

Лабораторная работа 1.5

«Интерфейсы»

Целью этой лабораторной является знакомство с концепцией интерфейсов на примере языке C#.

Интерфейс – это контракт класса, всё что он содержит это объявление методов. В свою очередь класс, реализующий этот интерфейс, обязан реализовать все его методы.

В отличие от абстрактных классов, интерфейсы не позволяют реализовывать внутри себя какой-либо базовый функционал. Другим отличием становится возможность реализовывать множество интерфейсов. Поскольку в языке C# не поддерживается множественное наследование, то наследоваться (реализовывать некий интерфейс, определенные через абстрактные метода) можно только от одного абстрактного класса. У классов, реализующих интерфейсы, нет ограничений на их количество.

Для демонстрации работы интерфейсов, определим два интерфейса Interface1 и Interface2, которые предлагают к реализации Method1 и Method2 соответственно:

```
interface Interface1
{
    void Method1();
}

interface Interface2
{
    void Method2();
}
```

Теперь определим класс Derived, реализующий оба эти интерфейса:

```
class Derived : Base, Interface1, Interface2
{
    public void Method1()
    {
        Console.WriteLine("Derived.Interface1.Method1");
    }

    public void Method2()
    {
        Console.WriteLine("Derived.Interface2.Method2");
    }
}
```

Обратим внимание, что, когда класс реализует интерфейсы, это не лишает его возможности наследование. Например, здесь мы, помимо реализации Interface1 и Interface2, ещё дополнительно наследуемся от класса Base, реализующим метод Method.

Проверим что всё работает:

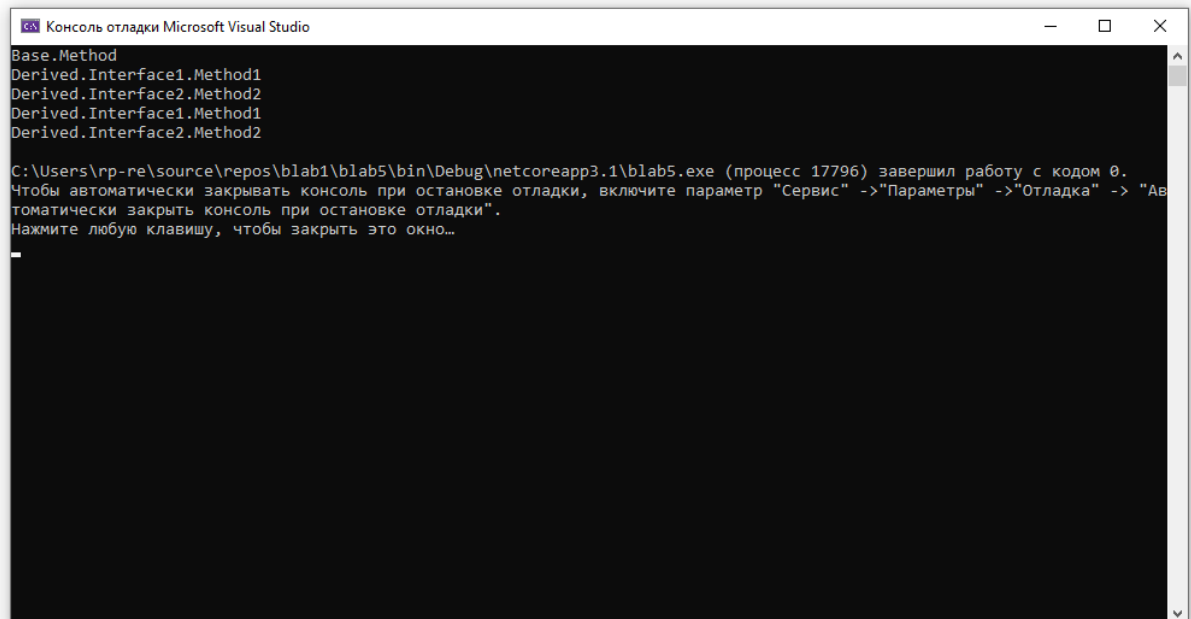
```
Derived dInstance = new Derived();
dInstance.Method();
dInstance.Method1();
```

```
dInstance.Method2();

Interface1 instance1 = dInstance;
instance1.Method1();

Interface2 instance2 = dInstance;
instance2.Method2();
```

Скриншот работы программы:



Приложение

1. Исходный код программы: <https://github.com/proxodilka/csharp-labs/tree/master/blab5>