

## REVIEW LESSON

MTA Course: 10754 Microsoft .NET Fundamentals

Lesson name: Microsoft .NET Fundamentals 1.2

Topic: Understand events and event handling in the .NET Framework (One 50-minute class period)

File name: 10754\_Msft.NET\_RL\_1.2

### Lesson Objective

**1.2:** Understand events and event handling in the .NET Framework. *This objective may include but is not limited to:* understanding the event-driven programming model and event handlers, raising events, and implementing delegates

### Preparation Details

#### Prerequisite student experiences and knowledge:

This MTA Certification Exam Review lesson is written for students who have learned about application programming with the Microsoft® .NET Framework. Students who do not have the prerequisite knowledge and experiences cited in the objective will find additional learning opportunities using resources such as those listed in the “Resources” section at the end of this review lesson.

In particular, students should be comfortable creating a project in Microsoft Visual Studio® and designing a basic form with the Designer. For additional details, see the “Suggested best practices” section below.

#### Instructor preparation activities:

- Make copies of the Student Activity 10754\_Msft.NET\_SA\_1.2.

#### Resources, software, and additional files needed for this lesson:

- 10754\_Msft.NET\_PPT\_1.2
- 10754\_Msft.NET\_SA\_1.2

## **Teaching Guide**

### **Essential Vocabulary**

**event**—an action that occurs in a program that the developer can respond to, or “handle,” in code.

**event-driven programming**—a model in which the flow of program execution is determined by events, such as user interface actions.

**event handler**—a code block (method or procedure) that is invoked when the corresponding event is raised.

**delegate**—objects that refer to methods; used by .NET applications to link events to event handlers.

## **Lesson Sequence**

### **Activating prior knowledge/lesson staging (10 minutes):**

1. As indicated in the Microsoft PowerPoint® presentation, direct students to answer the prompt about events in our daily lives.
2. When students are finished, they should discuss their observations with a partner.
3. If time allows, call on each pair to share their responses.

### **Lesson activity (30 minutes):**

1. Use the presentation to review the basics of event-driven programming in .NET applications.
2. As indicated in the presentation, direct students to complete the student activity. They will create a form with some simple event handlers. Verify their work by observation.

### **Assessment/lesson reflection (10 minutes):**

1. As directed on the “Lesson Review” slide in the slideshow, ask students to explain the basics of event-driven programming.
2. As time allows, call on students to share responses; or, collect their responses and review after class to ensure students adequately understand the concepts.

**Resources:**

- **MSDN®: Raising Events and Responding to Events**  
*<http://msdn.microsoft.com/en-us/library/ms973905.aspx>*
- **MSDN: Events (C# Programming Guide)**  
*<http://msdn.microsoft.com/en-us/library/awbftdfh.aspx>*
- **MSDN: Delegates (Visual Basic)**  
*<http://msdn.microsoft.com/en-us/library/ms172879.aspx>*  
*<http://msdn.microsoft.com/en-us/library/25zf0ze8.aspx>*

**Suggested best practices:**

- Because events are often (though not always) closely tied to the user interface, it may be beneficial to cover the basic use of the Visual Studio Form Designer, as covered at *<http://msdn.microsoft.com/en-us/beginner/bb308740.aspx>* (Microsoft Visual Basic®) and *<http://msdn.microsoft.com/en-us/beginner/bb308738.aspx>* (Microsoft C#).

A user interface created for such an assignment could be used as the basis for adding event handlers in this lesson.