Course Work

by Valdir Santos Fonseca

Submission date: 05-Apr-2018 09:39PM (UTC+0100)
Submission ID: 86090170
File name: w1669150.txt (8.05K)
Word count: 1258
Character count: 6301

```
//Valdir Santos Fonseca – w1669150

// CourseWork_Valdir.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <string>

#include<iostream>

#include<fstream>

using  namespace std;

//check rooms available

void roomsAvailable(int rooms_available, string hotel_rooms[])

{

    for (int i = 0; i < 10; i++)

    {
```

```
        if (hotel_rooms[i] == "e") // check if the room is empty or not
        {
            rooms_available++;
            cout << "Room " << i+1 << " is available!" << endl;
        }
        else
            cout << "Room " << i + 1 << " is not available. Sorry." << endl;
        // if it's not empty it will shows the information that it's not available
    }
    if (rooms_available == 0)
        cout << "Sorry, there is no available rooms." << endl;
    //by the end if there is no rooms available it will show this information
}
```

```
// Create a new customer by setting the room and the customer's name

void addNewCustomer(string customer_name, string hotel_rooms[], int

new_customer_room)

{

    if ((hotel_rooms[new_customer_room - 1] != "e") || ((new_customer_room >10) &&

(new_customer_room < 1)))

        // check if the room is empty or not

        cout << "Room not available. Please select another room!!" << endl;

    else

        hotel_rooms[new_customer_room-1] = customer_name;

}

//Return an existent customer by introducing the room number

string getCustomer(string hotel_rooms[], int num_room)
```

```
{
    string customer_name = " ";
    for (int i = 0; i < 10; i++)
    {
        if (i == num_room && hotel_rooms[i] != "e")
        {
            customer_name = hotel_rooms[i];
        }
    }
    return customer_name;
}
//Initialiase all rooms as empty
void emptyRooms(string hotel_rooms[])
```

```
{
    for (int i = 0; i < 10; i++) // going through all positions and set them as empty
    {
        hotel_rooms[i] = "e";
        cout << "Room - " << i+1 << " -> " << hotel_rooms[i] << endl;
    }
}
//Print the status of the rooms
void printStatus(string hotel_rooms[])
{
    for (int i = 0; i < 10; i++)// going through all positions and print the status of the
rooms
    {
```

```
            cout << "Room - " << i + 1 << " -> " << hotel_rooms[i] << endl;
    }
}
//Show the menu
void showMenu()
{
    cout << "Please introduce a valid Character:" << endl;
    cout << "To initialiase Rooms = 'I'" << endl;
    cout << "To Add a new Customer = 'A'" << endl;
    cout << "To View all rooms = 'V'" << endl;
    cout << "To empty rooms = 'E'"<<endl;
    cout << "To delete a customer from a specific room = 'D'" << endl;
    cout << "To search a room from a customer name = 'F'" << endl;
```

```
    cout << "To store data into a file = 'S'" << endl;

    cout << "To Load data from a file = 'L'" << endl;

    cout << "To view rooms ordered = 'O'" << endl;

    cout << "Thank you!!" << endl << endl;

    cout << "Please Introduce a Character: ";

}

//Display empty rooms

void displayEmptyRooms(string hotel_rooms[])

{

    for (int i = 0; i < 10; i++)// going through all positions and print just the empty rooms

    {

        if (hotel_rooms[i] == "e")

        {
```

```
            cout << "Room - " << i + 1 << " -> " << hotel_rooms[i] << endl;
        }
    }
}
//Deleting a costumer from a room
void deleteCustomer(string hotel_rooms[], string customer_name)
{
    int i = 0;
    bool find = false;
    while(!find && i < 10)//searching for customer introduced by the user
    {
        if (hotel_rooms[i] == customer_name)
        {
```

```
            hotel_rooms[i] = "e";

            find = true; // if customer found it will set the rooms as empty and while loop
it will end

        }

        else if (i == 9 && find == false)

            cout << endl << "Customer doesn't exist on our hotel. Please introduce a
existent customer!!" << endl;

        //print information if customer was not found

        i++;

    }

}

void searchingCustomer(string hotel_rooms[], string customer_name)

{
```

```
    int i = 0;
    bool find = false;
    while (!find && i < 10)//searching for customer introduced by the user
    {
        if (hotel_rooms[i] == customer_name)
        {
            cout<<endl << "Room - " << i + 1 << " --> " << hotel_rooms[i] << endl;
            find = true;//if customer found it will show which is in and while loop it will end
        }
        else if (i == 9 && find == false)
            cout<< endl<<"Customer doesn't exist on our hotel. Please introduce a existent
customer!!"<<endl;
        //print information if customer was not found
```

```
        i++;
    }
}
//Store the data of the array into the file rooms.txt
void storeFile(string hotel_rooms[])
{
    ofstream myfile;// this is to indicate that i'm going to write into a file
    myfile.open("rooms.txt"); // open the file
    for (int i = 0; i < 10; i++)//going through all positions and print print the information
on the array of string
    {
        myfile << "Room - " << i + 1 << " --> " << hotel_rooms[i] << endl;
```

```
        // giving instructions to write everything about the rooms into a line and then
change to the next line
    }
    myfile.close(); //close the file always after writing
}
//Load the data from the file
void loadFile()
{
    string s;
    ifstream myfile; // this is to indicate that i'm going to read from a file
    myfile.open("rooms.txt"); // Open the file rooms.txt
    while (getline(myfile, s)) // using a getline we going to read line by time and store it
into a string
```

```
    {
        cout << s << endl; //going to show each line that I've read
    }
    myfile.close(); //close the file always after reading
}
//View rooms Ordered alphabetically by the main customer's name
void orderAphabetical(string hotel_rooms[])
{
    string temp;
    for (int i = 0; i < 10; i++)// Checking position by position
    {
        for (int j = i+1; j < 10; j++)//compare the first position to the others and swope if
it's lower than the first position
```

```
        {
            if (hotel_rooms[j] < hotel_rooms[i])
            {
                temp = hotel_rooms[i];//put the value of the position i into a temporary string
                hotel_rooms[i] = hotel_rooms[j];
                hotel_rooms[j] = temp;
            }
        }
        cout << hotel_rooms[i] << endl;//print the name after being ordered
    }
}
int main()
{
```

```
    string hotel_rooms[10];

    string customer_name;

    int new_customer_room;

    int rooms_available = 0;

    char continue_menu = ' ';

    char menu;

    //it will show the menu at least once and shows the options that the user can
introduce

    do

    {

        showMenu();

        cin >> menu;

        switch (menu)
```

```
        {
        case'I':case 'i':// to initialiase the rooms as a empty 'e'
            emptyRooms(hotel_rooms);
            break;
        case'A':case'a': //Add a new customer into a room
            printStatus(hotel_rooms);
            cout << endl;
            roomsAvailable(rooms_available, hotel_rooms);
            cout << endl<< "Please introduce number of the room and the customer." <<
endl;
            cout << "Customer room: ";
            cin >> new_customer_room;
            cout << "Customer name: ";
```

```
        cin >> customer_name;
        addNewCustomer(customer_name, hotel_rooms, new_customer_room);
        cout << endl;
        printStatus(hotel_rooms);
        break;
    case'V':case'v': // View the status os the rooms
        printStatus(hotel_rooms);
        break;
    case'E':case'e':// Display the empty rooms(Rooms available)
        displayEmptyRooms(hotel_rooms);
        break;
    case'D':case'd'://delete a customer from a room
        printStatus(hotel_rooms);
```

```
            cout << endl << "Please introduce the customer name that you want to delete:
";

            cin >> customer_name;

            deleteCustomer(hotel_rooms, customer_name);

            cout << endl;

            printStatus(hotel_rooms);

            break;

        case'F':case'f'://searching a customer by introducing the customer's name

            printStatus(hotel_rooms);

            cout << endl << "Please introduce the customer name to check which room: ";

            cin >> customer_name;

            searchingCustomer(hotel_rooms,customer_name);

            break;
```

```
case'S':case's':// Write the information of the string hotel into a file
    storeFile(hotel_rooms);
    break;
case'L':case'l':// Read the information from the file and print
    loadFile();
    break;
case'O':case'o':
    printStatus(hotel_rooms);
    cout <<endl << "Alphabetical Order: " << endl;
    orderAphabetical(hotel_rooms);
    break;
default:
    cout << menu << " is not a valid character!!" << endl;
```

```
            break;

        }

        cout << endl << "You wish to continue? Yes(Y) or No(N): ";//Asking the user if he
wants to continue or not

        cin >> continue_menu;

    }while(continue_menu == 'Y' || continue_menu == 'y');

    return 0;

}
```