

Developing ASP.NET MVC Web Applications

Session: 13

Debugging and Monitoring

Objectives

- ◆ Define and describe how to perform debugging
- ◆ Explain how to perform health monitoring of an application
- ◆ Define and describe how to use the Performance Monitoring tool

For Aptech Centre Use Only

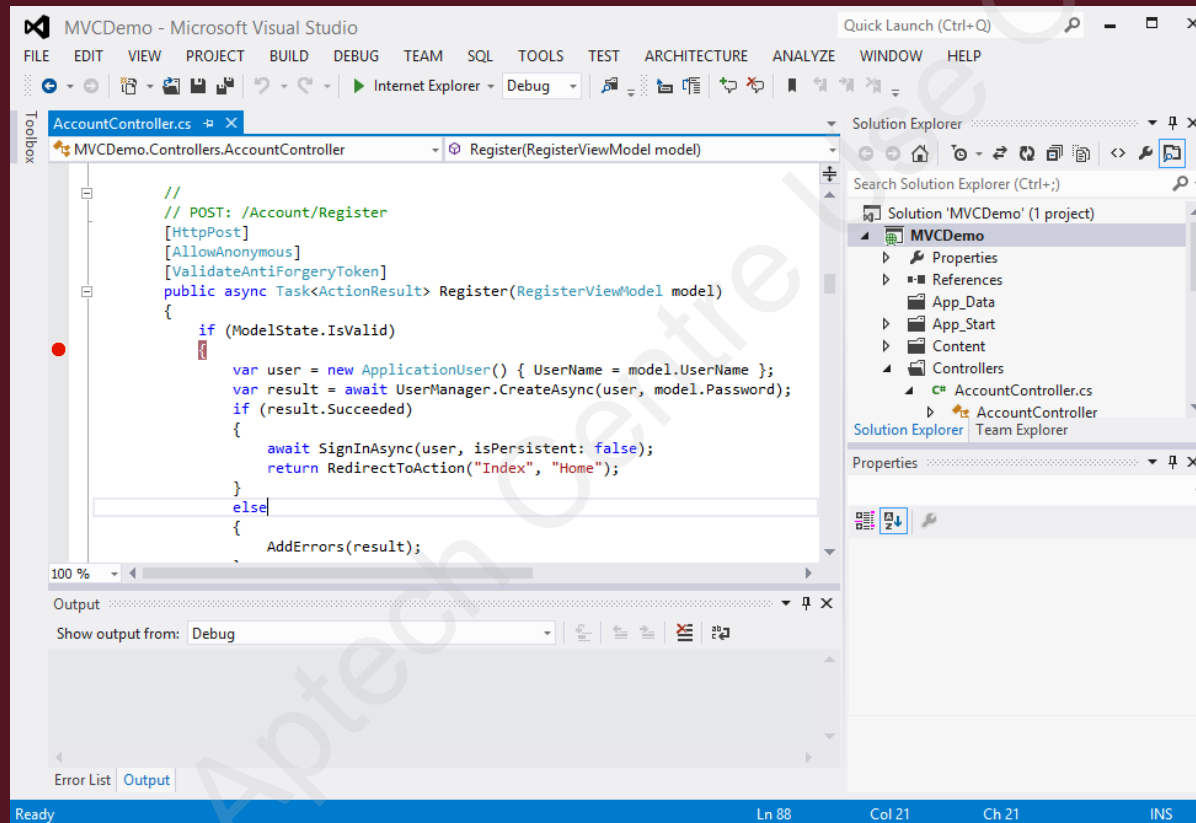
- ◆ While developing an application it may contain syntax errors, logical errors, and run-time errors.
- ◆ The syntax errors are resolved when you compile the application in Visual Studio 2013.
- ◆ However other errors, such as logical and run-time errors cannot be identified while compiling the application.
- ◆ To identify such errors you need to debug the code while it is running.
- ◆ Debugging is a technique that enables you to resolve such errors present in the application.
- ◆ To perform debugging in your application, you can use various techniques that Visual Studio 2013 provides.

- ◆ There are various tools available that you can use to debug your application.
- ◆ One such tool is Visual Studio debugger.
- ◆ The Visual Studio debugger:
 - ◆ Enables you to run your code line by line, so that you can monitor the program execution properly.
 - ◆ Allows you to check the state of the application objects such as variables and database table values to ensure that the application is running properly.
 - ◆ Includes various features that enable you to debug your application.

- ◆ Breakpoints are places that you can specify in the code where the debugger stops the execution of the application.
- ◆ This allows you to view the current state of data in your application.
- ◆ To set up a breakpoint in Visual Studio 2013, you need to perform the following steps:
 - ◆ Open a project for example the MVCDemo in Visual Studio 2013.
 - ◆ Click the AccountController controller class in the Solution Explorer window. The Code Editor of Visual Studio 2013 displays the code of the controller class.
 - ◆ Locate the Register() action method of the controller class.
 - ◆ Right-click the beginning of the if-else statement of the Register() action method and select Breakpoint → Insert Breakpoint from the context menu that appears. A red circle on the left side of the if-else statement of the Register() action method is displayed.

Breakpoints 2-4

- ◆ Following figure shows adding a breakpoint:



- ◆ In this figure, the red block over the code line is where the debugger will open.

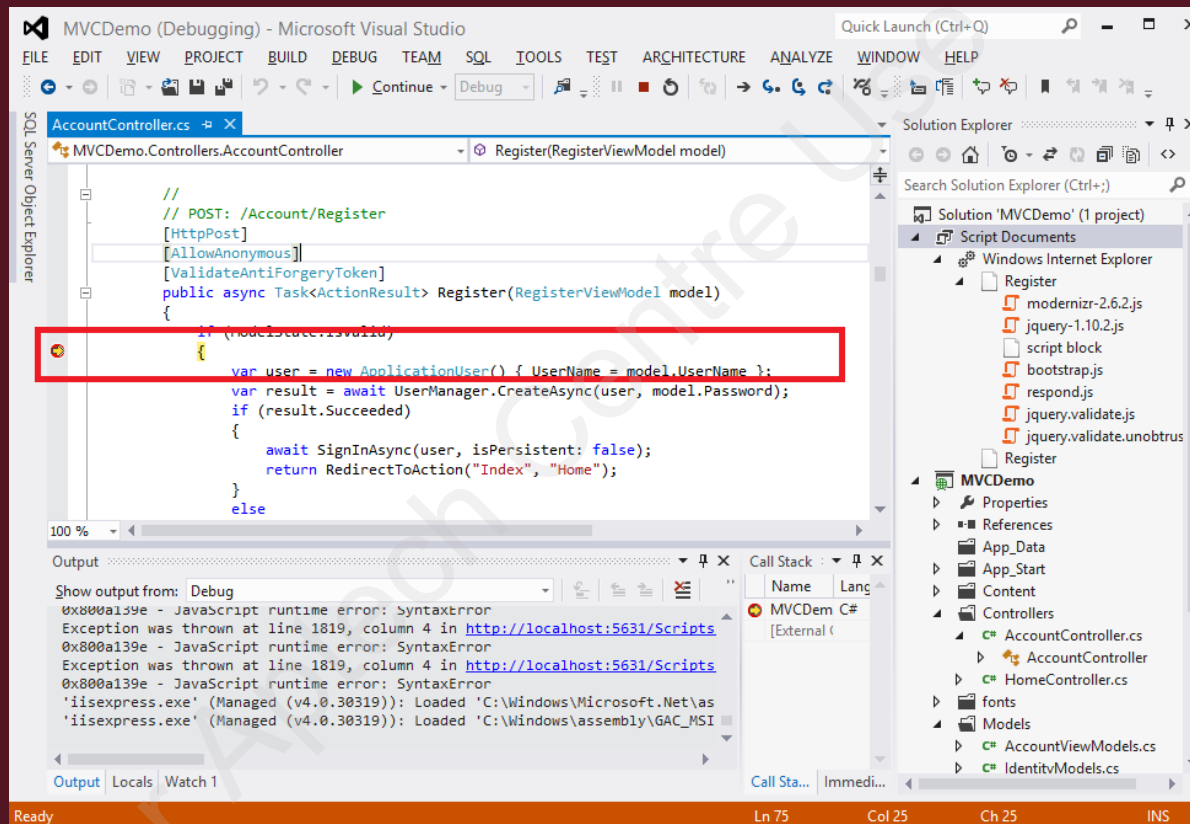
Breakpoints 3-4

- ◆ Select Debug→Start Debugging from the menu bar of Visual Studio 2013. This will start the debugging process.
- ◆ Click on the Register link on the Hope page that appears. The Registration page is displayed.
- ◆ In the Registration page, type a name in the User name text field, type a password in the Password field, and type the same password again in the Confirm password field.
- ◆ Click Register. When the program tries to execute the code where the breakpoint is added, the Code Editor displays the added breakpoint in yellow colour.

For Aptech Centre Use Only

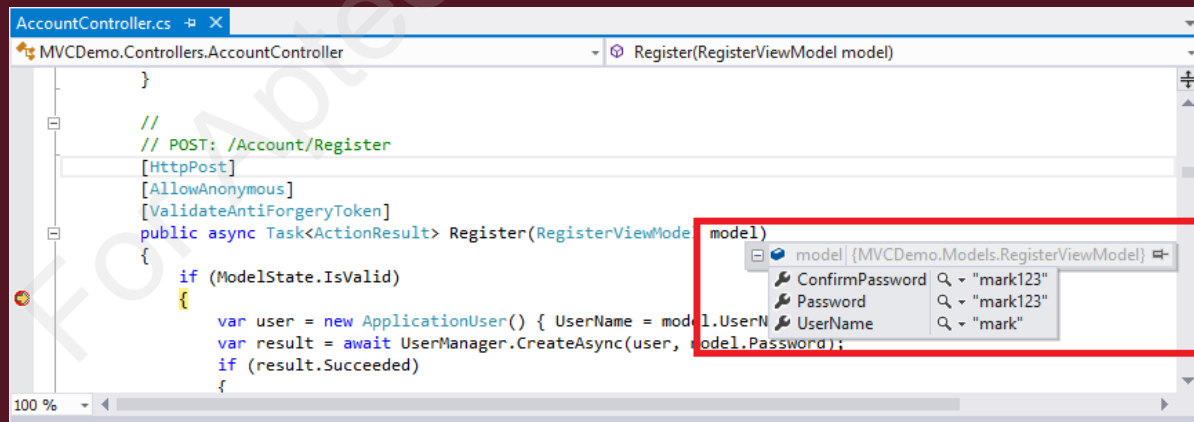
Breakpoints 4-4

- ◆ Following figure shows the breakpoint when the program tries to execute it:



Viewing Runtime Data

- ◆ You view runtime data when the application pauses at a breakpoint.
- ◆ By viewing the runtime data, you can analyse whether or not the application is processing data as expected.
- ◆ To view runtime data, click an object when the application is currently paused at a breakpoint.
- ◆ A drop-down list displays the data of the object.
- ◆ Following figure shows the data of the Model object of the default application:



- ◆ Once you set up breakpoints, you can use certain commands to step through your code line by line.
- ◆ This process of executing the code line by line is known as code stepping.
- ◆ In Visual Studio 2013, the Debug menu contains the following steps for debugging procedures:
 - ◆ Step into: You can use this command when you need to look inside a function call.
 - ◆ Step over: You can use this command when you want to avoid stepping into functions.
 - ◆ Step out: You can use this command when you are inside a function call and want to return to the calling function.

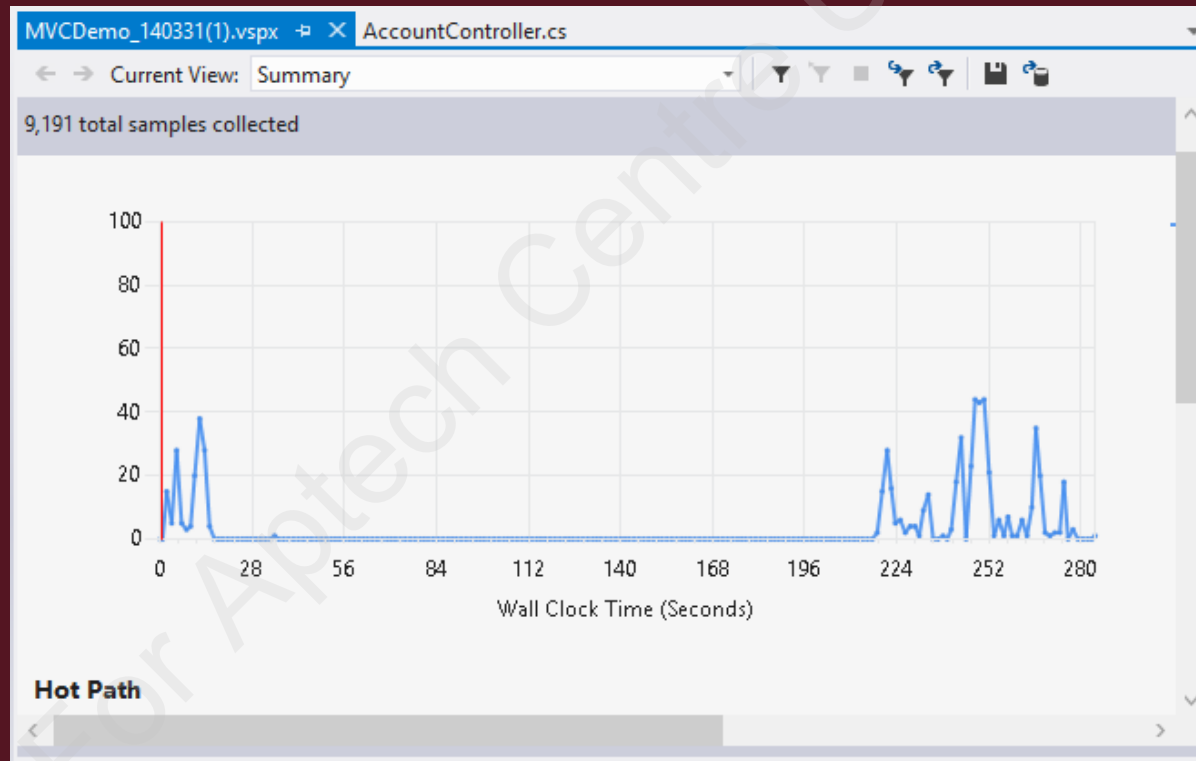
- ◆ Once you deploy an ASP.NET MVC application, you need to constantly monitor it for its proper functioning.
- ◆ By monitoring an application, you can detect the problems occurring in the application and troubleshoot them.
- ◆ ASP.NET provides health monitoring features that you can use to check for any problems in your application and then use appropriate troubleshooting technique.
- ◆ Using these features, you can monitor your application while it is running for any issues that could affect its performance.

Health Monitoring Features 1-2

- ◆ The health monitoring process:
 - ◆ Allows you to monitor the status of a deployed application.
 - ◆ Track system events and errors that affect the application performance.
 - ◆ Allows you to access the detailed run-time information about the resources that an ASP.NET MVC application uses.
- ◆ In Visual Studio 2013, the Debug menu contains a Start Performance Analysis command that you can use to check the performance of an application.
- ◆ To use the Start Performance Analysis command in Visual Studio 2013, Click Debug→Start Performance Analysis. The Output window displays the progress of the performance analysis of the application.
- ◆ Once the process of performance analysis is started you can access the application online and start performing the actions available in the application, such as register to the application and then logout.

Health Monitoring Features 2-2

- ◆ Close the browser to exit the application.
- ◆ Following figure shows the Summary section of Visual Studio 2013 displays the performance of the application in a graphical way:

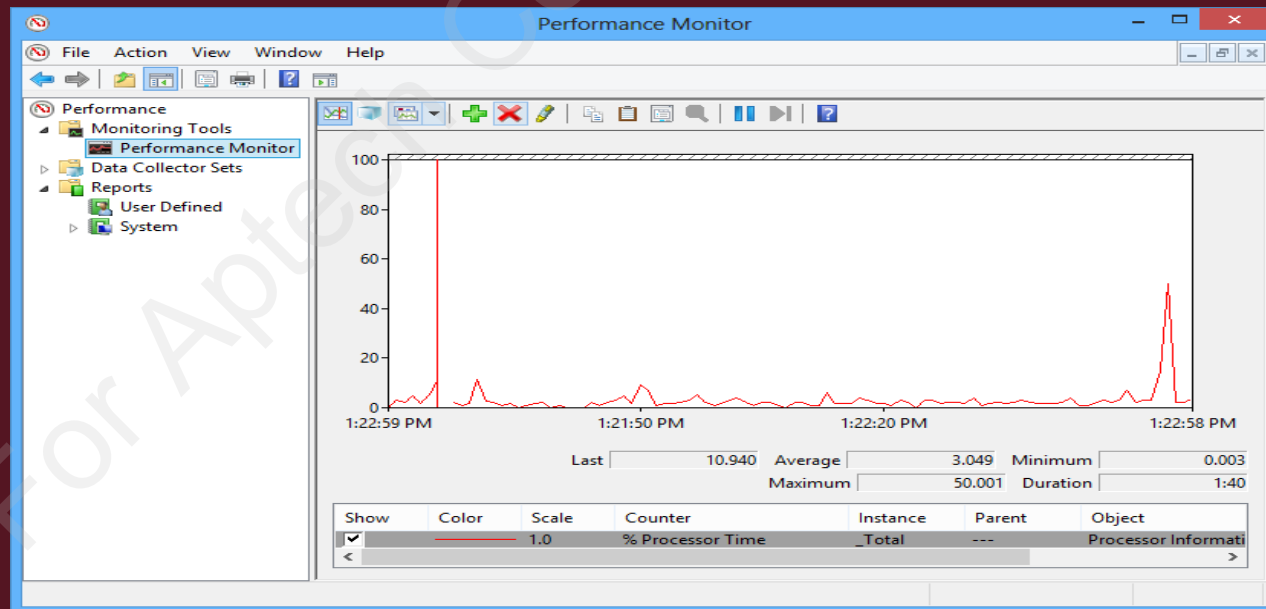


Performance Monitoring Tools 1-2

- ◆ The Windows Operating System (OS) provides the Performance Monitor tool that you can use to identify any system-level and application-level performance issues.
- ◆ As a developer you can use the data that the Performance Monitor tool provides to identify issues related to your application or related to the existing hardware environment, such as memory, disk, processor, and network.
- ◆ In the Performance Monitor tool, each of the resources that you monitor is known as performance object.
- ◆ Further, each of these objects provides counters representing data on specific aspects of a system or service.

Performance Monitoring Tools 2-2

- ◆ To use the Performance Monitor tool, you need to perform the following tasks:
 - ◆ Open Control Panel.
 - ◆ Click Administrative Tools icon in the Control Panel window. The Administrative Tool window is displayed.
 - ◆ Double-click Performance Monitor. The Performance Monitor window displays the performance of memory, processors, and physical disks.
- ◆ Following figure shows the shows the Performance Monitor window:

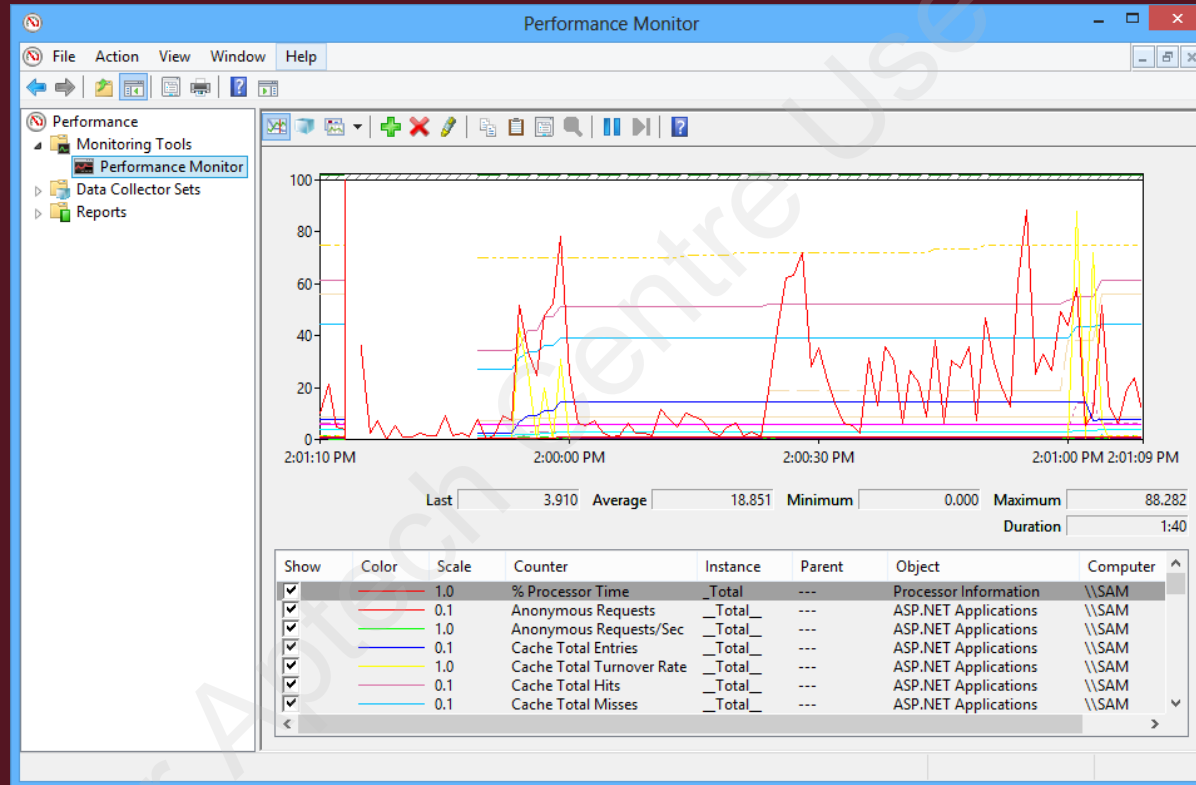


Analyzing Monitoring Results 1-2

- ◆ Once you have viewed the default performance results, you can also analyze the monitoring data of different resources.
- ◆ To analyze the monitoring result of a selected counter, you need to perform the following steps:
 - ◆ Right-click the graph and select Add Counters from the context menu to add a counter to the monitor. The Add Counters dialog box is displayed.
 - ◆ Select a counter from the Select counters from the computer drop-down list.
 - ◆ Select the counters you want to monitor and click the Add button. The Added Counters section displays the currently added counter.
 - ◆ Click OK. The Performance Monitor window displays the real-time state of the newly added counter graphically.

Analyzing Monitoring Results 2-2

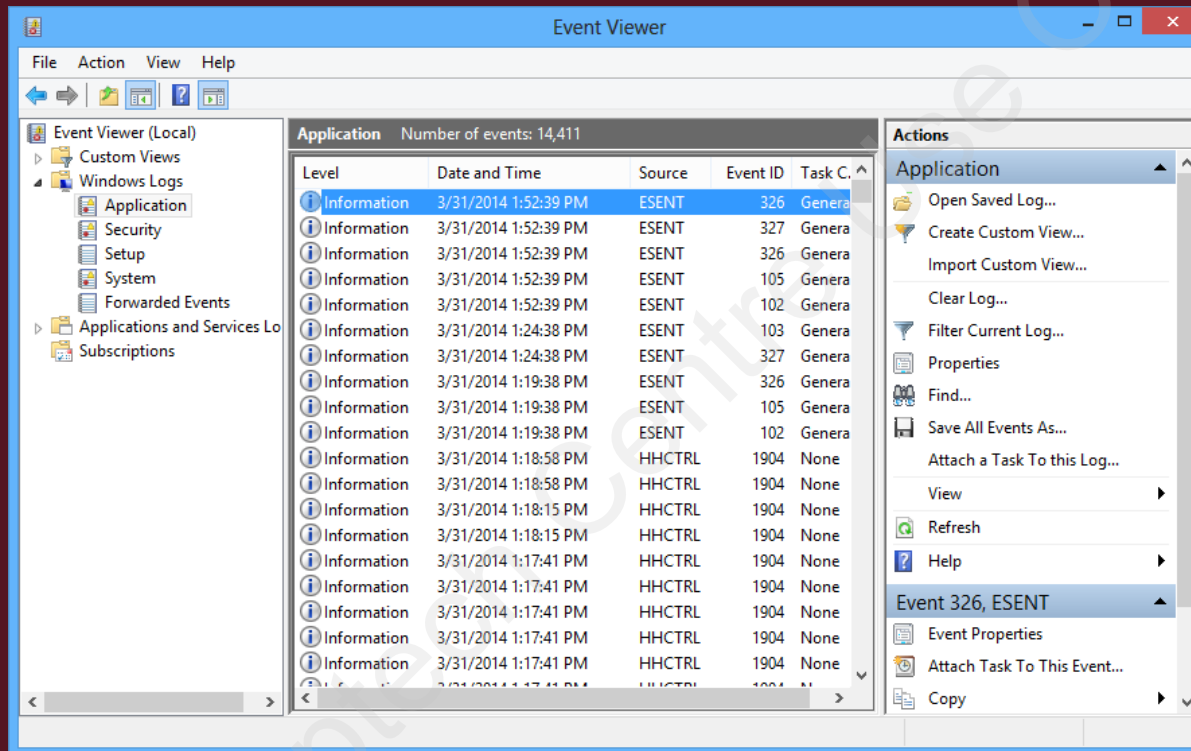
- ◆ Following figure shows the real-time state of the newly added counter:



- ◆ The Microsoft Management Console (MMC) provides an interface known as Event Viewer.
- ◆ You can use this interface to view the entries of various event logs.
- ◆ Viewer interface to view event logs, you need to perform the following steps:
 - ◆ Open Control Panel.
 - ◆ Click Administrative Tools icon in the Control Panel window. The Administrative Tools window is displayed.
 - ◆ Double-click the Event Viewer. The Event Viewer window is displayed. By default, it displays the Application, Security, and System logs.

Using Event Viewer 2-2

- ◆ Following figure shows the Event Viewer window:



- ◆ Click any one of the log to view its entries.

- ◆ Debugging is a technique that enables you to resolve such errors present in the application.
- ◆ Visual Studio debugger enables you to run your code line by line, so that you can monitor the program execution properly.
- ◆ Breakpoints are places that you can specify in the code where the debugger stops the execution of the application and thus you can view the current state of data in your application.
- ◆ You view runtime data when the application pauses at a breakpoint so that, you can analyze whether or not the application is processing data as expected.
- ◆ Once you analyze the runtime data, you can use code stepping that allows executing the code line by line.
- ◆ The health monitoring process allows you to monitor the status of a deployed application.
- ◆ The Windows Operating System (OS) provides the Performance Monitor tool that you can use to identify any system-level and application-level performance issues.