

Как работает модификатор видимости public?

Чтобы более понятно, как работает модификатор public, берем модификатор private, protected, internal и сравниваем их.

Создаем класс и в классе сделаем несколько переменных.

```
public class State
{
    // все равно, что private int defaultVar;
    int defaultVar;
    // поле доступно только из текущего класса
    private int privateVar;
    // доступно из текущего класса и производных классов, которые определены в этом же
    // проекте
    protected private int protectedPrivateVar;
    // доступно из текущего класса и производных классов
    protected int protectedVar;
    // доступно в любом месте текущего проекта
    internal int internalVar;
    // доступно в любом месте текущего проекта и из классов-наследников в других проектах
    protected internal int protectedInternalVar;
    // доступно в любом месте программы, а также для других программ и сборок
    public int publicVar;

    // по умолчанию имеет модификатор private
    void defaultMethod() => Console.WriteLine($"defaultVar = {defaultVar}");

    // метод доступен только из текущего класса
    private void privateMethod() => Console.WriteLine($"privateVar = {privateVar}");

    // доступен из текущего класса и производных классов, которые определены в этом же
    // проекте
    protected private void protectedPrivateMethod() =>
    Console.WriteLine($"protectedPrivateVar = {protectedPrivateVar}");

    // доступен из текущего класса и производных классов
    protected void protectedMethod() => Console.WriteLine($"protectedVar =
{protectedVar}");

    // доступен в любом месте текущего проекта
    internal void internalMethod() => Console.WriteLine($"internalVar = {internalVar}");

    // доступен в любом месте текущего проекта и из классов-наследников в других проектах
    protected internal void protectedInternalMethod() =>
    Console.WriteLine($"protectedInternalVar = {protectedInternalVar}");

    // доступен в любом месте программы, а также для других программ и сборок
    public void publicMethod() => Console.WriteLine($"publicVar = {publicVar}");
}
```

Поскольку класс State объявлен публичным, он будет доступен из любой программы, а также из других программ и сборок. Класс State имеет пять полей для каждого уровня

доступа. Плюс одна переменная без модификатора, которая по умолчанию является закрытой.

Также есть шесть методов, которые будут отображать значения полей класса на экране. Обратите внимание: поскольку все модификаторы позволяют использовать членов класса внутри этого класса, то есть все переменные класса, включая частные, доступны для всех его методов, поскольку все они находятся в контексте класса State.

Теперь давайте посмотрим, как мы можем использовать переменные нашего класса в программе (то есть в методе Main класса Program), если классы State и Program находятся в одном проекте:

```
class Program
{
    static void Main(string[] args)
    {
        State state1 = new State();

        state1.defaultMethod(); //Ошибка, получить доступ нельзя
        state1.privateMethod(); // Ошибка, получить доступ нельзя
        state1.protectedPrivateMethod(); // Ошибка, получить доступ нельзя
        state1.protectedMethod(); // Ошибка, получить доступ нельзя
        state1.internalMethod();    // норм
        state1.protectedInternalMethod(); // норм
        state1.publicMethod();      // норм
    }
}
```

Здесь нам было доступно всего три метода: internalMethod, protectedInternalMethod, publicMethod, которые имеют модификаторы internal, protected internal и public соответственно.

Ответ: вот так работает модификатор public. Он открыт и доступен всем в отличие от других модификаторов.