JAVASCRIPT COMMENTS

Javascript Comments

The // signifies a Javascript comment, similar to the <!-- HTML comment --> comment tag. This style of comment is restricted to a single line, so either goes on a line by itself, as here, or at the end of a line of code. Since anything following a // is treated as a comment.

A multiline comment begins with a forward slash and asterisk in combination (/*), and **must** end with the reverse (*/). UNless commenting out a large chunk of code, most people will simply use the single line comment. When the comment runs over more than one line, each new line is started with //.

Using comments

Comments allow you to annotate your script, and are useful as a reminder of what the script is doing and why. The double slash indicates a single line comment. Everything after it on the same line will be ignored by the script engine. The slash-asterisk indicates a block comment. Everything after it will be ignored by the script engine until an asterisk-slash is encountered.

```
<script type="text/javascript">
//This is a single line comment
/*
   This is a block comment and
   can extend over many lines.
*/
</script>
```

Commenting out script tags

This is not needed any more. All current browsers are aware of script tags, and how to treat their contents, since they have been part of the HTML standard since HTML 3. Browsers that do not understand HTML 3 or scripts (these are virtually none now) will display the script as if it was the content of the page. You can hide the script from them by commenting out your script with standard HTML comments. Browsers that understand script will simply ignore these comments, and will just interpret the script within them. The opening comment is ignored as if it were a normal single line JavaScript comment. The closing comment will not, so it needs to have a JavaScript comment put before it:

```
<script type="text/javascript">
<!--
//JavaScript goes here
//-->
</script>
```

HTML comments cannot be used anywhere within your scripts. They will not be recognised, and so cause an error.