**МІНІСТЕРСТВО**

**ОСВІТИ ТА НАУКИ УКРАЇНИ**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Кафедра інформаційних систем та технологій**

**Звіт**

**з лабораторної  роботи № 4**

**«Інтерфейси. Абстрактні класи.»**

**з дисципліни**

**«Програмування – 2. ООП»**

**Варіант №21**

|  |  |  |
| --- | --- | --- |
| **Перевірив:**  **доц. Корнага Ярослав Ігорович** |  | **Виконала:** |
|  |  | **Студентка гр. ІС-12 ФІОТ**  **Мельникова Катерина** |
|  |  | **1 курс,**  **залікова книжка № IC-1222** |

**Київ 2021**

**Тема:** Інтерфейси. Абстрактні класи.

**Завдання:**

* Проаналізувати завдання, виділити інформаційні об’єкти та дії. Визначити правильну ієрархію об’єктів
* Створити базовий суперклас (абстрактний клас або інтерфейс) і визначити загальні методи для даного класу. Створити підкласи, в які додати специфічні властивості та методи. Частину методів перевизначити.
* Розробити програму з використанням абстрактних класів та інтерфейсів. Чітко розуміти, де доцільно використати суперклас, а де звичайний.
* При розробці використовувати наслідування та поліморфізм
* У всіх класах повинні бути реалізовані доцільні для класу методи, навіть якщо це не вказано у завданні
* Використовувати об’єкти підкласів для моделювання реальних ситуацій на об’єктів

**21.** Створити суперклас Будівля і підкласи Громадська Будівля, Житлова Будівля, Кінотеатр, Готель. За допомогою конструктора задати дату побудови будівлі. Визначити вартість оренди одного місця в будівлі. Визначити вартість вхідного квитка в громадську будівлю. Визначити місткість будівлі.

Код:

Building.cs

public interface IRent

{

public int CountRent(int price, int quantity, int days);

}

public interface ITicket

{

public int CountPrice(int price, int customer\_age);

}

public abstract class Building

{

public string Date { get; set; }

public int Square { get; set; }

public int Floors { get; set; }

public Building(string date, int square, int floors)

{

this.Date = date;

this.Square = square;

this.Floors = floors;

}

public abstract int capacity();

}

public class PublicBuilding: Building, ITicket

{

public PublicBuilding(string date, int square, int floors): base(date, square, floors) {}

public int CountPrice(int price, int customer\_age)

{

if (customer\_age <= 6 || customer\_age >= 60)

{

return price/2;

}

return price;

}

public override int capacity()

{

return (this.Square \* this.Floors) / 5;

}

}

public class ResidentialBuilding: Building, IRent

{

public ResidentialBuilding(string date, int square, int floors) : base(date, square, floors) { }

public int CountRent(int price\_per\_meter, int square, int days)

{

return price\_per\_meter \* square \* days;

}

public override int capacity()

{

return (this.Square \* this.Floors) / 8;

}

}

public class Cinema: Building, ITicket

{

public Cinema(string date, int square, int floors) : base(date, square, floors) { }

public int Price { get; set; }

public int CountPrice(int price, int customer\_age)

{

if (customer\_age <= 6 || customer\_age >= 60)

{

return (price\*8)/10;

}

return price;

}

public override int capacity()

{

return (this.Square \* this.Floors) / 4;

}

}

public class Hotel: Building, IRent

{

public Hotel(string date, int square, int floors) : base(date, square, floors) { }

public int CountRent(int price\_per\_room, int number\_of\_rooms, int days)

{

return price\_per\_room \* number\_of\_rooms \* days;

}

public override int capacity()

{

return (this.Square \* this.Floors) / 7;

}

}

Program.cs

using System;

namespace Lab\_4

{

class Program

{

static void Main(string[] args)

{

Cinema florencia = new Cinema("12.03.2002", 150, 3);

Console.WriteLine(florencia);

Console.WriteLine("Дата побудови: " + florencia.Date);

Console.WriteLine("Вартiсть квитка (пiльговий): " + florencia.CountPrice(100, 3));

Console.WriteLine("Вартiсть квитка (звичайний): " + florencia.CountPrice(100, 30));

Console.WriteLine("Мiсткiсть: " + florencia.capacity());

Hotel California = new Hotel("15.04.1976", 400, 7);

Console.WriteLine(California);

Console.WriteLine("Дата побудови: " + California.Date);

Console.WriteLine("Вартiсть оренди: " + California.CountRent(100, 3, 10));

Console.WriteLine("Мiсткiсть: " + California.capacity());

PublicBuilding museum = new PublicBuilding("23.08.1923", 200, 4);

Console.WriteLine(museum);

Console.WriteLine("Дата побудови: " + museum.Date);

Console.WriteLine("Вартiсть квитка (звичайний): " + museum.CountPrice(50, 34));

Console.WriteLine("Вартiсть квитка (пiльговий): " + museum.CountPrice(50, 84));

Console.WriteLine("Мiсткiсть: " + museum.capacity());

ResidentialBuilding house = new ResidentialBuilding("01.01.2012", 150, 20);

Console.WriteLine(house);

Console.WriteLine("Дата побудови: " + house.Date);

Console.WriteLine("Вартiсть оренди: " + house.CountRent(150, 50, 30));

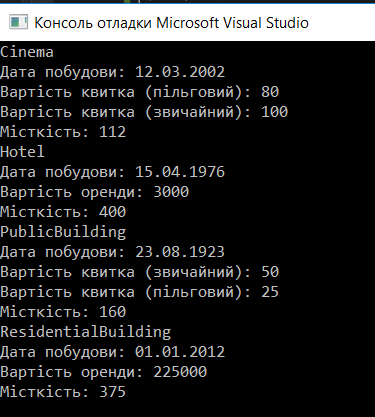
Console.WriteLine("Мiсткiсть: " + house.capacity());

}

}

}

Результат виконання програми:



Висновок:

Отже, під час виконання цієї лабораторної роботи, я навчилась працювати з абстрактними класами та інтерфейсами у мові програмування С#.