

Основы программирования

ОСНОВЫ ООП

Класс «Здание»

Класс «Здание»

Характеристики (свойства):

- кол-во этажей;
- общая площадь;
- кол-во людей.

Действия:

- вычисление площади, приходящейся на одного человека;
- вычисление максимального количества человек, которые могут находиться в здании.

Класс, поля класса

```
class Building    {  
    public int floors;  
    public int area;  
    public int person;  
}
```

Создание экземпляра

```
class Program    {  
  
    static void Main(string[] args)    {  
  
        Building house1 = new Building();  
        int areaPerPerson;  
        house1.floors = 4;  
        house1.area = 2500;  
        house1.person = 4;  
    }  
}
```

Работа с экземпляром класса

```
areaPerPerson = house1.area/house1.person;
```

```
Console.WriteLine("Дом общей площадью {0}  
имеет {1} этажа и {2} жильцов", house1.area,  
house1.floors, house1.person);
```

```
Console.WriteLine("На одного человека  
приходится {0} кв.м", areaPerPerson);
```

```
Console.ReadLine();  
}
```

```
}
```

Метод без параметров

```
class Building      {  
  
    public int floors;  
    public int area;  
    public int occupants;  
  
    public int AreaPerPerson() {  
        return area / person;  
    }  
}
```

Создание двух экземпляров

```
class Program {  
  
    static void Main(string[] args) {  
  
        Building house = new Building();  
        Building office = new Building();  
        int areaPP;  
  
        ...  
    }  
}
```

Вызов метода без параметров

```
Console.WriteLine("Дом общей площадью {0}  
имеет {1} этажа и {2} жильцов", house.area,  
house.floors, house.person);
```

```
areaPP = house.areaPerPerson();
```

```
Console.WriteLine("На одного человека  
приходится {0} кв. футов", areaPP);
```

```
}
```

```
}
```


Метод с параметром

```
class Building {  
  
    public int floors;  
    public int area;  
    public int person;  
  
    public int AreaPerPerson() {  
        return area / person;  
    }  
  
    public int MaxPeople(int minAreaPP) {  
        return area / minArea;  
    }  
}
```

Вызов метода с параметрами

...

```
Console.WriteLine("Максимальное кол-во человек  
{0}, если на одного человека должно приходиться  
{1} кв.м", house.MaxPeople(100), "100");
```

```
Console.WriteLine("Максимальное кол-во человек  
{0}, если на одного человека должно приходиться  
{1} кв.м", office.MaxPeople(50), "50");
```

...

Конструктор с параметрами

```
class Building      {  
    public int floors;  
    public int area;  
    public int person;  
  
    public Building(int f, int a, int p)    {  
        floors = f;  
        area = a;  
        person = p;  
    }  
  
    public int AreaPerPerson() {...}  
    public int MaxPeople (int minAreaPP) {...}
```

Вызов конструктор с параметрами

...

```
Building house = new Building(2, 2500, 4);
```

```
Building office = new Building(3, 4200, 25);
```

...

Конструктор без параметров

```
public Building() {  
    floors = 1;  
    area = 1;  
    person = 1;  
}
```

```
public Building(int f, int a, int o) {  
    floors = f;  
    area = a;  
    person = o;  
}
```

```
/* public Building()  
    { } */
```

Задание «Прямоугольник»

Написать класс `Rect`. Проверить работу в классе `Program`.

Поля класса:

- высота;
- ширина.

Методы класса:

- вычисление периметра;
- вычисление площади;
- вывода на консоль информации о прямоугольнике;
- уменьшения прямоугольника в N раз.

Конструкторы класса:

- без параметров;
- с параметрами.

Задание. Решение

```
class Rect
{
    public int width;
    public int height;

    public Rect(int w, int h)
    {
        width = w;
        height = h;
    }

    public int Area()
    {
        return width * height;
    }

    public int Square()
    {
        return 2 * (width + height);
    }

    public void ToConsole()
    {
        Console.WriteLine("Прямоугольник {0}x{1}, S = {2}, P = {3}", width, height,
Area(), Square());
    }

    public void Des(int s)
    {
        width = width / s;
        height = height / s;
    }
}
```

...

```
public Rect (int w, int h)      {  
    width = w;  
    height = h;  
}
```

```
public Rect (int width, int height)  {  
    this.width = width;  
    this.height = height;  
}
```


Задание. Решение

```
class Program
{
    static void Main(string[] args)
    {
        Rect r1 = new Rect(4, 8);
        Rect r2 = new Rect(12, 24);
        Console.WriteLine("Площадь прямоугольника r1: " + r1.Area());
        Console.WriteLine("Площадь прямоугольника r2: " + r2.Area());
        r1.Des(2);
        r1.ToConsole();
        r2.Des(3);
        r2.ToConsole();

        Console.ReadLine();
    }
}
```