

3. JavaScript Output

JavaScript is typically used to manipulate HTML elements.

3.1. Manipulating HTML Elements

To access an HTML element from JavaScript, you can use the `document.getElementById(id)` method.

Use the "id" attribute to identify the HTML element:

Example

```
Access the HTML element with the specified id,
and change its content:
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>

<p id="demo">My First Paragraph</p>

<script>
document.getElementById("demo").innerHTML="My
First JavaScript";
</script>

</body>
</html>
```

The JavaScript is executed by the web browser. In this case, the browser will access the HTML element with `id="demo"`, and replace its content (innerHTML) with "My First JavaScript".

3.2. Writing to The Document Output

The example below writes a `<p>` element directly into the HTML document output:

Example

```
<!DOCTYPE html>
<html>
<body>
```

```
<h1>My First Web Page</h1>

<script>
document.write("<p>My First JavaScript</p>");
</script>

</body>
</html>
```

Warning

Use `document.write()` only to write directly into the document output.

If you execute `document.write` after the document has finished loading, the entire HTML page will be overwritten:

Example

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>

<p>My First Paragraph.</p>

<button onclick="myFunction()">Try it</button>

<script>
function myFunction()
{
document.write("Oops! The document
disappeared!");
}
</script>

</body>
</html>
```

JavaScript in Windows 8



Microsoft supports JavaScript for creating Windows 8 apps.
JavaScript is definitely the future for both the Internet and Windows.