



Bahria University, Islamabad
Department of Software Engineering
Data Structures & Algorithms Lab
(Spring-2024)

Teacher: RAHEELA AMBRIN

Student : Abdul Rafay

Enrollment : 01-131232-004

Lab Journal: 1

Date: 23 / 09 / 24

Comments:

Signature

Code:

All the code files are uploaded on GitHub: https://github.com/CharlieFour/DSA_Lab

House.h

```
#pragma once

#include <string>
#include <vector>

class House
{
private:
    std::string ownerName;
    std::string address;
    int bedrooms;
    int price;
    float ratio;

public:
    House(){}

    //getters
    std::string getOwnerName() const;
    std::string getAddress() const;
    int getBedrooms() const;
    int getPrice() const;
    float getRatio() const;

    //setters
    void setOwnerName(std::string);
    void setAddress(std::string);
    void setBedrooms(int);
    void setPrice(int);
```

```
void setRatio(float);  
  
//file handling  
  
void loadFile(std::vector<House> &houses);  
  
void saveFile(const std::vector<House> &houses);  
  
};
```

House.cpp

```
#include <string>  
#include <iostream>  
#include <fstream>  
#include "../Header_files/house.h"  
#include <vector>  
#include <sstream>  
  
std::string House::getOwnerName() const  
{  
    return ownerName;  
}  
  
std::string House::getAddress() const  
{  
    return address;  
}  
  
int House::getBedrooms() const  
{  
    return bedrooms;  
}  
  
int House::getPrice() const  
{  
    return price;  
}  
  
float House::getRatio() const  
{  
    return ratio;  
}
```

```
//----setters----
void House::setOwnerName(std::string name)
{
    ownerName = name.length() > 20 ? name.substr(0, 20) : name;
}

void House::setAddress(std::string addr)
{
    address = addr.length() > 20 ? addr.substr(0, 20) : addr;
}

void House::setBedrooms(int bed)
{
    bedrooms = bed;
}

void House::setPrice(int pri)
{
    price = pri;
}

void House::setRatio(float r)
{
    ratio = r;
}

//----file handling----
void House::loadFile(std::vector<House> &houses)
{
    std::ifstream inFile("../Data/house_data.csv");
    if (inFile.is_open())
    {
        //declare the variables
        House house;
        std::string ownerName, city;
        int price, bedrooms;
        std::string line; // to store a row
        std::getline(inFile, line); //to store the row

        while(getline(inFile, line))
        {
            std::stringstream ss(line);

            // Extract the values separated by commas
            std::getline(ss, ownerName, ',');
            ss >> price;
```

```
ss.ignore(); // Ignore the comma after price
ss >> bedrooms;
ss.ignore(); // Ignore the comma after bedrooms
std::getline(ss, city, ',');
house.setOwnerName(ownerName);
house.setPrice(price);
house.setAddress(city);
house.setBedrooms(bedrooms);

houses.push_back(house);
}
}
else
{
    std::cout << "File not found at the specified location." << std::endl;
}
}

void House::saveFile(const std::vector<House> &houses)
{
    std::ofstream outFile("../Data/house_data.csv");

    //Header
    outFile << "Owner name,Price,Bedrooms,City\n";
    //data
    for (const auto&house : houses)
    {
        outFile << house.getOwnerName() << ","
            << house.getPrice() << ","
            << house.getBedrooms() << ","
            << house.getAddress() << "\n";
    }
    outFile.close();
    std::cout << "File saved successfully." << std::endl;
}
```

Main.cpp

```
#include <string>
#include "../Header_files/house.h"
#include <iostream>
#include <iomanip>
#include <vector>

using namespace std;
vector<House> available;

void addHouse();
void displayHouse();
void filterHouse();
void menu();
int main()
{
    House house;
    house.loadFile(available);
    menu();
    house.saveFile(available);
    system("pause");
}

void addHouse()
{
    string ownerName;
    string address;
    int bedrooms;
    int price;
    bool check;
    House house;
    cout << "-----Enter the House details-----" << endl;

    do
    {
        cin.ignore();
        cout << string(55, '-') << endl;
        cout << "Owner : ";
        getline(cin, ownerName);
        house.setOwnerName(ownerName);
```

```
    cout << "Address : ";
    getline(cin, address);
    house.setAddress(address);
    cout << "Bedrooms : ";
    cin >> bedrooms;
    house.setBedrooms(bedrooms);
    cout << "Price : ";
    cin >> price;
    house.setPrice(price);
    house.setRatio(price / bedrooms);

    available.push_back(house);

    cout << string(55, '-') << endl;
    char c;
    cout << "Do you want to add another house? (Y for yes, N for no) : ";
    cin >> c;
    if(c == 'Y' || c == 'y')
    {
        check = true;
    }
    else if (c == 'N' || c == 'n')
    {
        check = false;
    }
}
while(check == true );

}
void displayHouse()
{
    House house;
    cout << left << setw(25) << "Owner"
        << setw(30) << "Address"
        << setw(12) << "Bedrooms"
        << setw(10) << "Price" << endl;

    cout << string(77, '-') << endl;

    for (const auto& item : available)
```

```
{
    cout << left << setw(25) << item.getOwnerName()
        << setw(30) << item.getAddress()
        << setw(12) << item.getBedrooms()
        << setw(10) << item.getPrice() << endl;
}
}

void filterHouse()
{
    string address;
    int bedrooms;
    int price;
    float ratio;
    int choice;
    cout << "Note: The developers are working on the AI filter system, that the system predict the
price accordingly, so the house by location is not available at the moment." << endl;

    cout << "1. Filter by price" << endl;
    cout << "2. Filter by bedrooms" << endl;
    cout << "3. Filter by both price/size ratio" << endl;
    cout << "Enter your choice : ";
    cin >> choice;
    switch (choice)
    {
        case 1 :
            cout << "Enter the price max price : ";
            cin >> price;
            for (auto& item : available)
            {
                if (item.getPrice() <= price)
                {
                    cout << left << setw(25) << "Owner"
                        << setw(30) << "Address"
                        << setw(12) << "Bedrooms"
                        << setw(10) << "Price" << endl;

                    cout << string(77, '-') << endl;

                    cout << left << setw(25) << item.getOwnerName()
                        << setw(30) << item.getAddress()

```



```
        << setw(12) << item.getBedrooms()
        << setw(10) << item.getPrice() << endl;
    }
}
break;
case 2 :
    cout << "1. Specified number of rooms" << endl;
    cout << "2. Largest house" << endl;
    cout << "Enter your choice : ";
    cin >> choice;
    switch (choice)
    {
        case 1 :
            cout << "Enter the number of bedrooms : ";
            cin >> bedrooms;
            for (auto& item : available)
            {
                if (item.getBedrooms() == bedrooms)
                {
                    cout << left << setw(25) << "Owner"
                        << setw(30) << "Address"
                        << setw(12) << "Bedrooms"
                        << setw(10) << "Price" << endl;

                    cout << string(77, '-') << endl;

                    cout << left << setw(25) << item.getOwnerName()
                        << setw(30) << item.getAddress()
                        << setw(12) << item.getBedrooms()
                        << setw(10) << item.getPrice() << endl;
                }
            }
            break;
        case 2 :
            int max = 0;
            for (auto& item : available)
            {
                if (item.getBedrooms() > max)
                {
                    max = item.getBedrooms();
                }
            }
            break;
    }
}
```

```
        }
    }
    for (auto& item : available)
    {
        if (item.getBedrooms() == max)
        {
            cout << left << setw(25) << "Owner"
                << setw(30) << "Address"
                << setw(12) << "Bedrooms"
                << setw(10) << "Price" << endl;

            cout << string(77, '-') << endl;

            cout << left << setw(25) << item.getOwnerName()
                << setw(30) << item.getAddress()
                << setw(12) << item.getBedrooms()
                << setw(10) << item.getPrice() << endl;
        }
    }
    break;

}
break;
case 3 :
    cout << "Enter the price : " ;
    cin >> price;
    cout << "Enter the number of bedrooms : ";
    cin >> bedrooms;
    ratio = price / bedrooms;
    for (const auto & item : available)
    {
        if (item.getRatio() <= ratio && item.getBedrooms() == bedrooms)
        {
            cout << left << setw(25) << "Owner"
                << setw(30) << "Address"
                << setw(12) << "Bedrooms"
                << setw(10) << "Price" << endl;

            cout << string(77, '-') << endl;
```

```
        cout << left << setw(25) << item.getOwnerName()
            << setw(30) << item.getAddress()
            << setw(12) << item.getBedrooms()
            << setw(10) << item.getPrice() << endl;
    }
}
break;
default :
    cout << "Invalid choice" << endl;
}
}
void menu()
{
    char choice;
    do
    {
        cout << "-----" << endl;
        cout << "1. Add a new house" << endl;
        cout << "2. Display all houses" << endl;
        cout << "3. Filter all houses" << endl;
        cout << "4. Exit" << endl;
        cout << "-----" << endl;
        cout << "Enter your choice : ";
        cin >> choice;
        if (choice == '1')
        {
            addHouse();
        }
        else if (choice == '2')
        {
            cin.ignore();
            displayHouse();
        }
        else if (choice == '3')
        {
            cin.ignore();
            filterHouse();
        }
    }
    while(choice != '4');
}
```

Execution ScreenShots:

```
1. Add a new house
2. Display all houses
3. Filter all houses
4. Exit
-----
Enter your choice : 1
-----Enter the House details-----
-----
Owner : Abdul Rafay
Address : Islamabad
Bedrooms : 4
Price : 950000
-----
Do you want to add another house? (Y for yes, N for no) : y
-----
Owner : Jibran
Address : Multan
Bedrooms : 5
Price : 90
-----
Enter your choice : 3
Note: The developers are working on the AI filter system, that the system predict the price accordingly, so the house by
location is not available at the moment.
1. Filter by price
2. Filter by bedrooms
3. Filter by both price/size ratio
Enter your choice : 1
Enter the price max price : 50000
Owner          Address          Bedrooms    Price
-----
Jibran          Multan          5           90
-----
1. Add a new house
2. Display all houses
3. Filter all houses
4. Exit
-----
Enter your choice : 2
Owner          Address          Bedrooms    Price
-----
Jibran          Multan          5           90
-----
1. Add a new house
2. Display all houses
3. Filter all houses
4. Exit
-----
Enter your choice : 3
Note: The developers are working on the AI filter system, that the system predict the price accordingly, so the house by
location is not available at the moment.
1. Filter by price
2. Filter by bedrooms
3. Filter by both price/size ratio
Enter your choice : 3
Enter the price : 50000
Enter the number of bedrooms : 5
Owner          Address          Bedrooms    Price
-----
Jibran          Multan          5           90
-----
```