

Объектіге-бағытталған программалау

Класс мысалы

```
class имя_класса {  
    // Объявление переменных экземпляра.  
    доступ тип переменная1;  
    доступ тип переменная2;  
    //...  
    доступ тип переменнаяN;  
  
    // Объявление методов.  
    доступ возвращаемый_тип метод1 (параметры) {  
        // тело метода  
    }  
    доступ возвращаемый_тип метод2 (параметры) {  
        // тело метода  
    }  
  
    // ...  
    доступ возвращаемый_тип методы (параметры) {  
        // тело метода  
    }  
}
```

Класс экземпляры

- доступ тип имя_переменной;

Класс

- `class Building {`
- `public int Floors; // количество этажей`
- `public int Area; // общая площадь здания`
- `public int Occupants; // количество жильцов`
- `}`

Объект

- `Building house = new Building();`
- `//` создать объект типа `Building`

Оператор-нүкте

- объект.член
- `house.Floors = 2;`

Мысал

```
// Программа, в которой используется класс Building.

using System;

class Building {
    public int Floors; // количество этажей
    public int Area; // общая площадь здания
    public int Occupants; // количество жильцов
}

// В этом классе объявляется объект типа Building.
class BuildingDemo {
    static void Main() {
        Building house = new Building(); // создать объект типа Building
        int areaPP; // площадь на одного человека

        // Присвоить значения полям в объекте house.
        house.Occupants = 4;
        house.Area = 2500;
        house.Floors = 2;

        // Вычислить площадь на одного человека.
        areaPP = house.Area / house.Occupants;
    }
}
```

Мысал

```
Console.WriteLine("Дом имеет:\n " +  
    house.Floors + " этажа\n " +  
    house.Occupants + " жильца\n " +  
    house.Area +  
    " кв. футов общей площади, из них\n " +  
    areaPP + " приходится на одного человека");
```

```
}  
}
```


Нәтиже

Дом имеет:

2 этажа

4 жильца

2500 кв. футов общей площади, из них

625 приходится на одного человека

Мысал

house	→	<table border="1"><tr><td>Floors</td><td>2</td></tr><tr><td>Area</td><td>2500</td></tr><tr><td>Occupants</td><td>4</td></tr></table>	Floors	2	Area	2500	Occupants	4
Floors	2							
Area	2500							
Occupants	4							
office	→	<table border="1"><tr><td>Floors</td><td>3</td></tr><tr><td>Area</td><td>4200</td></tr><tr><td>Occupants</td><td>25</td></tr></table>	Floors	3	Area	4200	Occupants	25
Floors	3							
Area	4200							
Occupants	25							

Объектке мысал

```
// В этом классе объявляются два объекта типа Building.
class BuildingDemo {
    static void Main() {
        Building house = new Building();
        Building office = new Building();

        int areaPP; // площадь на одного человека

        // Присвоить значения полям в объекте house.
        house.Occupants = 4;
        house.Area = 2500;
        house.Floors = 2;

        // Присвоить значения полям в объекте office.
        office.Occupants = 25;
        office.Area = 4200;
        office.Floors = 3;
    }
}
```

Әдіс(метод)

- // Добавить метод в класс Building.
- using System;
- class Building {
- public int Floors; // количество этажей
- public int Area; // общая площадь здания
- public int Occupants; // количество жильцов
- // Вывести площадь на одного человека,
- public void AreaPerPerson()
- { Console.WriteLine(" " + Area / Occupants + " приходится на одного человека"); } }
- // Использовать метод AreaPerPerson().
- class BuildingDemo {
- static void Main() {
- Building house = new Building();
- Building office = new Building();
- // Присвоить значения полям в объекте house.
- house.Occupants = 4;
- house.Area = 2500;
- house.Floors = 2;
- // Присвоить значения полям в объекте office.
- office.Occupants = 25;

Жалғасы

- office.Area = 4200;
- office.Floors = 3;
- Console.WriteLine("Дом имеет:\n " + house.Floors + " этажа\n " + house.Occupants + " жильца\n " + house.Area + " кв. футов общей площади, из них");
house.AreaPerPerson();
- Console.WriteLine();
- Console.WriteLine("Учреждение имеет:\n " + office.Floors + " этажа\n " + office.Occupants + " работников\n " + office.Area + " кв. футов общей площади, из них");
office.AreaPerPerson(); } }