Events (C# Programming Guide)

Visual Studio 2013 9 out of 13 rated this helpful

Events enable a class or object to notify other classes or objects when something of interest occurs. The class that sends (or *raises*) the event is called the *publisher* and the classes that receive (or *handle*) the event are called *subscribers*.

In a typical C# Windows Forms or Web application, you subscribe to events raised by controls such as buttons and list boxes. You can use the Visual C# integrated development environment (IDE) to browse the events that a control publishes and select the ones that you want to handle. The IDE automatically adds an empty event handler method and the code to subscribe to the event. For more information, see How to: Subscribe to and Unsubscribe from Events (C# Programming Guide).

Events Overview

Events have the following properties:

- The publisher determines when an event is raised; the subscribers determine what action is taken in response to the event.
- An event can have multiple subscribers. A subscriber can handle multiple events from multiple publishers.
- Events that have no subscribers are never raised.
- Events are typically used to signal user actions such as button clicks or menu selections in graphical user interfaces.
- When an event has multiple subscribers, the event handlers are invoked synchronously when an event is raised. To invoke events asynchronously, see Calling Synchronous Methods Asynchronously.
- In the .NET Framework class library, events are based on the EventHandler delegate and the EventArgs base class.

Related Sections

For more information, see:

- How to: Subscribe to and Unsubscribe from Events (C# Programming Guide)
- How to: Publish Events that Conform to .NET Framework Guidelines (C# Programming Guide)
- How to: Raise Base Class Events in Derived Classes (C# Programming Guide)
- How to: Implement Interface Events (C# Programming Guide)
- Thread Synchronization (C# and Visual Basic)
- How to: Use a Dictionary to Store Event Instances (C# Programming Guide)
- How to: Implement Custom Event Accessors (C# Programming Guide)

C# Language Specification

For more information, see the C# Language Specification. The language specification is the definitive source for C# syntax and usage.

Featured Book Chapters

Delegates, Events, and Lambda Expressions in C# 3.0 Cookbook, Third Edition: More than 250 solutions for C# 3.0 programmers

Delegates and Events in Learning C# 3.0: Master the fundamentals of C# 3.0

See Also

Reference
Delegates (C# Programming Guide)
EventHandler
Concepts
C# Programming Guide
Other Resources
Creating Event Handlers in Windows Forms
Multithreaded Programming with the Event-based Asynchronous Pattern