

Делегат является типом методов, экземпляром делегата является метод, соответствующий делегату.

Пример объявления делегата:

```
delegate int PlusOrMinus(int p1, int p2);
```

Пример объявления метода с делегатным параметром:

```
static void PlusOrMinusMethod(  
    string str,  
    int i1,  
    int i2,  
    PlusOrMinus PlusOrMinusParam)  
{  
    int Result = PlusOrMinusParam(i1, i2);  
    Console.WriteLine(str + Result.ToString());  
}  
)
```

Лямбда-выражения или лямбда-функции пришли в язык С# из функционального программирования. Лямбда-выражение обладает свойством сигнатурности, но у него нет свойства именованности. То есть у лямбда-выражения есть набор входных параметров и тип возвращаемого значения, но нет имени. Оно записывается в том месте, где должно вызываться.

Пример объявления лямбда-выражения:

```
PlusOrMinus pm4 = (int x, int y) =>  
{  
    int z = x + y;  
    return z;  
};
```

Пример передачи лямбда-выражения в качестве параметра:

```
PlusOrMinusMethod(  
    "Создание экземпляра делегата на основе лямбда-выражения 1: ",  
    i1,  
    i2,  
    (int x, int y) =>  
    {  
        int z = x + y;  
        return z;  
    }  
);
```

Если в параметр делегатного типа передать лямбда-выражение с меньшим количеством аргументов, чем указано в делегате, то при этом компилятор выдаст в строке определения лямбда-выражения следующую ошибку: «Делегат "Delegates.PlusOrMinus" не принимает аргументы 1».