

**Chapter  
1**

**GETTING  
STARTED**

*Get on the  
Fast Track!*



TM

**SYS-ED/  
COMPUTER  
EDUCATION  
TECHNIQUES, INC.**

**Objectives**

You will learn:

- C Use of scripts in web applications.
- C Scripts in web applications and browsers.
- C How to embed VBScript in a HTML document.
- C <SCRIPT> tag.
- C Handling events.
- C Commenting conventions.
- C Coding style.
- C How to write reusable scripts.

---

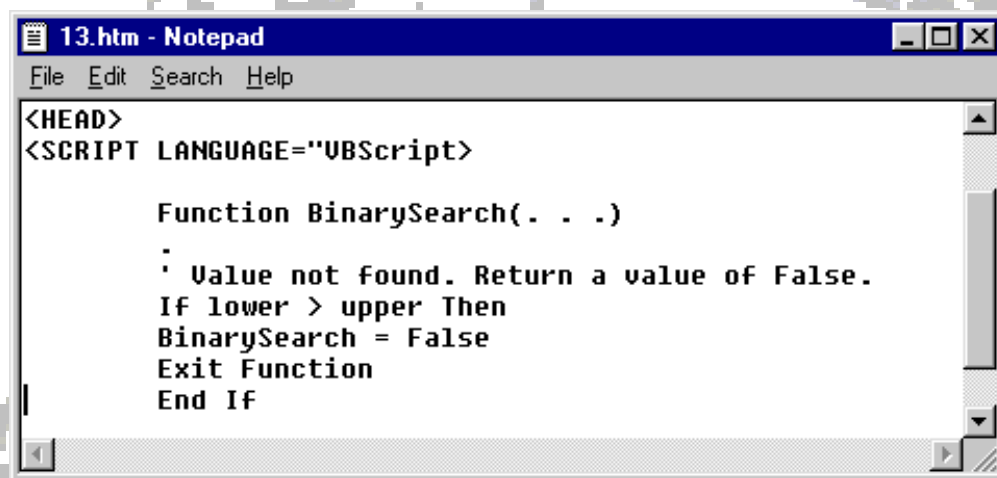
## 1 Scripts in Web Applications

HTML tags are used for displaying text, images, or links in a page.

A script or program can be embedded in a web page to perform specific functions and used for:

- C Controlling what happens when a user clicks a button, enters text, or submits a form.
- C Navigating to a specific page based on a condition such as user preference.
- C Collecting and storing user information in order to customize Web applications dynamically.
- C Querying a database and displaying results.

### Sample VBScript Source Code

A screenshot of a Notepad window titled "13.htm - Notepad". The window contains the following VBScript code:

```
<HEAD>
<SCRIPT LANGUAGE="VBScript">

    Function BinarySearch(. . .)
    :
    : Value not found. Return a value of False.
    If lower > upper Then
    BinarySearch = False
    Exit Function
    End If
```

---

## 2 VBScript in Applications and Browsers

Microsoft provides binary implementations of VBScript for:

C	32-bit Windows API	C	16-bit Windows API	C	Macintosh
---	--------------------	---	--------------------	---	-----------

VBScript is integrated with World Wide Web browsers.

VBScript and ActiveX scripting can also be used as a general scripting language in other applications.

---

### 3 Adding VBScript to a HTML Document

VBScript codes can be added to a HTML file in the following ways:

1. Using the SCRIPT element to add VBScript code to an HTML page.
2. As inline codes--to handle events of form objects.

---

#### 3.1 The <SCRIPT> Tag

VBScript code is written within paired <SCRIPT> tags.

##### Example:

This procedure tests a delivery date:

```
<SCRIPT LANGUAGE="VBScript">
<!--
    Function CanDeliver(Dt)
        CanDeliver = (CDate(Dt) - Now()) > 2
    End Function
-->
</SCRIPT>
```

Beginning and ending <SCRIPT> tags surround the code. The LANGUAGE attribute indicates the scripting language.

The language must be specified because browsers can use other scripting languages.

The CanDeliver function is embedded in comment tags (<!-- and -->). This prevents browsers that don't understand the <SCRIPT> tag from displaying the code.

Since the example is a general function—it isn't tied to any particular form control—it can be included in the HEAD section of the page:

```
<HTML>
<HEAD>
<TITLE>Place Your Order</TITLE>
<SCRIPT LANGUAGE="VBScript">
<!--
    Function CanDeliver(Dt)
        CanDeliver = (CDate(Dt) - Now()) > 2
    End Function
-->
</SCRIPT>
</HEAD>
<BODY>
.....
```

---

### 3.2 Location of <SCRIPT> Block

SCRIPT blocks can be used anywhere in an HTML page. They can be placed in both the BODY and HEAD sections. However, it is a good practice to put all general-purpose scripting code in the HEAD section.

Keeping your code in the HEAD section ensures that all code is read and decoded before it's needed by any calls from within the BODY section.

---

### 3.3 Event Handler

Inline scripting code can be provided within forms in order to respond to the events of objects in the form.

#### Example:

This embedded scripted code responds to a button clicked in a form:

```
1      <HTML>
2      <HEAD>
3      <TITLE>Test Button Events</TITLE>
4      </HEAD>
5      <BODY>
6      <FORM NAME="Form1">
7          <INPUT TYPE="Button" NAME="Button1" VALUE="Click">
8          <SCRIPT FOR="Button1" EVENT="onClick" LANGUAGE="VBScript">
9              MsgBox "Button Pressed!"
10     </SCRIPT>
11     </FORM>
12 </BODY>
13 </HTML>
```

Most code appear in either Sub or Function procedures and will be called only when code has been written which causes it to execute. However, VBScript code can be written outside procedures, but still within a SCRIPT block.

This code is executed only once, when the HTML page loads. This allows you to initialize data or dynamically change the look of your Web page when it loads.

---

## 4 VBScript Page

Microsoft Internet Explorer can be used for viewing the page produced by the following HTML code.

When the button on the page is clicked, the VBScript will run and a dialog box will display "Hello, programmer".

```
1      <HTML>
2      <HEAD><TITLE>My First VBScript Page</TITLE>
3      <SCRIPT LANGUAGE="VBScript">
4      <!--
5      Sub cmdWelcome_OnClick
6          MsgBox "Hello, programmer"
7      End Sub
8      -->
9      </SCRIPT>
10     </HEAD>
11     <BODY>
12     <H3>This Page Contains VBScript Code</H3><HR>
13     <FORM><INPUT NAME="cmdWelcome" TYPE="BUTTON" VALUE="Welcome"></FORM>
14     </BODY>
15     </HTML>
```

When Internet Explorer reads the page, it finds the <SCRIPT> tags, recognizes there is a piece of VBScript code, and saves the code.

When the button is clicked, Internet Explorer makes the connection between the button and the code, and runs the procedure.

The Sub procedure in the <SCRIPT> tags is an event procedure.

There are two parts to the procedure name:

- C the name of the button, cmdWelcome, from the NAME attribute in the <INPUT> tag.
- C an event name, OnClick.

The two names are joined by an underscore(\_). Any time the button is clicked, Internet Explorer looks for and runs the corresponding event procedure, cmdWelcome\_OnClick.

Pages can use combinations of controls and procedures, too. VBScript and Forms shows some simple interactions between controls.

---

## 4.1 Inline Coding

VBScript code can be attached to events in two other ways.

Internet Explorer allows short sections of inline code to be added in the tag defining the control.

When the button on the page is clicked, the VBScript will run and a dialog box will display "Hello, programmer".:

```
1     <INPUT NAME="cmdWelcome" TYPE="BUTTON"  
2         VALUE="Welcome" OnClick='MsgBox "Welcome programmer" '>
```

The function call itself is enclosed in single quotation marks, and the string for the MsgBox function is enclosed in double quotation marks. Multiple statements can be used as long as the statements are separated with colons (:).

A <SCRIPT> tag can also be written so that it applies only to a particular event for a specific control:

```
1     <SCRIPT LANGUAGE="VBScript" EVENT="OnClick" FOR="cmdWelcome">  
2     <!--  
3         MsgBox "Welcome programmer"  
4     -->  
5     </SCRIPT>
```

Since the <SCRIPT> tag already specifies the event and the control, the Sub and End Sub statements are not used.

---

## 5 Commenting Conventions

All procedures should begin with a brief comment describing what they do.

This description should not describe the implementation details because these often change over time, resulting in unnecessary comment maintenance work, or worse, erroneous comments. The code itself and any necessary inline comments describe the implementation.

Arguments passed to a procedure should be described when their purpose is not obvious and when the procedure expects the arguments to be in a specific range. Return values for functions and variables that are changed by a procedure, especially through reference arguments, should also be described at the beginning of each procedure.

### Guidelines

- C Every important variable declaration should include an inline comment describing the use of the variable being declared.
- C Variables, controls, and procedures should be named clearly in order that inline comments are only needed for complex implementation details.
- C At the beginning of the script, include an overview that describes the script, enumerating objects, procedures, algorithms, dialog boxes, and other system dependencies. Pseudocode describing the algorithm can be helpful.

---

### 5.1 Adding Comments

Explanatory remarks can be included in a program by using:

- C `Rem comment`
- or
- C `' comment`

The comment argument is the text of any comment to be included. After the Rem keyword, a space is required before comment.

An apostrophe (') can be used instead of the Rem keyword. If the Rem keyword follows other statements on a line, it must be separated from the statements by a colon.

However, when using an apostrophe, the colon is not required after other statements.

**Example:**

This code illustrates the use of the Rem statement:

```
Dim MyStr1, MyStr2
MyStr1 = "Hello" : Rem Comment after a statement separated by a colon.
MyStr2 = "Goodbye" ' This is also a comment; no colon is needed.
Rem Comment on a line with no code; no colon is needed.
```

---

## 6 Formatting the Code

Screen space should be conserved as much as possible, while still allowing code formatting to reflect logic structure and nesting.

### Guidelines:

- C Standard nested blocks should be indented four spaces.
- C The overview comments of a procedure should be indented one space.
- C The highest level statements that follow the overview comments should be indented four spaces, with each nested block indented an additional four spaces.

### Example:

```
*****
' Purpose:  Locates the first occurrence of a specified user.
' Inputs:  strUserList():  the list of users to be searched.
' strTargetUser:  the name of the user to search for.
' Returns:  The index of the first occurrence of the strTargetUser.
'          array. If the target user is not found, return -1.
*****

Function intFindUser (strUserList(), strTargetUser)
    Dim I                ' Loop counter.
    Dim blnFound        ' Target found flag
    intFindUser = -1
    I = 0                ' Initialize loop counter
    Do While I <= Ubound(strUserList) and Not blnFound
        If strUserList(I) = strTargetUser Then
            blnFound = True    ' Set flag to True
            intFindUser = I    ' Set return value to loop count
        End If
        I = I + 1          ' Increment loop counter.
    Loop
End Function
```

---

## 7 Writing Reusable Scripts

In most Web applications, blocks of HTML content will be displayed on multiple pages. Likewise, a script can be included in multiple files.

A script can include references to other files in the <SCRIPT> block. The SRC attribute is used for specifying the name of a file that contains the script to be included.

For example, the following <SCRIPT> block includes a reference to a page containing error message routines.

```
<SCRIPT SRC="Errmsg.htm"></SCRIPT>
```

After a reference has been made to a file in this way, script can be called on the page as if it were in the current page.