

Статическая переменная - это общая переменная для всех экземпляров класса, которая хранится в объекте.

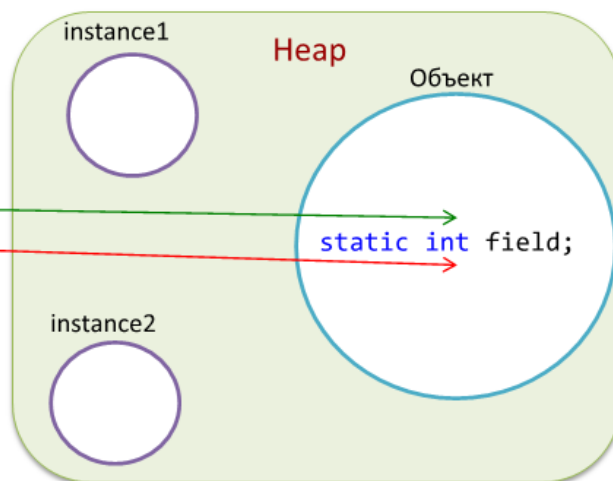
Объекты содержат в себе статические поля и методы.

```
static void Main()
```

```
{  
    MyClass instance1 = new MyClass();  
    MyClass instance2 = new MyClass();  
  
    MyClass.field = 2;  
    MyClass.field = 5;  
}
```

```
class MyClass
```

```
{  
    public static int field;  
}
```



Константы

Константа не может быть объявлена как **static**, поскольку по своему поведению, уже является статической

Поле **const** относится к типу, а не к экземплярам типа. По этому к полям **const** можно обращаться с использованием той же нотации **ИмяКласса.ИмяЧлена**, что и в используемой для статических полей

Static method and field

Статическими могут быть методы и свойства:

```
static void Main()  
{  
    Console.WriteLine("static Method");  
}
```

```
public static int Property  
{  
    get { return field; }  
    set { field = value; }  
}
```

Статические методы не могут обращаться к не статическим полям.

Статические члены не могут быть виртуальными, переопределенными и абстрактными.

Static classes

Статический класс— это контейнер, который содержит в себе только статические члены.

Доступ к членам статического класса осуществляется на **Классе-Объекте**

```
static class StaticClass
{
    public static int item;
}
```

```
static void Main()
{
    StaticClass.item = 10;
}
```

Если класс содержит статические поля, должен быть предоставлен статический конструктор, который инициализирует эти поля при загрузке класса.

Static Constructors

Классы и статические классы могут иметь статические конструкторы.

Статический конструктор всегда отрабатывает первым

```
class MyClass
{
    public static int field;

    static MyClass()
    {
        field = 10;
    }
}
```

Свойства статического конструктора

- Статический конструктор не имеет модификаторов доступа и не принимает параметров.
- Статический конструктор вызывается автоматически для инициализации класса перед созданием первого экземпляра или ссылкой на какие-либо статические члены.
- Статический конструктор нельзя вызывать напрямую.
- Пользователь не управляет тем, когда статический конструктор выполняется в программе.
- Типичным использованием статических конструкторов является случай, когда класс использует файл журнала и конструктор применяется для добавления записей в этот файл.

Extension Methods

Методы расширения позволяют "**добавлять**" методы в существующие типы без создания нового производного типа, перекомпиляции или иного изменения исходного типа

Расширяющие методы могут быть только статическими и создаваться только в статических классах

```
static class MyClass
{
    public static void Method(this string value)
    {
        Console.WriteLine(value);
    }
}
```

```
static void Main()
{
    string text = "Тестовая строка";

    MyClass.Method(text);
    text.Method();
}
```

Аргумент расширения всегда должен быть только один и стоять первым в списке аргументов

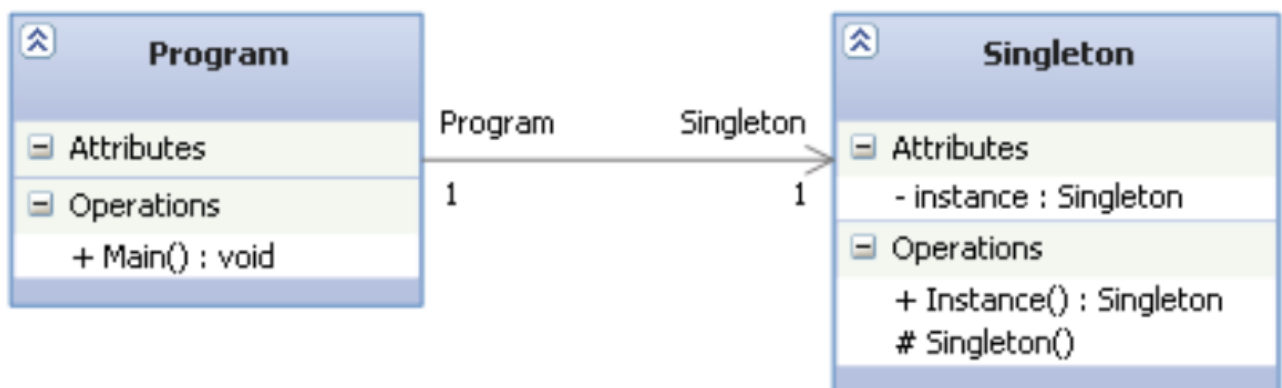
Nested Classes

Тип, определенный внутри класса или структуры, называется вложенным типом

Экземпляр внутреннего класса не может существовать без привязки к включающему его классу верхнего уровня

Статические классы могут в себе содержать не статические **Nested** классы.

Паттерн Singleton



Паттерн Singleton гарантирует, что у класса есть только один экземпляр, и предоставляет к нему глобальную точку доступа