerName(""), days(0), ratePerDay(0.0) {}

    // Parameterized Constructor
    Hotel(int r, string n, int d, float rate)
    {
        roomNo = r;

        customerName = n;

        days = d;

        ratePerDay = rate;
    }

    void setData()
    {
```

```

    cout << "\nEnter Room No: ";
    cin >> roomNo;
    cout << "Enter Customer Name: ";
    cin.ignore();
    getline(cin, customerName);
    cout << "Enter No. of Days Stayed: ";
    cin >> days;
    cout << "Enter Rate per Day: ";
    cin >> ratePerDay;
}

void getData()
{
    cout << left << setw(10) << roomNo
         << setw(20) << customerName
         << setw(10) << days
         << setw(10) << ratePerDay
         << setw(10) << totalBill() << endl;
}

// Function Overloading for bill calculation

float totalBill()
{
    return days * ratePerDay;
}

float totalBill(float discountPercent)
{
    float amount = totalBill();
    float discount = (amount * discountPercent) / 100;

```

```

        return amount - discount;
    }

// Operator Overloading
    bool operator==(int r)
    {
        return roomNo == r;
    }

// File handling
    void saveToFile()
    {
        ofstream fout("hotel_data.txt", ios::app);

        fout << roomNo << "," << customerName << "," << days << "," <<
ratePerDay << "\n";

        fout.close();
    }
};

int main()
{
    Hotel h[10];

    int n;

    cout << "Enter number of customers: ";

    cin >> n;

    for (int i = 0; i < n; i++)
    {
        cout << "\n--- Enter details for Customer " << i + 1 << " ---\n";

        h[i].setData();

        h[i].saveToFile(); // Saving each record
    }
}

```

```

cout << "\n--- Hotel Billing Details ---\n";

cout << left << setw(10) << "RoomNo" << setw(20) << "Name" << setw(10)
    << "Days" << setw(10) << "Rate" << setw(10) << "Bill" << endl;

for (int i = 0; i < n; i++)
{
    h[i].getData();
}

// Demonstrate function overloading

int searchRoom;

cout << "\nEnter Room No to find discounted bill: ";

cin >> searchRoom;

for (int i = 0; i < n; i++)
{
    if (h[i] == searchRoom) // Operator Overloading
    {
        cout << "Enter discount percent: ";

        float d;

        cin >> d;

        cout << "Discounted Bill: " << h[i].totalBill(d) << endl;
    }
}

cout << "\nData saved successfully in 'hotel_data.txt'\n";

return 0;
}

```

Output

Enter number of customers: 2

--- Enter details for Customer 1 ---

Enter Room No: 101

Enter Customer Name: Rahul Patil

Enter No. of Days Stayed: 3

Enter Rate per Day: 1500

--- Enter details for Customer 2 ---

Enter Room No: 102

Enter Customer Name: Sneha Desai

Enter No. of Days Stayed: 2

Enter Rate per Day: 2000

--- Hotel Billing Details ---

RoomNo	Name	Days	Rate	Bill
101	Rahul Patil	3	1500	4500
102	Sneha Desai	2	2000	4000

Enter Room No to find discounted bill: 101

Enter discount percent: 10

Discounted Bill: 4050

Data saved successfully in 'hotel_data.txt'

Conclusion

This project demonstrates the practical use of Object-Oriented Programming (OOP) concepts in C++. It efficiently manages hotel billing operations and shows how encapsulation, file handling, and operator overloading can be applied in real-world scenarios.

Teacher's Remark

.....

.....

Signature of Teacher: _____