Including inline script

JavaScript code can be included in a web page by adding HTML **<script> </script>** tags, to enclose the script, and the opening tag must have a **type** attribute specifying the unique MIME type of "text/javascript" – to identify the element's contents as JavaScript.

A HTML **<script>** element may also include helpful code comments. The JavaScript engine ("parser") ignores everything between **/*** and ***/** characters, allowing multi-line comments, and ignores everything between **//** characters and the end of a line, allowing single-line comments – like this:

```
<script type="text/javascript">

/* This is a multi-line comment that might describe the script's
purpose and provide information about the author and date. */

// This is a single line comment that might describe a line of code.

</script>
```

Alternative text can be provided, for occasions when JavaScript support is absent or disabled, by adding **<noscript> </noscript>** HTML tags to enclose an explanatory message.

The **<script>** element can appear anywhere within the HTML document's body section to include "inline" JavaScript code, which will be executed as the browser reads down the document. Additionally inline JavaScript code can be assigned to any of the HTML event attributes, such as **onload**, **onmouseover**, etc., which will be executed each time that event gets fired by a user action.

Hot tip

MIME (Multipart Internet Mail Extension) types describe content types – **text/html** for HTML, **text/css** for style sheets, and **text/javascript** for JavaScript code.

inline.html

1. Create a HTML document and add a **<div>** element to its body section, in which to write from JavaScript, and assign its **id** attribute a value of "panel"

```
<body>
        <div id="panel">        </div>
</body>
```

2. In the **<div>** element, insert a **<script>** element containing inline code to write a greeting in the panel

```
<script type="text/javascript">

// Dynamically write a text string as this page loads.
document.write( "Hello World!" ) ;

</script>
```

Don't forget

Notice how each JavaScript statement must be terminated by a semi-colon character.

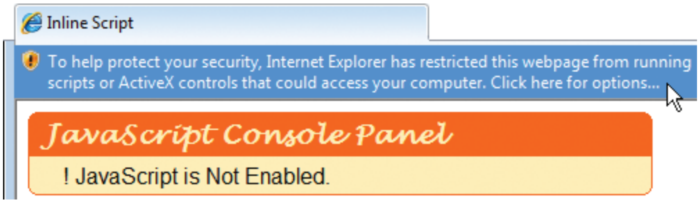
3 After the **<script>** element, insert a **<noscript>** element for alternative text when JavaScript support is absent

```
<noscript>
        <div>! JavaScript is Not Enabled.</div>
</noscript>
```

4 Now add an attribute to call a JavaScript method whenever the document gets loaded into the browser

```
// Display a message dialog after the page has loaded.
<body onload="window.alert( 'Document Loaded!' );">
```

5 Save the HTML document and disable JavaScript support in your browser, then open the web page to see the alternative text get written in the panel



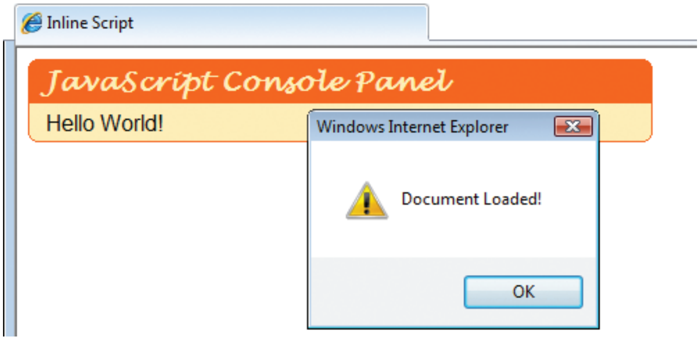
Inline Script

To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer. Click here for options...

JavaScript Console Panel

! JavaScript is Not Enabled.

6 Enable JavaScript support to see the inline script write the greeting in the panel and open a dialog box



Inline Script

JavaScript Console Panel

Hello World!

Windows Internet Explorer

Document Loaded!

In this example the JavaScript code calls upon the **write()** method of the **document** DOM object, to write the text string within its parentheses into the HTML document, then calls upon the **alert()** method of the **window** DOM object to display the text string specified within its parentheses on the face of a dialog box.

Hot tip

Extra HTML elements have been added around the panel and styled to give it the Web 2.0 look – the actual panel **<div>** contains black text.

Beware

Text strings must be enclosed within quote characters. Nested inner strings should be surrounded by single quote characters to avoid conflict with the double quote characters that surround outer strings.