**10 МЕХАНИЗМ НАСЛЕДОВАНИЯ. ОТНОШЕНИЯ МЕЖДУ КЛАССАМИ**

Задание 1. Базовый класс: Квартира (поля: название, стоимость 1м 2, площадь) Метод: Стоимость квартиры. Потомок: Квартира в центре (поле название района) Изменения в потомках: Увеличить стоимость с учетом надбавки за расположение на 0.01 стоимости квартиры.

Листинг программы:

using System;

namespace task1

{

class Flat

{

public string name;

public double price1m2;

public double square;

public Flat(string name, double price1m2, double square)

{

this.name = name;

this.price1m2 = price1m2;

this.square = square;

}

public double Price()

{

double price = square \* price1m2;

return price;

}

}

class FlatCenter:Flat

{

private string district;

public FlatCenter(string name, double price1m2, double square,string district):base(name,price1m2,square)

{

this.name = name;

this.price1m2 = price1m2;

this.square = square;

this.district = district;

}

public new double Price()

{

double price = base.Price() + base.Price()/100;

return price;

}

}

class Program

{

static void Main(string[] args)

{

Flat a = new Flat("lol", 3, 14);

Console.WriteLine("Цена обычной квартиры: " + a.Price());

FlatCenter b = new FlatCenter("lol1", 3, 14,"Московский");

Console.WriteLine("Цена квартиры в центре : " + b.Price());

}

}

}

Таблица 10.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | 42, 42.42 |

Анализ результатов:



Рисунок 10.1 – Результат работы программы

Источник – собственная разработка

Задание 2. В рамках консольного приложения разработать класс В-наследник класса А (класс А с полями а и b и свойством с. Свойство – значение выражения над полями а и b (выражение и типы полей – см. вариант в таблице 1). Поля инициализировать при объявлении класса. Конструктор оставить по умолчанию. Проследить, чтобы поля а и b напрямую в других классах были недоступны.) с полем d и свойством с2. Свойство с2 – результат вычисления выражения над полями a, b, d. В теле свойства использовать управляющий оператор (см. вариант в таблице 1) Для класса В определить 2 конструктора: один – наследуется от конструктора класса А, второй –собственный. В теле программы создать объекты классов А и В, продемонстрировав работу всех конструкторов. Вывести значения свойства на экран.

Листинг программы:

using System;

namespace task2

{

class A

{

protected int a = 9;

protected int b = 3;

protected int c = 0;

public A()

{

}

public A(int a, int b)

{

this.a = a;

this.b = b;

}

public int C

{

get

{

return c;

}

set

{

c = value;

if (a > 5)

{

c \*= b;

}

else if (a == 5)

{

c++;

}

else

{ c = a - b; }

}

}

}

class B : A

{

protected int d;

public B(int d) : base()

{

this.d = d;

}

public B(int d, int a, int b) : base(a, b)

{

this.d = d;

}

public int C2

{

get

{

return c;

}

set

{

c = value;

if (c <= d)

{

c = 66;

}

else if (c > a)

{

c \*= a;

}

else

{

c += b;

}

}

}

}

class Program

{

static void Main(string[] args)

{

A a = new A(3, 6);

a.C = 4;

Console.WriteLine(a.C);

B b = new B(0, 10, 6);

b.C2 = 17;

Console.WriteLine(b.C2);

B bb = new B(10);

bb.C2 = 9;

Console.WriteLine(bb.C2);

}

}

}

Таблица 10.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | -3, 170, 66 |

Анализ результатов:



Рисунок 10.2 – Результат работы программы

Источник – собственная разработка