Introduction to JavaScript

JavaScript is a high-level language that is **interpreted** - translated into machine language at the time of usage (run-time)

JavaScript allows for dynamic, real-time changes to the web page the user is accessing. The user causes an "event" (e.g., moving the mouse to press a button) and the programmer can use JavaScript to program a response.

JavaScript was developed by Netscape as a web programming language.

Characteristics of the language:
- allows interactive content on a Web page.
- client based (works on the user machine).
- does not manipulate files.
- does not carry out graphics.

Important Issues for JavaScript:
- the instructions are written in lowercase.
- all instructions must be spelled correctly or the interpreter will not understand them.
- parts of an instruction must be separated by a space and not run together.
- the correct punctuation must be used.

Overview of JavaScript

Inserting JavaScript into a Web page:
```
<script language="JavaScript" type="text/javascript">
    ...
</script>
```

JavaScript alert("string") function:
> The **"alert"** function requests that the browser pop-up a small window that contains the words in the string. The "string" must be enclosed within a set of either double quotes ("...") or single quotes ('...').

```
<script language="JavaScript" type="text/javascript">
    alert("your message goes here");
</script>
```

JavaScript prompt("string") function:
> The **"prompt"** function causes the browser to pop-up a small window that contains the words in the string. In addition, it provides a text box for user input. The value (or string) that is input, is assigned to the variable.

```
<script language="JavaScript" type="text/javascript">
    var mytext = prompt("Please enter some text");
    alert(mytext);
</script>
```

Static write to a window:
```
<script language="JavaScript" type="text/javascript">
    window.document.write("This is a statement");
</script>
```

alternate version: (note commands to change colors)
```
<script language="JavaScript" type="text/javascript">
    document.write("This is a statement");
    document.bgColor = "yellow";
    document.fgColor = "blue";
</script>
```

JavaScript confirm() method:
```
<script language="JavaScript" type="text/javascript">
    var reply = confirm("Do you like this color?");
    document.write("Your answer was " + reply);
</script>
```

Note: The variable **"reply"** will have a value of **true** or **false**.

Example:
    See demo

Mouse Events:
OnMouseOver Event Handler:
- The user moves the mouse over a particular part of the Web page (as defined within an anchor).
- First, the programmer has to define the part of the Web page to be monitored.
- If the user has moved the mouse there, the program will detect this and react in some way.

```
<a href = "#"
    onMouseOver = "document.bgColor = 'red';
    return true"
>
    Watch me!
</a>
```

Note: two levels of quotes are needed.

OnMouseOut Event Handler:
- The user moves the mouse away from the referenced part of the Web page.
- The program will detect this and react.

```
<a href = "#"
    onMouseOver = "document.bgColor = 'red';
    return true"
    onMouseOut = "document.bgColor = 'white';
    return true"
>
    Watch me!
</a>
```

Example:
See demo

Opening a New Window:
window.open();

Opening a New Window with Content:
window.open(newpage);

e.g.,
window.open("www.brooklyn.cuny.edu");

Textboxes and Buttons:
Use the **<input>** tag to create textboxes and buttons

e.g.,
Enter your name:
<input type=text id="name1" />

Enter another name:
<input type=text id="name2" value="Mickey Mouse"/>

<input type=button value="Press Here When Ready" />

Using document.getElementById():
Use document.getElementById("*id of textbox*") to read from and write to a textbox.

document.getElementById('name1').value = "John Doe";

answer = prompt("Enter another name",
    document.getElementById('name2').value);
document.getElementById('name2').value = answer;

Example:
See Demo