**Инкапсуляция**

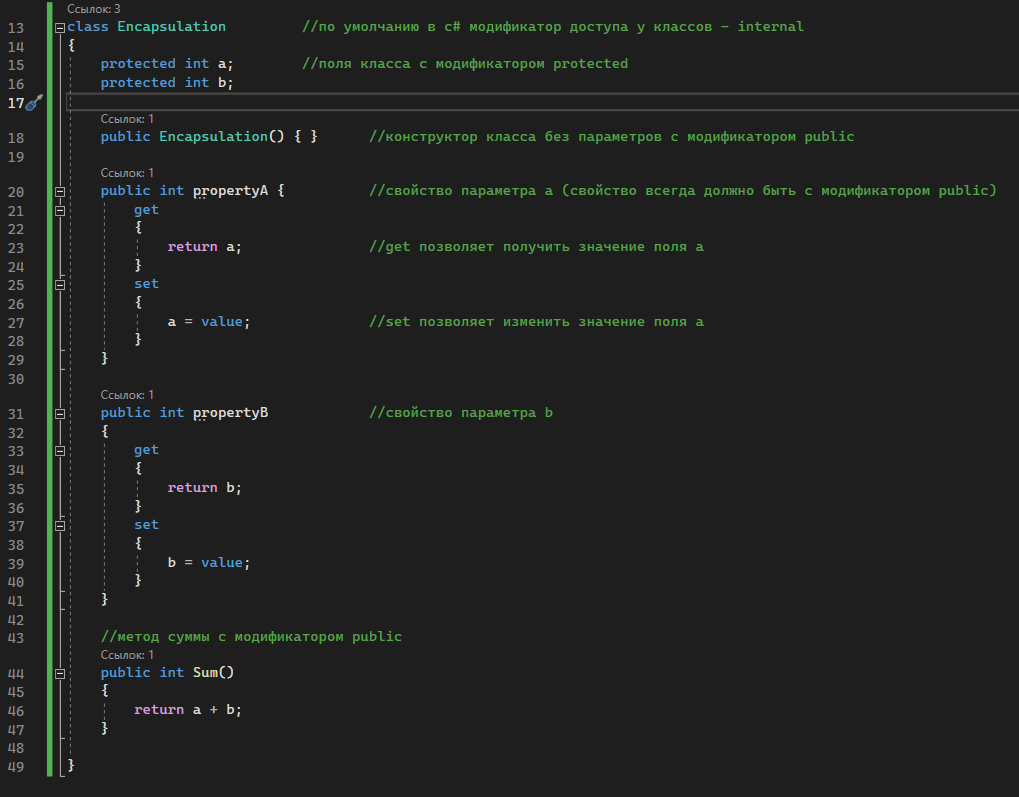
**Инкапсуляция – это механизм программирования , объединяющий вместе код и данные , которыми он манипулирует, исключая как вмешательство извне , так и неправильное использование данных. В объектно-ориентированном языке данные и могут быть объединены в совершенно автономный чёрный ящик. Внутри такого ящика находятся все необходимые данные и код.**

**Т.е инкапсуляция представляет собой способности языка скрывать излишние детали реализации от пользователя объекта.**

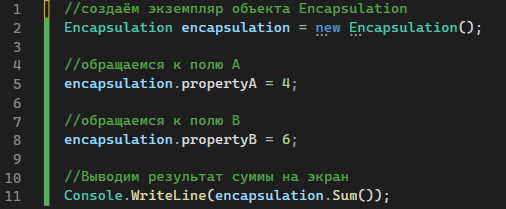
**Сигнатура инкапсуляции складывается из следующих аспектов :**

* **Имя метода.**
* **Кол-во параметров.**
* **Типы параметров.**
* **Порядок параметров.**
* **Модификаторы параметров.**

**Пример класса с использованием инкапсуляции**

****

**В методе main необходимо написать следующий код**

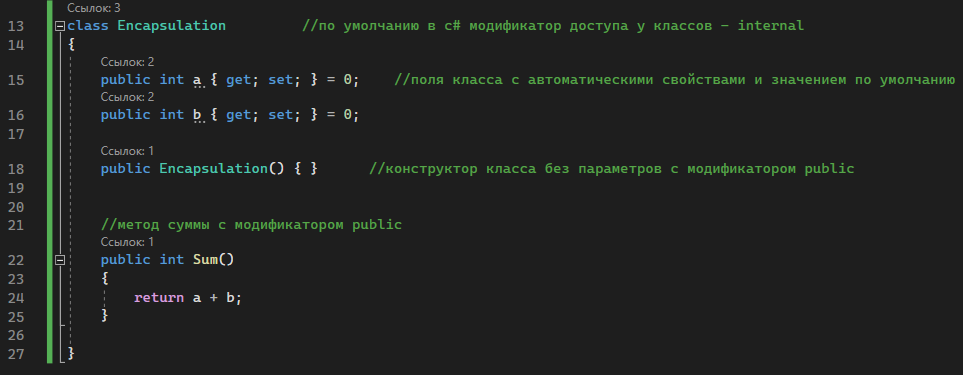
****

**В этой простенькой программе мы создаём класс Encapsulation, используя базовые правила соблюдения инкапсуляции**

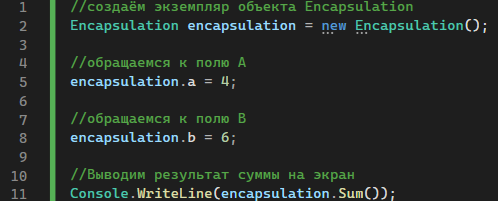
**Основным моментом в программе являются свойства . Именно благодаря им мы можем получить и изменить значения полей (a, b) с модификатором protected**

**Автоматические свойства**

**Свойства служат для управления доступа к полям. Однако если у нас возникает потребность в бесчисленном множестве полей , то писать для каждого поля своё свойство было-бы очень утомительно. Поэтому в С# добавлены автоматические свойства . Они имею сокращённое объявление**

****

**Метод main тоже немного изменился**

****

**Главное преимущество автоматических свойств в том , что их можно развернуть в любом месте в коде и это займёт минимум усилий.**

**Заключение:**

**Всегда используйте инкапсуляцию при разработке вашего класса! Несложные принципы инкапсуляции уберегут ваше приложение от ошибок связанных с намеренным или не намеренным внесением, непредвиденных разработчиком, корректировок в логике программы.**