Your First 20 Minutes

PrimalScript

Your First 20 Minutes

Contents

Before Installing PrimalScript
Install PrimalScript
Launch PrimalScript
Set Script and Project Folders
Create a New Script
Insert WMI Code
Use PrimalSense
Run a Script with Arguments
Add a Function

Start here to be productive with PrimalScript in just 20 minutes

Before Installing PrimalScript

To take advantage of all possible features of PrimalScript, there are several things that should be installed before installing PrimalScript.

Install Internet Information Services (IIS)

In order to take advantage of PrimalScript's ability to create virtual web folders and to run your web projects on your local computer, Internet Information Services must be installed. While it is not required, it is recommended that you install IIS before you install PrimalScript.

Make sure to install WSH 5.6.

To enable all Windows Script Host features of PrimalScript, it is necessary to install WSH 5.6. Information on downloading the WSH can be found at www.sapien.com/wsh.htm.

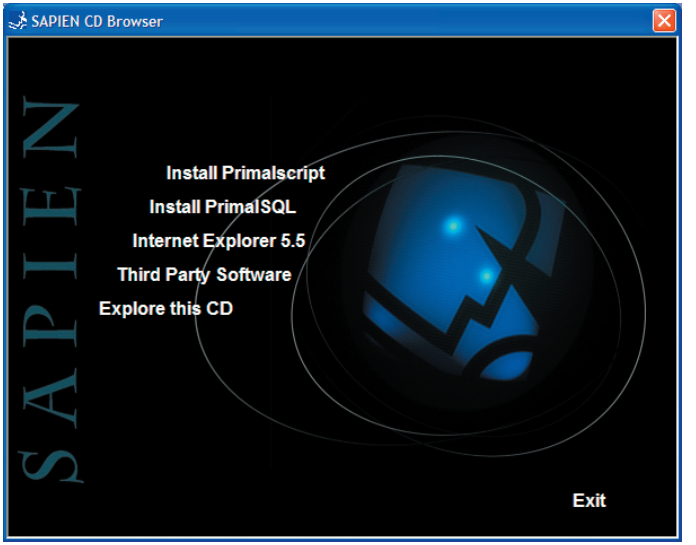
For older machines, download and install the latest WMI libraries.

To enable the use of WMI scripting on a Windows 98 or Windows NT machine, you will need to download and install WMI. See details at www.sapien.com/wmi.htm.

Install PrimalScript

Start the clock. Installing PrimalScript takes just a few minutes, so the beginning of your 20 minutes starts here.

To begin, place the PrimalScript CD into your computer's CD-ROM or DVD drive, and you should see the following autostart program.

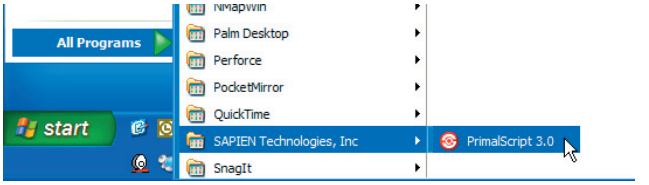


If the autostart program does not appear, manually run the cdstart.exe program from the PrimalScript CD by selecting the Run option from the Windows Start menu and running the program X:\cdstart.exe (where X: is the drive letter of your CD-ROM or DVD drive).

To launch the PrimalScript installer, select Install PrimalScript from the main menu of the autostart program. Follow the screens in the installer and PrimalScript will be installed within just a few minutes.

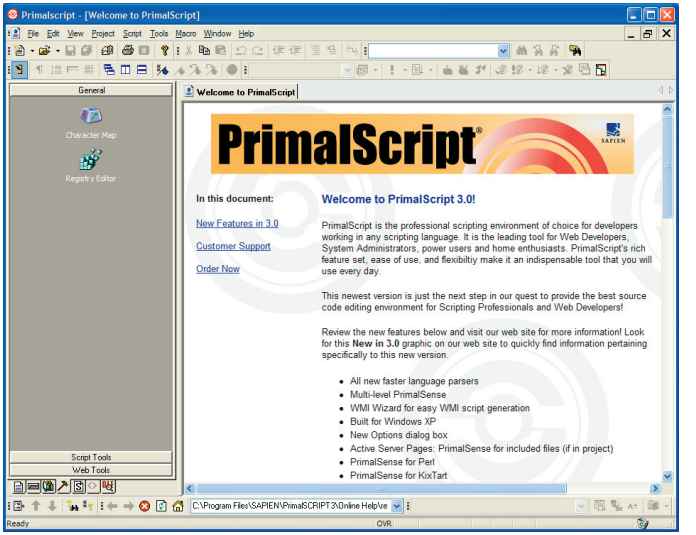
Launch PrimalScript

Now that installation is complete, launch PrimalScript by double-clicking on the PrimalScript icon that the installer created for you or start PrimalScript from the Windows Start menu.



Start PrimalScript via the Windows Start Menu.

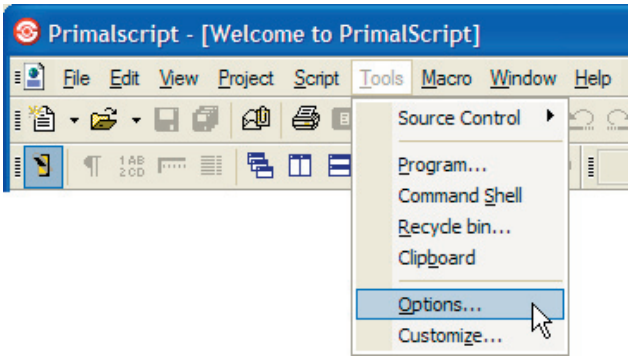
When PrimalScript starts for the first time, you will see a Getting Started page on the right side of the main window. On this page you will find all last minute updates and additional information about how to get started with PrimalScript.



PrimalScript showing the Tools Nexus on the left and the Getting Started page on the right.

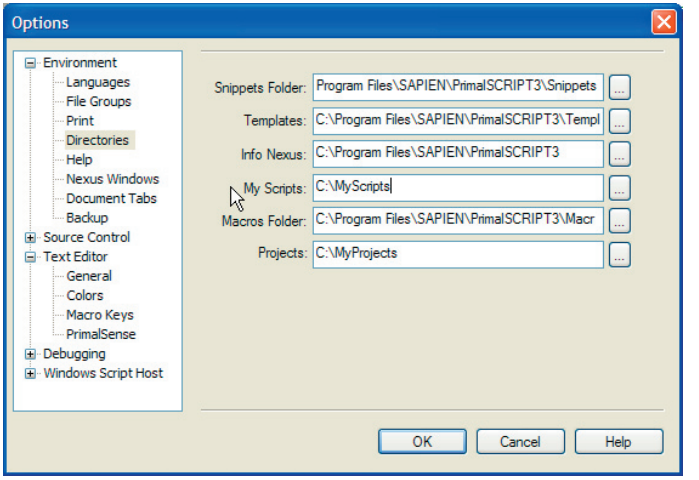
Set Script and Project Folders

Before you begin creating projects, you can set the folders in which you want your new scripts and projects to be created. Select Options... from the Tools menu.



By default, PrimalScript sets the project folder to: c:\Program Files\SAPIEN\PrimalScript\Projects

You'll see the main **Options** screen for PrimalScript. There is a tree on the left that allows you to navigate among all the options screens. Select the **Directories** branch under the **Environment** node. Change the My Scripts field and the Projects field to folders on your computer where scripts and projects will be stored by default, then click Ok.



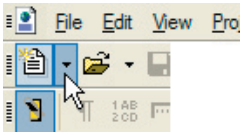
The Options dialog box holds a wide array of settings to personalize the way you use PrimalScript. Select each node in the tree to see the various screens.

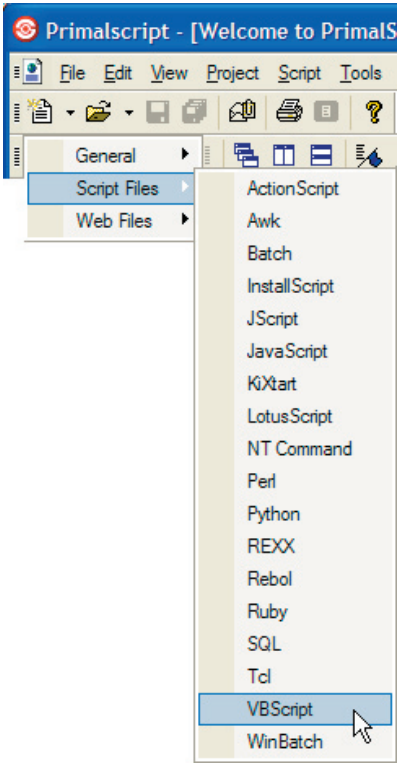
Create a New Script

Now that we have a default location for our new scripts and projects, we are going to jump right into the process of creating a WMI script that shows the **Startup commands** for your computer. We'll get acquainted with the PrimalScript main window and more as we go along.

Start by clicking the small arrow next to the new button on the toolbar and select new VBScript file as shown here.

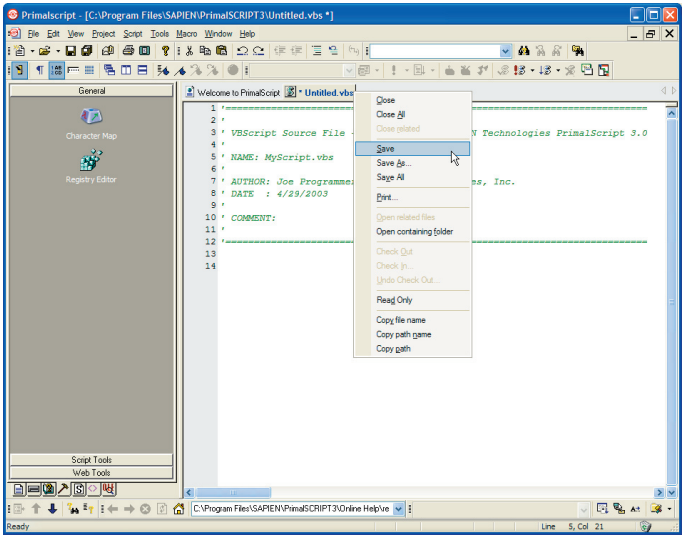
Start a new VBScript file using the New File button on the toolbar.





The new VBScript file, based on the VBScript file template is shown in the PrimalScript editing window.

The VBScript file template is customizable by editing the vbscript.vbs file in the PrimalScript program folder (c:\Program Files\SAPIEN\PrimalScript, by default).



The caret is blinking next to the word Name in the comment header. Type MyScript.vbs, then use the down arrow key to move the caret to line 14.

Save your work often by clicking on the Save button on the toolbar.

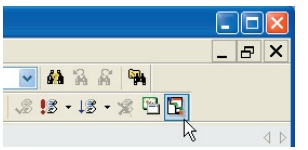
Now is a good time to save and name our new script. Right click on the document tab labeled untitled.vbs and select Save from the context menu.

Insert WMI Code

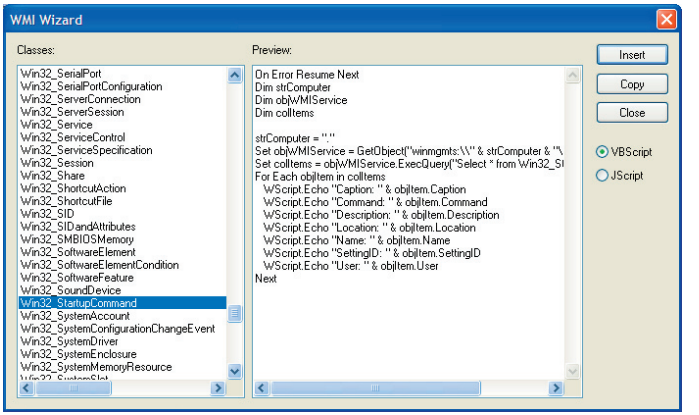
At this point we start to get down to business by adding code that will use WMI to show the Startup commands that are currently installed to load on your computer.

The WMI Script Wizard can also be started by selecting WMI Wizard from the Script menu.

Start the WMI Script Wizard by clicking on its toolbar button above the editor window.

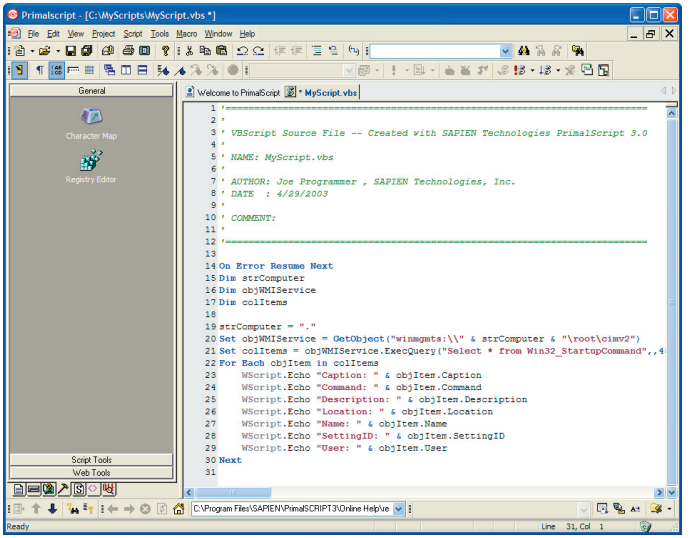


The WMI Wizard is the fast way to insert WMI code into your file.



Since we are going to be looking at Startup commands, select the **Win32_StartupCommand** class in the left pane of the wizard. Sample code appears on the right side of the window. Click the Insert button to insert the new WMI sample code into the editor window at line 14 where your caret was last.

A working WMI script that displays the Startup commands for the local machine.

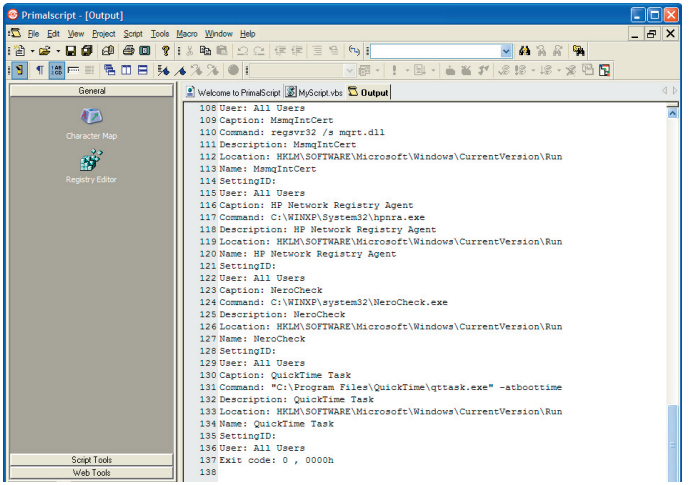


The Run Script button on the toolbar:

Click the Run Script button on the toolbar to view the output of this quick script.

Without having coded anything yet, the output shows that we have a script that displays quite a lot of information about Startup commands on a computer.

No coding yet, but we have our first script.



Click on the MyScript.vbs file tab above the editor window to return to your new script.

Use PrimalSense

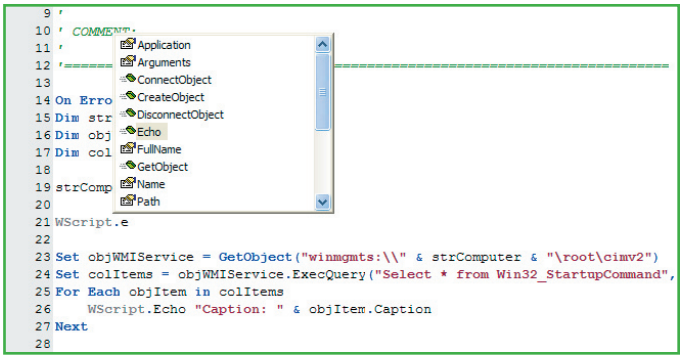
PrimalSense is the key to fast development. Keep your fingers on the keyboard and make fewer mistakes.

Now we'll enter some code and work with PrimalSense. First let's simplify the output of our script by deleting several lines of code that the wizard provided for us. Select lines 24 through 29 and press the Delete key leaving just the caption as the output for every Startup command. Run the script again to see the simplified output.

Auto-case Correction
& Pop-up Member Lists

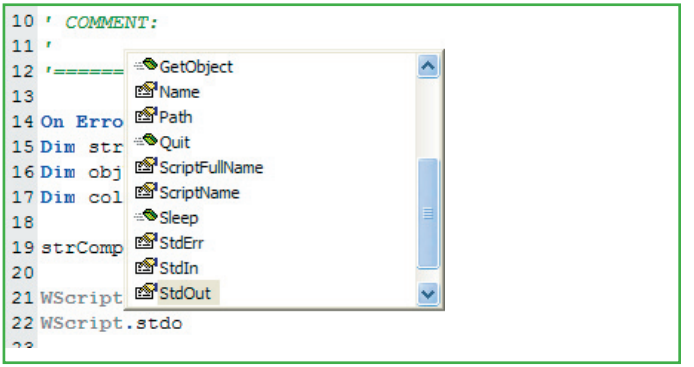
After line 19, add a few blank lines. Type the word `wscript` and a period and PrimalSense does two things: it automatically corrects the case of the word WScript and a pop-up window displays the members of the WScript object.

Now type `E` and the word Echo is selected in the list. Press the Enter key to insert the word Echo into your code.



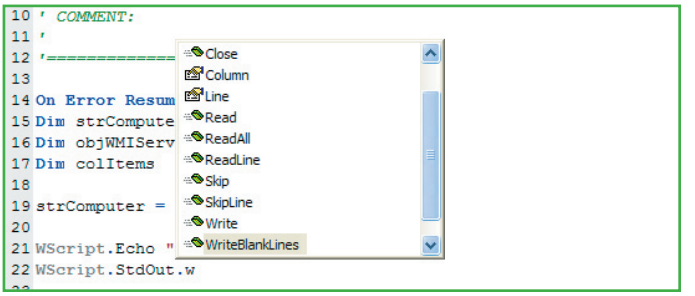
Finish the line by typing a SPACE and `"Startup commands - " & strComputer`. Press Enter to start a new line and type `wscript.stdo` and StdOut is highlighted in the pop-up list of members.

PrimalSense for the WScript object: auto-case correction and member list.



Press Enter to insert "StdOut" and then type another period to see PrimalScript's new multi-level PrimalSense and select WriteBlankLines from the list, press Enter and type a 1.

Multi-level PrimalSense for the StdOut object inside the WScript object.



And our script is ready to run again.

```
10 ' COMMENT:
11 '
12 '================================================================================
13
14 On Error Resume Next
15 Dim strComputer
16 Dim objWMIService
17 Dim colItems
18
19 strComputer = "."
20
21 WScript.Echo "Startup Commands - " & strComputer
22 WScript.StdOut.WriteBlankLines 1
23
24 Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")
25 Set colItems = objWMIService.ExecQuery("Select * from Win32_StartupCommand",
26 For Each objItem in colItems
27     WScript.Echo "Caption: " & objItem.Caption
28 Next
29
```

Using the screen shown here as a guide, add code at line 21 to check the arguments to this script. As you type, take note of the PrimalSense pop-up windows that appear and continue to get familiar with how they operate.

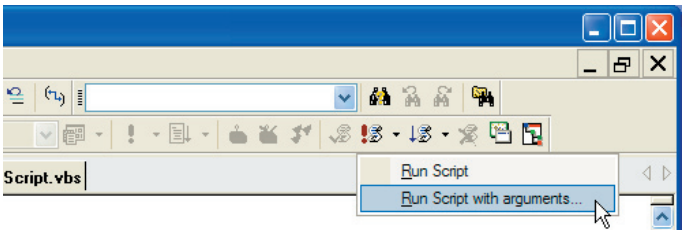
Add code to detect arguments to the script and watch PrimalSense help.

```
19 strComputer = "."
20
21 If WScript.Arguments.Count > 0 Then
22     strComputer = WScript.Arguments(0)
23 End if
24
25 WScript.Echo "Startup Commands - " & strCopmuter
26 WScript.StdOut.WriteBlankLines 1
27
28 Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")
29 Set colItems = objWMIService.ExecQuery("Select * from Win32_StartupCommand",
30 For Each objItem in colItems
31     WScript.Echo "Caption: " & objItem.Caption
32 Next
```

The code we just added looks at the arguments coming into the script and assigns the first argument as the computer name for which you'd like to see Startup commands.

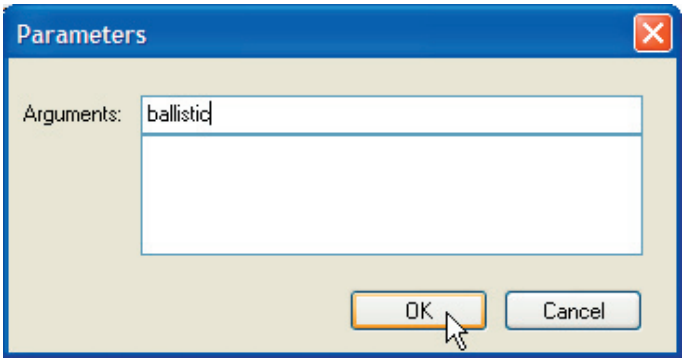
Run a Script with Arguments

PrimalScript lets us easily run a script for which we will be asked to enter arguments. Using the down arrow next to the Run Script button, select Run Script with arguments.



The Run Script button has a drop down menu for running with and without arguments.

PrimalScript will show the script arguments dialog box for you to enter arguments for this script. For this exercise, enter the name of your computer or the name of a computer on your network if you have Domain Administrator rights for your network, click Ok.



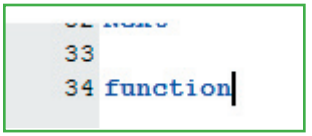
The Arguments (or Parameters) that you enter are saved in a the history list for future use.

Now your script is run using that argument and the script tries to display the Startup commands for the machine whose name you entered. If you do not enter a name, the script defaults to the local machine.

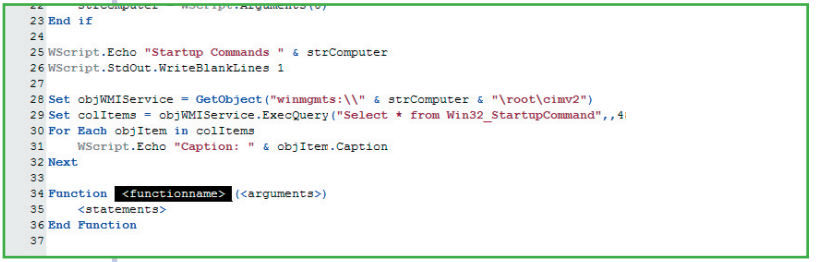
Add a Function

Now we will add a function to the code to demonstrate both code snippets, code completion and function parameter help. At the end of the file, type the word `function` and press CTRL-J to enact the insertion of the code snippet named "function."

With the caret right of the snippet name, press CTRL-J to insert that snippet.



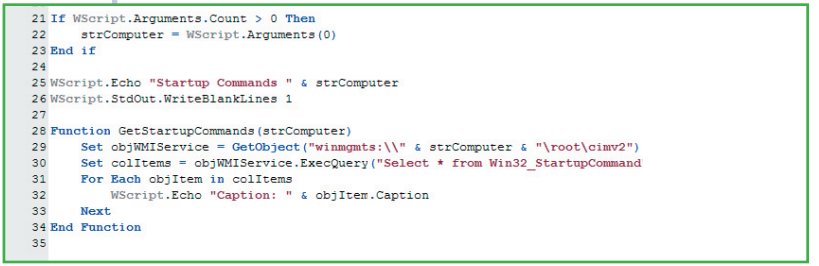
The "function" snippet after insertion. All snippets are stored in the Snippets Nexus.



A fast way to move a highlighted section of code is to simply click in it drag it to its destination.

Call the name of the function **GetStartupCommands** and give it a parameter called **strComputer**. Highlight lines 28 through 34 and drag them inside the function. While those lines are still highlighted, press the TAB key to indent the whole block.

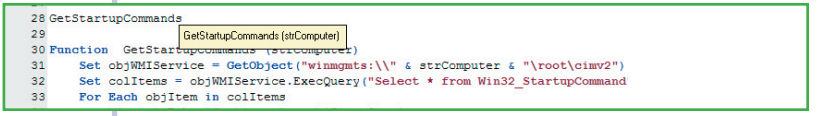
The completed function.



Code Completion

Now we need to call this function from the body of the script. On line 28, we will add some code by typing `getst` and then pressing CTRL-SPACE to enact code completion. PrimalScript fills in the function name GetStartupCommands. Press the space bar and PrimalScript displays parameter help for the function that we are calling.

Parameter Help



Type `strComputer` as your parameter and save your file. You now have a functioning script for checking the Startup commands for any computer.

The completed script.

```
10 ' COMMENT:
11 '
12 '=====================================================================
13
14 On Error Resume Next
15 Dim strComputer
16 Dim objWMIService
17 Dim colItems
18
19 strComputer = "."
20
21 If WScript.Arguments.Count > 0 Then
22     strComputer = WScript.Arguments(0)
23 End if
24
25 WScript.Echo "Startup Commands - " & strComputer
26 WScript.StdOut.WriteBlankLines 1
27
28 GetStartupCommands strComputer
29
30 Function  GetStartupCommands (strComputer)
31     Set objWMIService = GetObject("winmgmts:\\" & strComputer & "\root\cimv2")
32     Set colItems = objWMIService.ExecQuery("Select * from Win32_StartupCommand
33     For Each objItem in colItems
34         WScript.Echo "Caption: " & objItem.Caption
35     Next
36 End Function
```

Running the script again with no arguments results in the listing of Startup commands for the local machine, which is represented as a period (.).

Here is the output window when our script is run with arguments and we enter a machine name “ballistic” as the argument.

Final output using Ballistic as the argument to the script.

```
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.

Startup Commands - ballistic

Caption: desktop
Caption: desktop
Caption: MSMSGS
Caption: AIM
Caption: Yahoo! Pager
Caption: Mozilla Quick Launch
Caption: desktop
Caption: Adobe Gamma Loader.exe
Caption: Adobe Gamma Loader
Caption: desktop
Caption: got password
Caption: HotSync Manager
Caption: Microsoft Office
Caption: PowerReg Scheduler
Caption: Service Manager
Caption: MsmqIntCert
Caption: HP Network Registry Agent
Caption: NeroCheck
Caption: QuickTime Task
Exit code: 0 , 0000h
```

Stop the Clock!

That was your first 20 minutes with PrimalScript. We hope you enjoyed this quick tour. You've actually learned a lot in less than ½ an hour!

You now know how to change program options, create new script files, insert WMI script code, use PrimalSense for faster script development, and run scripts with arguments.

More information and support.

For more information, view the PrimalScript help file by selecting Index and Contents from the Help menu. You may also send e-mail to support@sapien.com with any questions about using PrimalScript.

Thanks for choosing PrimalScript and we look forward to working with you long into the future!

PrimalScript, PrimalCode, and PrimalSense are either trademarks or registered trademarks of SAPIEN Technologies, Inc. All other trademarks are the property of their respective owners. Copyright © 2003,2004 SAPIEN Technologies, Inc. All Rights Reserved.

SAPIEN Technologies, Inc.
3212 Jefferson Street, #288
Napa, CA 94558
1-707-252-8700
1-866-PRIMALS (774-6257) Toll Free
1-707-598-0160 FAX
http://www.sapien.com

ps31_gs_20040101