

## REVIEW LESSON

MTA Course: Windows Development Fundamentals

Lesson name: Windows Development Fundamentals 3.1

Topic: Create a Windows Service application (One 50-minute class period)

File name: WinDevFund\_RL\_3.1

### **Lesson Objective:**

**3.1:** Create a Windows Service application. *This objective may include but is not limited to:* inheriting the *ServiceBase* class; writing code in the *Main* method; overriding the *OnStart* and *OnStop* procedures.

### **Preparation Details**

#### **Prerequisite student experiences and knowledge**

This MTA Certification Exam Review lesson is written for students who have learned about Windows® application programming. 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 Microsoft® resources and Web links at the end of this review lesson.

#### **Instructor preparation activities**

- Familiarize yourself with the review questions in the Microsoft PowerPoint® slideshow. Modify or add to the questions to suit your needs if desired.

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

- WinDevFund\_PPT\_3.1

## Teaching Guide

### Essential vocabulary:

**Windows Service application**—a long-running application that does not show a user interface.

**ServiceBase**—the class that must be inherited in order to create a Windows Service application.

**service lifetime**—the length of time that a process is active; during its lifetime, the process may undergo several changes in internal state .

**serviceController component**—used to connect to an installed service and manipulate its state

**override (a method, function, etc.)**—language feature that allows a subclass to have a specific implementation of a method from a superclass.

## Lesson Sequence

### Activating prior knowledge/lesson staging (Anticipatory Set—5 minutes)

1. *Student prompt (as indicated in the slideshow):* “List some differences between an antivirus application (or another Windows Service application you know) and a traditional user interface (UI)–based application, such as Microsoft Word or Windows Internet Explorer®.”
2. Call on students to read their responses. You may choose to list the responses on the board.
  - Responses will vary, of course. Be sure to point out that although a program such as an antivirus application may include a user interface (to change settings and manage threats that are detected), Windows Service applications generally function without a user interface. Other important considerations: UI applications are started and stopped by users as they are needed; Windows Services applications run constantly “in the background”; users may not be aware of the Windows Services applications that are running.

### Lesson activity (25–35 minutes)

1. Question-and-answer session (5 Minutes)
  - Using the PowerPoint slideshow, ask students questions and discuss their responses.
  - See the “Suggested Best Practices” section at the end of this lesson for a few active learning strategies to use with this lesson.
  - Each question slide is followed by a slide with relevant information.

**Assessment/lesson reflection (10–15 minutes)**

1. Use Microsoft Visual Studio® to create an empty service application using the Windows Service template.
2. Look at the generated code in the *Service1* class (and the *Program* class in Microsoft Visual C#®). Use code comments to identify and label the methods reviewed in the presentation.

**Microsoft resources and Web links****MSDN Library: Windows Service Applications**

(<http://msdn.microsoft.com/en-us/library/y817hyb6.aspx>)

**MSDN Library: Introduction to Windows Service Applications**

(<http://msdn.microsoft.com/en-us/library/d56de412.aspx>)

**MSDN Library: Service Application Programming Architecture**

(<http://msdn.microsoft.com/en-us/library/yzk7ksy2.aspx>)

**MSDN Library: How to: Create Windows Services**

(<http://msdn.microsoft.com/en-us/library/9k985bc9.aspx>)

**MSDN Library: Walkthrough: Creating a Windows Service Application in the Component Designer**

(<http://msdn.microsoft.com/en-us/library/zt39148a.aspx>)

**Additional activities (homework or enrichment):**

- If time allows, and especially if your students do not have very much experience creating Windows Service applications, consider having them complete the “Walkthrough” listed in the resources.

**Suggested best practices:**

“Popsicle Stick Responses”—Put each student’s name on a Popsicle stick and place the sticks in a cup. For each question, pull a Popsicle stick to decide who responds to the question. (Notecards could be used in place of Popsicle sticks.) Important tips:

- This strategy ensures that all students think about the questions, rather than just calling on someone right away and letting him or her answer. Likewise, it ensures that the teacher doesn’t inadvertently call on the same student repeatedly.
- Be sure to ask the question and give “wait time” before drawing a Popsicle stick or notecard. This ensures that all students think about the answer—if you call on a student right after you ask the question (or before), the rest of the students know they won’t have to think of an answer.

- Try to avoid letting students off the hook by saying, “I don’t know.” Encourage them to take a guess or state something they know about the topic in general. If you must move on, be sure to return to the student to answer the next question.
- After answering a question, the student’s Popsicle stick or notecard goes back in the cup. This keeps students accountable for each question; otherwise, students may not think of responses after they have been called upon once.

“Think-Pair-Share”—Ask a question, then direct students to take some time to think about a response. Next, have students turn to a partner and share their responses. Finally, call on a few students to respond—they can give their own answer or their partner’s.

Important tips:

- It is crucial to give time for students to think immediately after the question is asked. Be sure students do not blurt out answers without being called upon. It also helps to use a timer to ensure that you’re giving students a chance to think.
- Establish pairs ahead of time so that there is no confusion when students are directed to discuss with their partners.
- You can combine this with the “Popsicle Stick Responses” activity when you ask students to share their responses with the class.

“Think-Pair-Square”—A variation of “Think-Pair-Share,” you ask a question, and students again think of an answer and discuss with a partner. However, instead of reporting to the class, two pairs combine to discuss their answers as a group of four (a “square”). Important tips:

- You can still call on individuals to report answers to the class. You could also walk around the room and listen to student discussions to ensure that they are on the right track.
- Once again, be sure to establish the pairs and “squares” before starting.