

## REVIEW LESSON

MTA Course: Software Development Fundamentals

Lesson name: Software Development Fundamentals 1.3

Topic: Identify the appropriate method for handling repetition (One 50-minute class period)

File name: SoftDevFund\_RL\_1.3

### Lesson Objective

**1.3:** Identify the appropriate method for handling repetition. *This objective may include but is not limited to:* `for` loops, `while` loops, `do...while` loops, and recursion.

### Preparation Details

#### Prerequisite student experiences and knowledge

Students should have had practice evaluating loop structures and stepping through recursive methods using a trace table. This MTA Certification Exam Review lesson is written for students who have learned about 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

None

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

- SoftDevFund\_PPT\_1.3
- SoftDevFund\_SA\_1.3\_1
- SoftDevFund\_SA\_1.3\_1\_key
- SoftDevFund\_SA\_1.3\_1
- SoftDevFund\_SA\_1.3\_2\_key

## **Teaching Guide**

### **Essential Vocabulary:**

**iterate**—to execute one statements or a block of statements repeatedly. Statements so executed are said to be in a loop.

**recursion**—the ability of a routine to call itself. Recursion enables certain algorithms to be implemented with small, simple routines, but it does not guarantee speed or efficiency. Erroneous use of recursion can cause a program to run out of stack space during execution, causing the program—and sometimes the entire system—to crash.

## **Lesson Sequence**

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

1. Introduce the lesson and have students do the Activator in the PowerPoint® presentation. The slide shows students some inefficient code and asks them what is required to perform the same task, but with fewer lines of code.

### **Lesson activity (25 minutes)**

1. Show the PowerPoint presentation.
  - Review the key terms in the lesson and use a show of hands to quickly illustrate which concepts need more review than others.
  - Talk about where we see repetition in real life.
  - Go over the different types of looping structures and their components.
  - Review the situations in which different loops are more appropriate.
  - Take time to compare the different loops to one another

### **Assessment/lesson reflection (15 minutes)**

1. Student Activity (SoftDevFund\_SA\_1.3\_1)
2. Student Activity (SoftDevFund\_SA\_1.3\_2)

### **Microsoft resources and Web links**

Loops (C# Programming Guide)

<http://msdn.microsoft.com/en-us/library/f0e10e56.aspx>

**Suggested best practices:**

- Assign one of the worksheets as homework if more time is needed.

**Additional notes to the instructor:**

- The use of trace tables should be suggested to students who are having trouble with evaluating loops, and especially for students who are having trouble with recursion. Trace tables are tables that are created during run time where each column displays the value of a variable and each row is generated during an iteration.