

Forum:[Hack Forums](#)

Thread: [Comprehensive Lambda Tutorial](#)

SIM-1725686 - September 26, 2015, 12:21 pm

Hi, HF. I wrote a tutorial on lambda functions today, and I thought it'd be a great idea to share it with you all.

<http://blog.thosakwe.com/2015/csharp-lambdas/>

Feel free to leave me a comment, or share it with your friends. Thanks for reading!

StayTech666 - September 27, 2015, 8:13 pm

Thanks for the tutorial! I've been meaning to learn lambdas for some time now and this definitely helps. Short, sweet and simple.

Could you help me understand this line (where firstHalf and secondHalf are int arrays):
Code var result = secondHalf.Zip(firstHalf, (f, s) => new[] { f, s }).SelectMany(f => f);

StayTech666 - September 28, 2015, 12:43 pm

(Sep 28, 2015 01:15 AM)xadamxk Wrote: [\[CITE\]](#) From what I understand, it's similar to the function operator?

Can you give a little more to exclaim it, like a basic example?

I've seen a few of them in samples of code, but I had no idea what it was so I didn't use it. One of his examples is pretty easy to follow:

Code public void Main()

```
{  
    var foo = 1337;  
    Func<int, int> f = x => x + 3;  
}
```

So, in that code he is declaring a function f that takes a variable x. The declaration of Func<int, int> is defining the input and output types (int for an input, int for an output) and the lambda expression is basically defining the actual return value, so it is simply x + 3. (Using the x we declared earlier.)

Now, you can easily call this function by simply writing:

Code public int AddThree(int valueToAdd)

```
{  
    Func<int, int> f = x => x + 3;  
  
    return f(valueToAdd); // if valueToAdd is 7, then this returns 10  
}
```

Here is another quick example:

Code static void Main(string[] args)

```
{  
    Func<String, String> Wrap = (word => "\"" + word + "\"");  
  
    String message = "Hello world!";  
    Console.WriteLine(Wrap(message));  
}
```

```
    Console.Read();  
}
```

So all this does is take a string and put quotations around it.
(Hello world! becomes "Hello world!")

Lambdas are awesome! ?

SIM-1725686 - September 28, 2015, 3:03 pm

(Sep 27, 2015 08:13 PM)StayTech666 Wrote: [\[CITE\]](#) Thanks for the tutorial! I've been meaning to learn lambdas for some time now and this definitely helps. Short, sweet and simple.

Could you help me understand this line (where firstHalf and secondHalf are int arrays):

Code var result = secondHalf.Zip(firstHalf, (f, s) => new[] { f, s }).SelectMany(f => f);

Thanks! I put effort into it to make it easy to read. As for the code, it does two things:

Takes f and s (whatever they are), and creates an array containing them

The predicate for SelectMany is redundant. It returns the variable passed to it.

(Sep 28, 2015 12:46 PM)xadamxk Wrote: [\[CITE\]](#) I suppose I understand it. I just don't know why this would be used before the conventional methods.

It's convenient to use this with LINQ. Also, an advantage of lambdas is that you can see what you're doing, rather than having to scroll to other code that may be anywhere else. Even 5000 lines deep in another subfolder. It makes it easy to understand the code, basically.

typeofBlue - October 6, 2015, 8:11 am

Dont mean to nitpick or anything, but I think it would have been useful to show the other way of writing a anonymous method using the old delegate() {} way, since this might be more intiuitive for people. It helped me understand it better when I first learned it at least. Anyways good post, here's free bump ;)