**Выполнили студент 215 группы**

**Марков К.А.**

**Практическое занятие 14**

**Тема: «Применение делегатов»**

**Цель работы:** изучить возможности применения делегатов в языке C#.

**Задание:**

1. Создайте приложение C# в среде MS Visual Studio.

2. В соответствии с вариантом индивидуального задания реализуйте пользовательский тип делегата требуемой сигнатуры и выполните с его

Вариант 11: Action<Func<int>, char>

internal class Program

{

//Этот код создает пользовательский делегат MyDelegate с сигнатурой Action<Func<int>, char>,

//а затем создает экземпляр делегата, который вызывает переданный метод типа Func<int>

//и выводит его результат вместе с переданным символом.

// Определение пользовательского делегата

delegate void MyDelegate(Func<int> func, char c);

static void Main(string[] args)

{

// Создание экземпляра делегата с помощью лямбда-выражения

MyDelegate myDelegate = (func, c) =>

{

// Вызов переданной функции и передача ей результата в метод Console.WriteLine()

Console.WriteLine($"Result: {func()}, Char: {c}");

};

// Вызов метода, передаваемого в делегат, и символа 'A'

myDelegate(() => 42, 'A');

// Вызов другого метода и символа 'B'

myDelegate(() => 10 \* 5, 'B');

Console.ReadKey();

}

}

**Вопросы:**

1. Что такое тип делегата?

- Тип делегата в C# представляет собой ссылочный тип данных, который используется для определения методов, которые можно вызывать во время выполнения программы. Делегаты позволяют передавать методы как аргументы других методов, хранить ссылки на методы в коллекциях и т. д. Они служат для создания гибких и расширяемых систем обратного вызова.

1. Основные направления использования делегатов:

- События и обработчики событий: Делегаты используются для создания и привязки обработчиков событий.

- Потоки и параллельное программирование: Делегаты могут быть использованы для запуска методов в отдельных потоках или параллельно.

- Реализация шаблона обратного вызова: Делегаты позволяют реализовывать шаблон обратного вызова, позволяя классам вызывать методы, определенные в других классах или модулях.

- Использование в LINQ: В LINQ (Language Integrated Query) делегаты используются для передачи методов в методы расширения для выполнения запросов к данным.

1. Для чего предназначен тип Action<T>? Чем он отличается от Func<T>?

- `Action<T>` представляет делегат, который не возвращает результат (возвращает `void`) и принимает один или несколько параметров. Он обычно используется для представления методов, которые выполняют какие-то действия, но не возвращают результат.

- `Func<T>` представляет делегат, который возвращает результат определенного типа и может принимать ноль или более параметров. Он используется для представления методов, которые принимают аргументы и возвращают результат.

4. Чем пользовательские делегаты отличаются от библиотечных?

- Пользовательские делегаты создаются пользователем для определения собственных типов делегатов, а библиотечные делегаты представлены в стандартной библиотеке .NET Framework и предназначены для решения общих задач, таких как обработка событий, многопоточность и другие.

- Пользовательские делегаты могут быть настроены на определенные требования пользовательского приложения, в то время как библиотечные делегаты предоставляются для общего использования в различных сценариях.

**Вывод:** изучил возможности применения делегатов в языке C#.