Where do I use delegates? [closed]

Asked 16 years, 3 months ago Modified 4 months ago

Viewed 50k times

114

Closed. This question is opinion-based. It is not currently accepting answers.

Want to improve this question? Update the question so it can be answered with facts and citations by editing this post.

Closed 11 years ago.

Improve this question

What are some real world places that call for delegates? I'm curious what situations or patterns are present where this method is the best solution. No code required.

oop design-patterns delegates

Share

Improve this question

Follow

edited Aug 14 at 16:25

asked Aug 28, 2008 at 1:58

slipsec

3,052 ● 3 ● 34 ● 46

8 Answers

Sorted by: Highest score (default)

164

As stated in "Learning C# 3.0: Master the fundamentals of C# 3.0"

General Scenario: When a head of state dies, the President of the United States typically does not have time to attend the funeral personally. Instead, he dispatches a delegate. Often this delegate is the Vice President, but sometimes the VP is unavailable and the President must send someone else, such as the Secretary of State or even the First Lady. He does not want to "hardwire" his delegated authority to a single person; he might delegate this responsibility to anyone who is able to execute the correct international protocol.

The President defines in advance what responsibility will be delegated (attend the funeral), what parameters will be passed (condolences, kind words), and what value he hopes to get back (good will). He then assigns a particular person to that delegated responsibility at "runtime" as the course of his presidency progresses.

> **In programming Scenario:** You are often faced with situations where you need to execute a particular action, but you don't know in advance which method, or even which object, you'll want to call upon to execute it.
>
> **For Example:** A button might not know which object or objects need to be notified. Rather than wiring the button to a particular object, you will connect the button to a delegate and then resolve that delegate to a particular method when the program executes.

Share Improve this answer

Follow

edited Jun 20, 2020 at 9:12

Community Bot

1 ● 1

answered Jun 26, 2013 at 10:43

Atish Kumar Dipongkor

10.4k ● 9 ● 51 ● 77

1 Why not simply use anonymous functions / lambdas then? – Pacerier Nov 5, 2014 at 23:19

6 @Pacerier This is a late response and you may have discovered the answer, but for the sake of other readers who find this: an anonymous function is a type of delegate. – Anthony Jan 28, 2015 at 20:50

11 @Anthony+Pacerier: An anoymous method is NOT a delegate. An anonymous method is a piece of code. A delegate is a pointer which points to that piece of code. You cannot have a anoymous method without a delegate pointing to it, otherwise it would never be called. A lambda is an

operator, used to construct lambda expressions. Lambda expressions mostly results in anoymous methods which will be pointed to by... a delegate. – Martin Mulder Apr 22, 2015 at 8:36

Silly me, but I would just create an If ElseIf statement if I didn't know the use when the button was pressed. I said silly me. – JustJohn Jun 28, 2016 at 21:23

2 @Pacerier anonymous method introduced in C# 2.0 and Lambda introduced in C# 3.0 if Lambda introduced first,there never would have been anonymous methods and even delegate! – AminM Jun 22, 2017 at 21:18

37

> A delegate is a named type that defines a particular kind of method. Just as a class definition lays out all the members for the given kind of object it defines, the delegate lays out the method signature for the kind of method it defines.

Based on this statement, a delegate is a function pointer and it defines what that function looks like.

A great example for a real world application of a delegate is the Predicate. In the example from the link, you will notice that Array.Find takes the array to search and then a predicate to handle the criteria of what to find. In this case it passes a method ProductGT10 which matches the Predicate signature.

Share Improve this answer edited Aug 28, 2008 at 3:11

Follow

answered Aug 28, 2008 at 2:12

Dale Ragan

18.3k ● 3 ● 55 ● 71

20

One common use of delegates for generic Lists are via Action delegates (or its anonymous equivalent) to create a one-line foreach operation:

```
myList.Foreach( i => i.DoSomething());
```

I also find the Predicate delegate quite useful in searching or pruning a List:

```
myList.FindAll( i => i.Name == "Bob");
myList.RemoveAll( i => i.Name == "Bob");
```

I know you said no code required, but I find it easier to express its usefulness via code. :)

Share Improve this answer

Follow

answered Aug 28, 2008 at 2:32

Jon Limjap

95.3k ● 15 ● 103 ● 153

13

Binding Events to Event Handlers is usually your first introduction to delegates...You might not even know you were using them because the delegate is wrapped up in the EventHandler class.

Share Improve this answer

Follow

answered Aug 28, 2008 at 2:00

FlySwat

175k ● 75 ● 248 ● 314

6

I had the same question as you and went to this site for an answer.

Apparently, I didn't understood it better even though I skimmed through the examples on this thread.

I found a great use for delegates now that I read: http://www.c-sharpcorner.com/UploadFile/thiagu304/passdata05172006234318PM/passdata.aspx

This might seem more obvious for new users because Forms is much more complicated to pass values than ASP.NET websites with POST/GET (QueryString) ..

Basically you define a delegate which takes **"TextBox *text*"** as parameters.

// Form1

```
// Class Property Definition
public delegate void delPassData(TextBox text);


// Click Handler
private void btnSend_Click(object sender,
System.EventArgs e)
{
```

```
        Form2 frm= new Form2();
        delPassData del=new delPassData(frm.funData);
        del(this.textBox1);
        frm.Show();
}
```

// SUMMARY: Define delegate, instantiate new Form2 class, assign funData() function to delegate, pass in your textBox to the delegate. Show the form.

// Form2

```
public void passData(TextBox txtForm1)
{

        label1.Text = txtForm1.Text;
}
```

// SUMMARY: Simply take TextBox txtForm1 as parameters (as defined in your delegate) and assign label text to textBox's text.

I hope this enlightens some use on delegates :) ..

Share Improve this answer

Follow

answered May 10, 2010 at 8:42

dza

1,504 ● 2 ● 13 ● 25

Either the Form2 class method needs to be named funData (instead of passData), or Form1 function assignment needs to be named frm.passData (instead of frm.funData). – ethan Feb 5, 2018 at 17:16

5

If you're interested in seeing how the Delegate pattern is used in real-world code, look no further than Cocoa on Mac OS X. Cocoa is Apple's preferred UI toolkit for programming under Mac OS X, and is coded in Objective C. It's designed so that each UI component is intended to be extended via delegation rather than subclassing or other means.

For more information, I recommend checking out what Apple has to say about delegates here.

Share Improve this answer

Follow

answered Aug 28, 2008 at 2:30

spate

386 ● 2 ● 5

5

I had a project which used win32 Python.

Due to various reasons, some modules used odbc.py to access the DB, and other modules - pyodbc.py.

There was a problem when a function needed to be used by both kinds of modules. It had an connection object passed to it as an argument, but then it had to know whether to use dbi.dbiDate or datetime to represent times.

This was because odbc.py expected, as values in SQL statements, dates as dbi.dbiDate whereas pyodbc.py expected datetime values.

One further complication was that the connection objects created by odbc.py and pyodbc.py did not allow one to set additional fields.

My solution was to wrap the connection objects returned by odbc.odbc(...) and pyodbc.pyodbc(...) by a delegate class, which contains the desired time representation function as the value of an extra field, and which delegates all other field requests to the original connection object.

Share Improve this answer

Follow

answered Sep 16, 2008 at 14:39

Omer Zak

1,157 ● 2 ● 8 ● 25

2

A quick google search came up with this http://en.wikipedia.org/wiki/Delegation_pattern . Basically, anytime that you use an object that forwards it's calls to another object then you are delegating.

Share Improve this answer

Follow

answered Aug 28, 2008 at 2:03

martinatime

2,487 ● 1 ● 17 ● 23