INTRODUCTION TO JAVASCRIPT

WHAT IS JAVASCRIPT?

- JavaScript is the scripting language of the web.
- A scripting language is a lightweight programming language embedded directly into HTML document.
- JavaScript is a client side scripting language.
- JavaScript is an interpreted language which means that scripts execute without preliminary compilation.
- JavaScript works in all major browsers, such as Internet explorer , firefox, chrome, opera & safari.

WHAT IS JAVASCRIPT?

- JavaScript is designed to add interactivity to HTML pages
- A JavaScript consists of lines of executable computer code
- A JavaScript is usually embedded directly into HTML pages

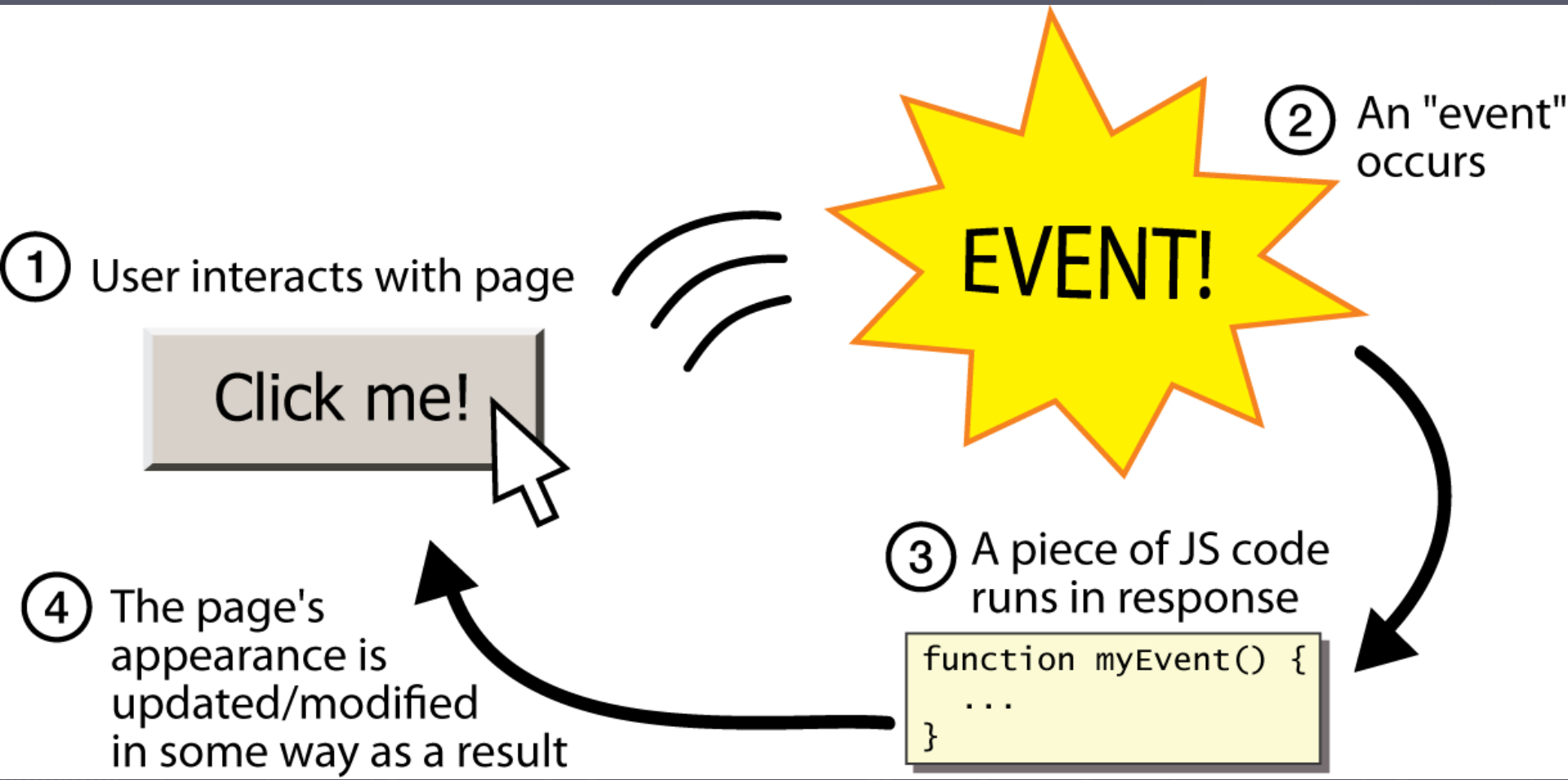
Uses of JavaScript

- Creating Drop down menus
- JavaScript is designed to add interactivity to HTML pages
- Adding Alert Messages
- Validating Html form data
- Reacting to events like clicking on the page with a mouse
- For storing and retrieving information on the visitors computer
- Creating popup windows

client-side scripting (JavaScript) benefits

- **usability**: can modify a page without having to post back to the server (faster UI)
- **efficiency**: can make small, quick changes to page without waiting for server
- **event-driven**: can respond to user actions like clicks and key presses

Event-driven programming



Are Java and JavaScript the Same?

- NO!
- Java and JavaScript are two completely different languages in both concept and design!
- Java (developed by Sun Microsystems) is a powerful and much more complex programming language - in the same category as C and C++.

Are Java and JavaScript the Same?

Sr. No.	JavaScript	Java
1)	JavaScript is an interpreted language(not compiled by client)	It is compiled (byte codes downloaded from server , executed on client)
2)	It is Object Based. Uses built-in, extensible objects, but no classes or inheritance.	It is Object-oriented. Applet consists of object classes with inheritance.
3)	It is loosely typed i.e. variable data types are not declared.	It is strongly typed i.e. variable data types must be declared.
4)	Dynamic binding i.e. object references checked at runtime.	Static binding i.e. object references must exist at compile-time.

How to Put a JavaScript Into an HTML Page?

1] Scripts in Head Section

Scripts to be executed when they are called, or when an event is triggered, go in the head section

```
<html>
<head>
<script type="text/javascript">
document.write("Hello World!")
</script>
</head>
<body>…..</body>
</html>
```

How to Put a JavaScript Into an HTML Page?

2] Scripts in Body Section

Scripts to be executed when the page loads go in the body section.

```
<html>
<body>
<script type="text/javascript">
document.write("Hello World!")
</script>
</body>
</html>
```

How to Put a JavaScript Into an HTML Page?

3] External JavaScripts

- External javascripts are used when we want to use same javascript on several pages without having to write same script on every page.
- Save the external javascript with a .js file extension.
- The external scripts cannot contain <script> tag.
- To use external javascript we use **src** attribute of the <script> tag to specify name of the script.

```
<script src="scriptname.js">
</script>
```

Ending Statements With a Semicolon?

- With traditional programming languages, like C++ and Java, each code statement has to end with a semicolon (;).
- Many programmers continue this habit when writing JavaScript, but in general, semicolons are **optional**! However, semicolons are required if you want to put more than one statement on a single line.

JavaScript Variables

- Variables are used to store data.
- A variable is a "container" for information you want to store. A variable's value can change during the script. You can refer to a variable by name to see its value or to change its value.
- Rules for variable names:
  - Variable names are case sensitive
  - They must begin with a letter or the underscore character
    - strname – STRNAME (not same)

Variables

```
var name = expression;
```

```
var clientName = "Connie Client";
var age = 32;
var weight = 127.4;
```

- variables are declared with the var keyword (case sensitive)
- types are not specified, but JS does have types ("loosely typed")
  - `Number, Boolean, String, Array, Object, Function, Null, Undefined`
  - can find out a variable's type by calling `typeof`

Number type

```
var enrollment = 99;
var medianGrade = 2.8;
var credits = 5 + 4 + (2 * 3);
```

JS

- integers and real numbers are the same type
- same operators: + - * / % ++ -- = += -= *= /= %=
- similar precedence to Java

Special values: null

```
var a = null;
var b = 9;
```

- `null` : exists, but was specifically assigned an empty or null value

Comments

```
1] // single-line comment



2] /* multi-line comment */
```

JavaScript Operators

Arithmetic Operators

Operator	Description	Example	Result
+	Addition	x=2 y=2 x+y	4
-	Subtraction	x=5 y=2 x-y	3
*	Multiplication	x=5 y=4 x*y	20
/	Division	15/5 5/2	3 2,5
%	Modulus (division remainder)	5%2 10%8 10%2	1 2 0

JavaScript Operators – 2

Assignment Operators

Operator	Example	Is The Same As
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
*=	x*=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

JavaScript Operators - 3

Comparison Operators

Operator	Description	Example
==	is equal to	5==8 returns false
===	is equal to (checks for both value and type)	x=5 y="5" x==y returns true x===y returns false
!=	is not equal	5!=8 returns true
>	is greater than	5>8 returns false
<	is less than	5<8 returns true
>=	is greater than or equal to	5>=8 returns false
<=	is less than or equal to	5<=8 returns true

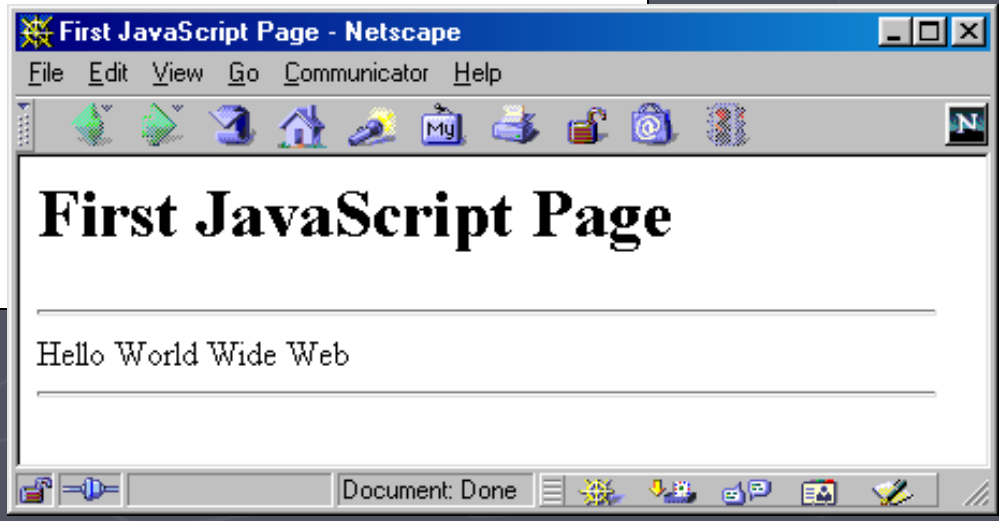
JavaScript Operators - 4

Logical Operators

Operator	Description	Example
&&	and	x=6 y=3 (x < 10 && y > 1) returns true
||	or	x=6 y=3 (x==5 || y==5) returns false
!	not	x=6 y=3 !(x==y) returns true

A Simple Script

```
<html>
<head><title>First JavaScript
Page</title></head>
<body>
<h1>First JavaScript Page</h1>
<script type="text/javascript">
document.write("<hr>");  document.write("Hello
World Wide Web");  document.write("<hr>");
</script>
</body>
</html>
```



Embedding JavaScript

```
<html>
<head><title>First JavaScript Program</title></head>
<body>
<script type="text/javascript"
src="your_source_file.js"></script>
</body>
```

Inside your source file.js

```
document.write("<hr>");
document.write("Hello World Wide Web");
document.write("<hr>");
```

- Use the `src` attribute to include JavaScript codes from an external file.
- The included code is inserted in place.

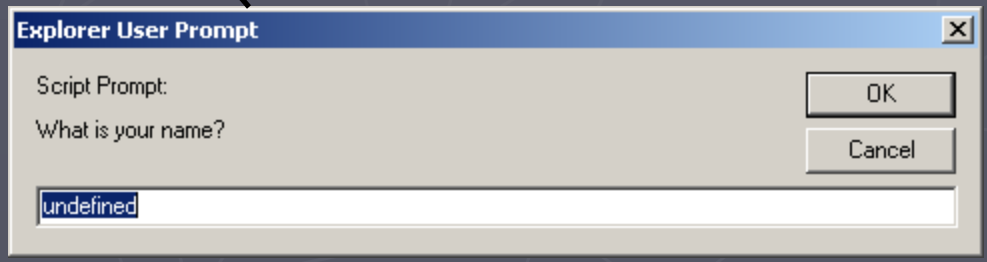
Example

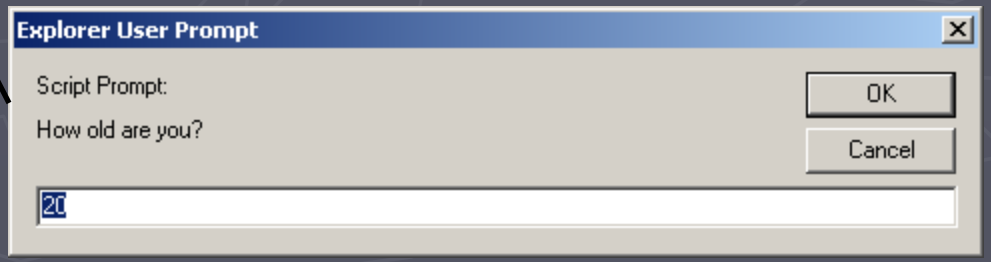
Write a Javascript to display value of variable.

```
<html>
<head></head>
<body>
<script type="text/Javascript">
x="Hello World!"
document.write(x)
</script>
</body>
</html>
```

alert(), confirm(), and prompt()

```
alert("This is an Alert method");
confirm("Are you OK?");
prompt("What is your name?");
prompt("How old are you?","20");
```





JavaScript Popup Boxes

- Alert Box
  - An alert box is often used if you want to make sure information comes through to the user.
  - When an alert box pops up, the user will have to click "OK" to proceed.

```
<script>
alert("Hello World!")
</script>
```

JavaScript Popup Boxes - 2

- Confirm Box
  - A confirm box is often used if you want the user to verify or accept something.
  - When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.
  - If the user clicks "OK", the box returns true. If the user clicks "Cancel", the box returns false.

JavaScript Popup Boxes - 3

- Prompt Box
  - A prompt box is often used if you want the user to input a value before entering a page.
  - When a prompt box pops up, the user will have to click either "OK" or "Cancel" to proceed after entering an input value.
  - If the user clicks "OK", the box returns the input value. If the user clicks "Cancel", the box returns null.

`alert()` and `confirm()`

```
alert("Text to be displayed");
```

Display a message in a dialog box.

The dialog box will block the browser.

```
var answer = confirm("Are you sure?");
```

Display a message in a dialog box with two buttons: "OK" or "Cancel".

`confirm()` returns `true` if the user click "OK". Otherwise it returns `false`.

prompt()

```
prompt("What is your student id number?");
prompt("What is your name?", "No name");
```

Display a message and allow the user to enter a value

The second argument is the "default value" to be displayed in the input textfield.

Without the default value, "undefined" is shown in the input textfield.

If the user click the "OK" button, **`prompt()`** returns the value in the input textfield as a string.

If the user click the "Cancel" button, **`prompt()`** returns null.