CUI Abbottabad

Department of Computer Science

Web Technologies

Lecture -10-[CLO-2]
Javascript-Part1

COMSATS University Islamabad, Abbottabad Campus

Agenda

- Introduction to Javascript
- Simple Javascript program
- Input/output statements
- Use of HTML in Javascript
- Variables
- Use of var, let and const
- Operators
- Memory concept
- Template string

INTRODUCTION

- **JavaScript**
  - Is a scripting language which is used to enhance the functionality and appearance of web pages.
  - Is used to program the behavior of web pages
  - Previously it was usually used as client-side scripting language but now it can be used for both client side and server side
  - Scripting language is also called interpreted languages(compilation and execution both done combine at runtime)
- **What JavaScript can do?**
  - It can change HTML contents(can change values of html elements)
  - It can change HTML attribute values
  - It can change HTML styles(CSS) dynamically
  - It can hide HTML elements
  - It can show HTML elements
- Firefox, Chrome, Opera, Safari (including on the iPhone) and the Android browser have JavaScript enabled by default.

INTRODUCTION

- Where to write JavaScript code?
  - In HTML, JavaScript code is inserted between <script>and</script> tags
  - Old JavaScript examples may use a type attribute:

    <script type="text/javascript">.
  - But now the type attribute is not required, because JavaScript is the default scripting language in HTML.
  - You can place any number of <script> tags in HTML document
  - <Script> tag can be placed in:
    - the <body> section, or
    - in the <head> section or
    - in both <body> and <head> section of html page

INTRODUCTION

External JavaScript

- JavaScript coding can be placed in separate External file
- External JavaScript files have the file extension .js
- External scripts are suitable to use when the same code is used in many different web pages
- External JavaScript file cannot contain <script> tags: only coding of JavaScript needed in External JavaScript file

How to use External JavaScript file in our HTML page?

- To use an external script, put the name of the script file in the src (source) attribute of a <script> tag
- Example:

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

- You can place an external script reference in <head> section or in <body> section, as you needed.
- To add several script files to one page - use several script tags

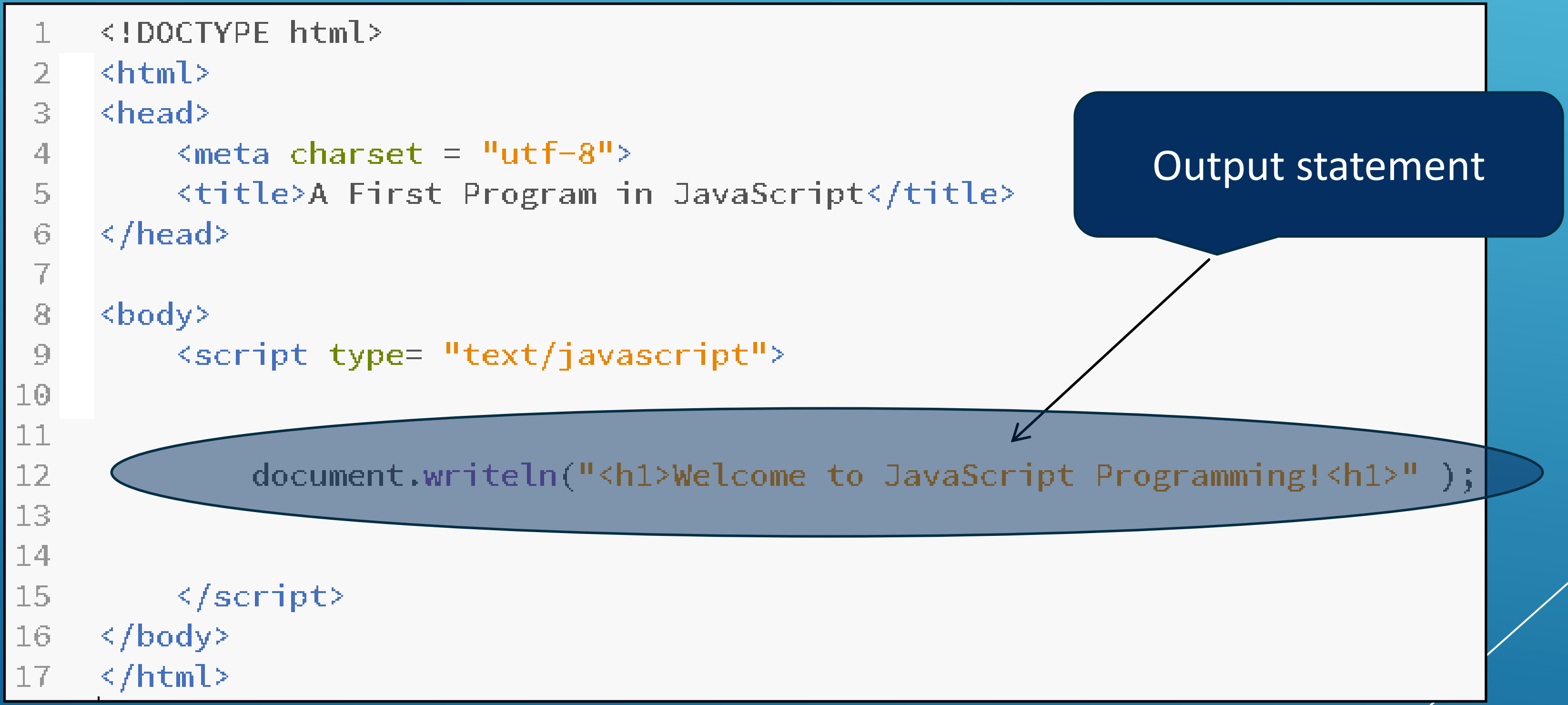
INTRODUCTION

Advantages of External JavaScript file

- It separates HTML and javascript code
- It makes HTML and JavaScript easier to read and maintain
- Cached JavaScript files can speed up page loads

FIRST JAVASCRIPT PROGRAM

- We begin with a simple script that displays the text "Welcome to JavaScript Programming!" in the HTML5 document.
- All advanced web browsers contain JavaScript **interpreters**, which process the commands written in JavaScript.
- The JavaScript code and its result are shown in Fig. 10.1.



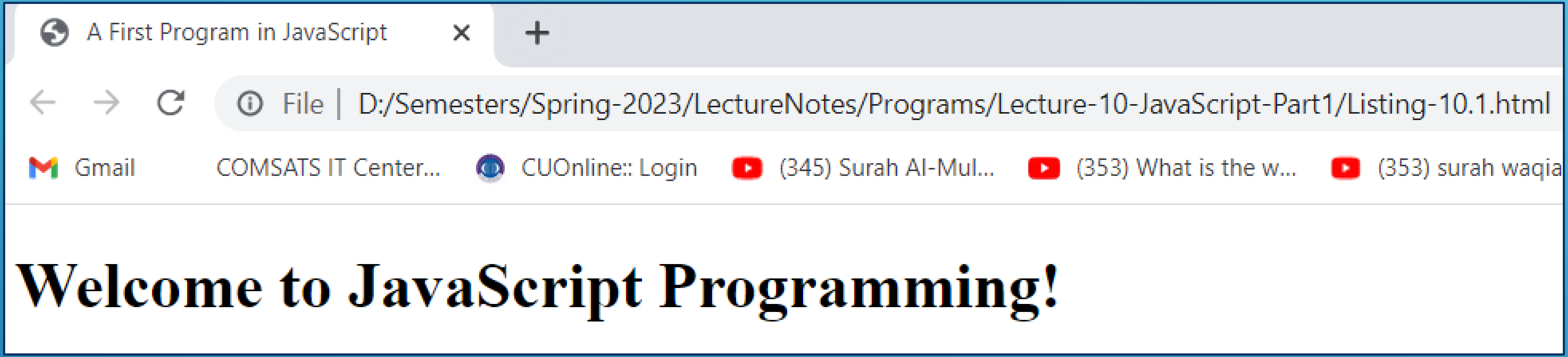
```
<!DOCTYPE html>
<html>
<head>
    <meta charset = "utf-8">
    <title>A First Program in JavaScript</title>
</head>

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


        document.writeln("<h1>Welcome to JavaScript Programming!<h1>" );


    </script>
</body>
</html>
```

Output statement

Listing-10.1.html

FIRST JAVASCRIPT PROGRAM

Output--------Listing 10.1


A First Program in JavaScript
File
D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.1.html
Gmail
COMSATS IT Center...
CUOnline:: Login
(345) Surah Al-Mul...
(353) What is the w...
(353) surah waqia

```
Welcome to JavaScript Programming!
```

Figure--10.1—Displaying a line of text

FIRST JAVASCRIPT PROGRAM

- Spacing displayed by a browser in a web page is determined by the HTML5 elements used to format the page
- Often, JavaScripts appear in the <head> section of the HTML5 document
- The browser interprets the contents of the <head> section first
- Line#9 <script> tag indicates to the browser that the text that follows is part of a script. Attribute type specifies the scripting language used in the script—such as text/javascript

The script Element and Commenting Your Scripts

- The <script> tag indicates to the browser that the text which follows is part of a script.
- The type attribute specifies the MIME type of the script as well as the **scripting language** used in the script—in this case, a text file written in javascript.
- In HTML5, the default MIME type for a <script> is "text/javascript", so you can omit the type attribute from your <script> tags.

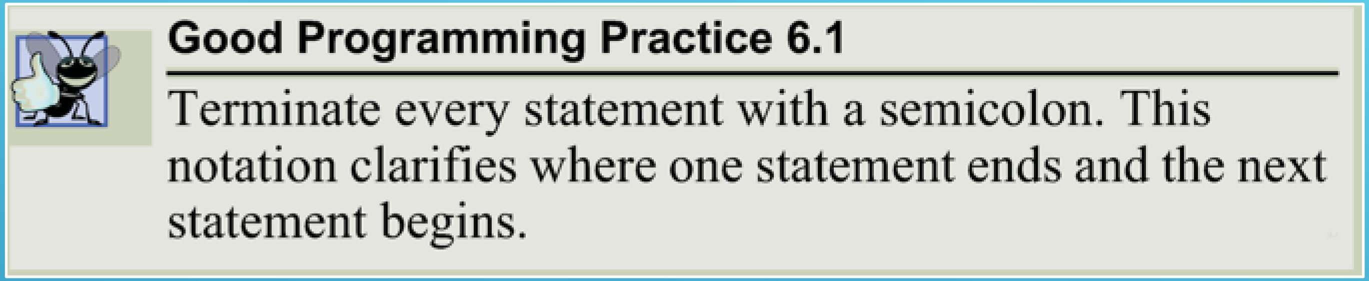
FIRST JAVASCRIPT PROGRAM

- In JavaScript strings can be contained between double quotation(" ") or single quotation (' ') marks (also called a **string literal**)
- Line#12: document.writeln("…");
- Browser's document object represents the HTML5 document currently being displayed in the browser
  - Allows a you to specify HTML5 text to be displayed in the HTML5 document
  - **All html elements can be used in inside writeln()/write()** ,Browser will interpret the HTML5 elements as it normally does to render the final text in the document
- Browser contains a complete set of objects that allow script programmers to access and manipulate every element of an HTML5 document

FIRST JAVASCRIPT PROGRAM

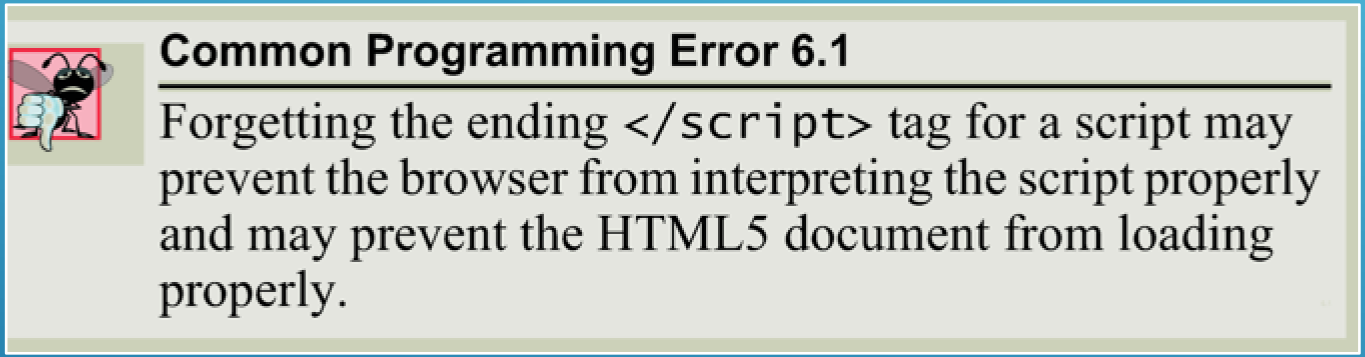
- Object
  - Resides in the computer’s memory and contains information used by the script
  - The term object normally implies that attributes (data) and behaviors (methods) are associated with the object
  - An object’s methods use the attributes’ data to perform useful actions for the client of the object—the script that calls the methods
- The parentheses following the name of a method contain the arguments that the method requires to perform its task (or its action)
- Every statement should end with a semicolon (also known as the **statement terminator**), although none is required by JavaScript
- JavaScript is case sensitive
  - Not using the proper uppercase and lowercase letters is a syntax error

FIRST JAVASCRIPT PROGRAM



Good Programming Practice 6.1

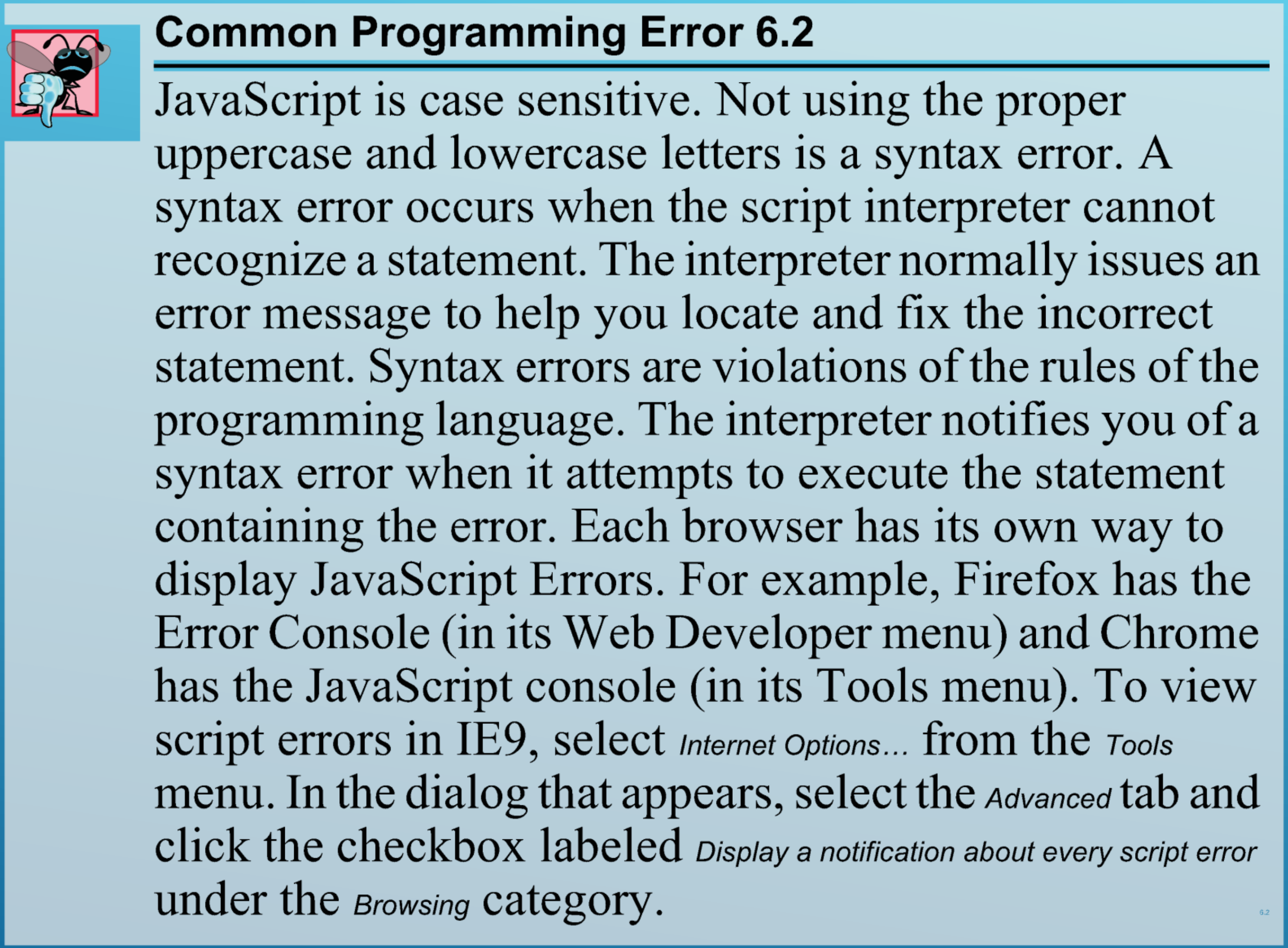
Terminate every statement with a semicolon. This notation clarifies where one statement ends and the next statement begins.



Common Programming Error 6.1

Forgetting the ending `</script>` tag for a script may prevent the browser from interpreting the script properly and may prevent the HTML5 document from loading properly.

FIRST JAVASCRIPT PROGRAM

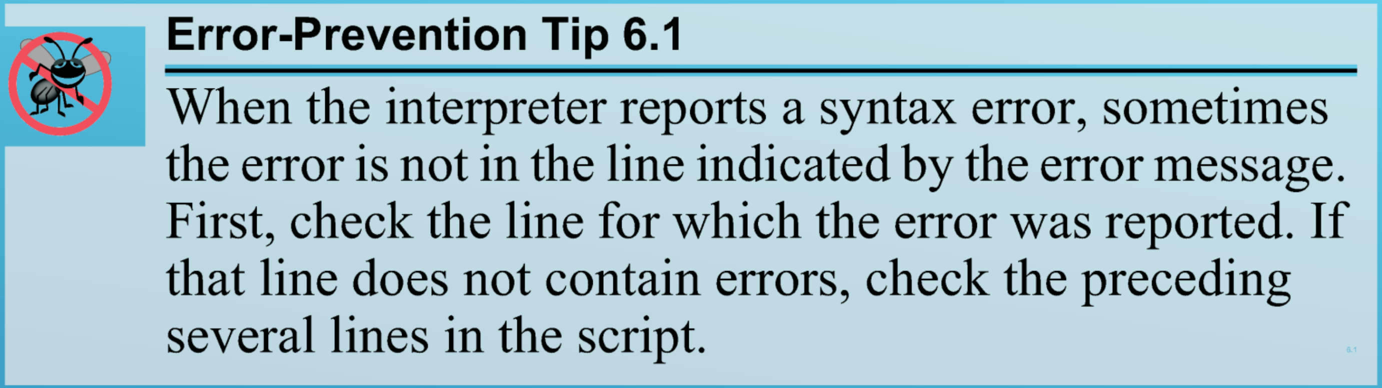


Common Programming Error 6.2

JavaScript is case sensitive. Not using the proper uppercase and lowercase letters is a syntax error. A syntax error occurs when the script interpreter cannot recognize a statement. The interpreter normally issues an error message to help you locate and fix the incorrect statement. Syntax errors are violations of the rules of the programming language. The interpreter notifies you of a syntax error when it attempts to execute the statement containing the error. Each browser has its own way to display JavaScript Errors. For example, Firefox has the Error Console (in its Web Developer menu) and Chrome has the JavaScript console (in its Tools menu). To view script errors in IE9, select *Internet Options…* from the *Tools* menu. In the dialog that appears, select the *Advanced* tab and click the checkbox labeled *Display a notification about every script error* under the *Browsing* category.

6.2

FIRST JAVASCRIPT PROGRAM



Error-Prevention Tip 6.1

When the interpreter reports a syntax error, sometimes the error is not in the line indicated by the error message. First, check the line for which the error was reported. If that line does not contain errors, check the preceding several lines in the script.

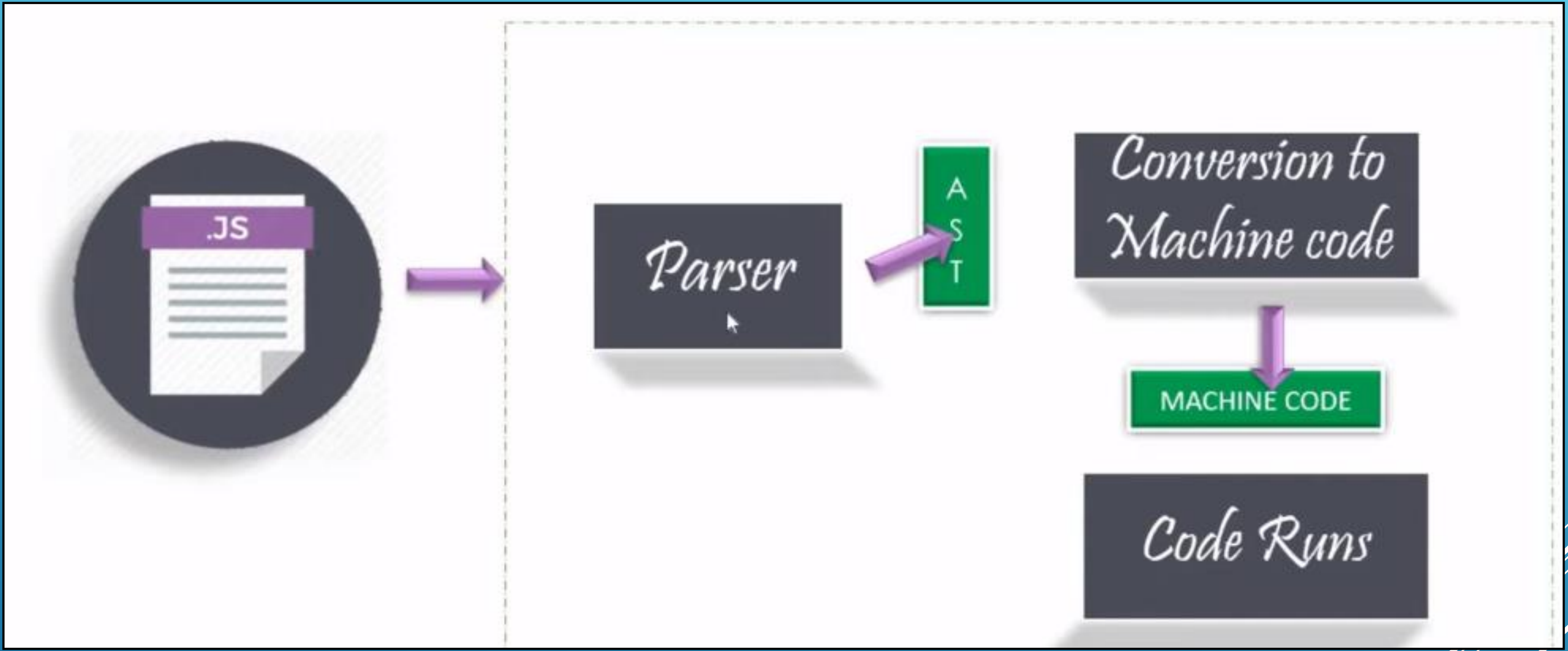
6.1

FIRST JAVASCRIPT PROGRAM

- A Note About Embedding JavaScript Code into HTML5 Documents
- JavaScript code is typically placed in a separate file, then included in the HTML5 document that uses the script.
- This makes the code more reusable, because it can be included into any HTML5 document—as is the case with the many JavaScript libraries used in professional web development today.
- We'll begin separating both CSS3 and JavaScript into separate files starting in next lectures.

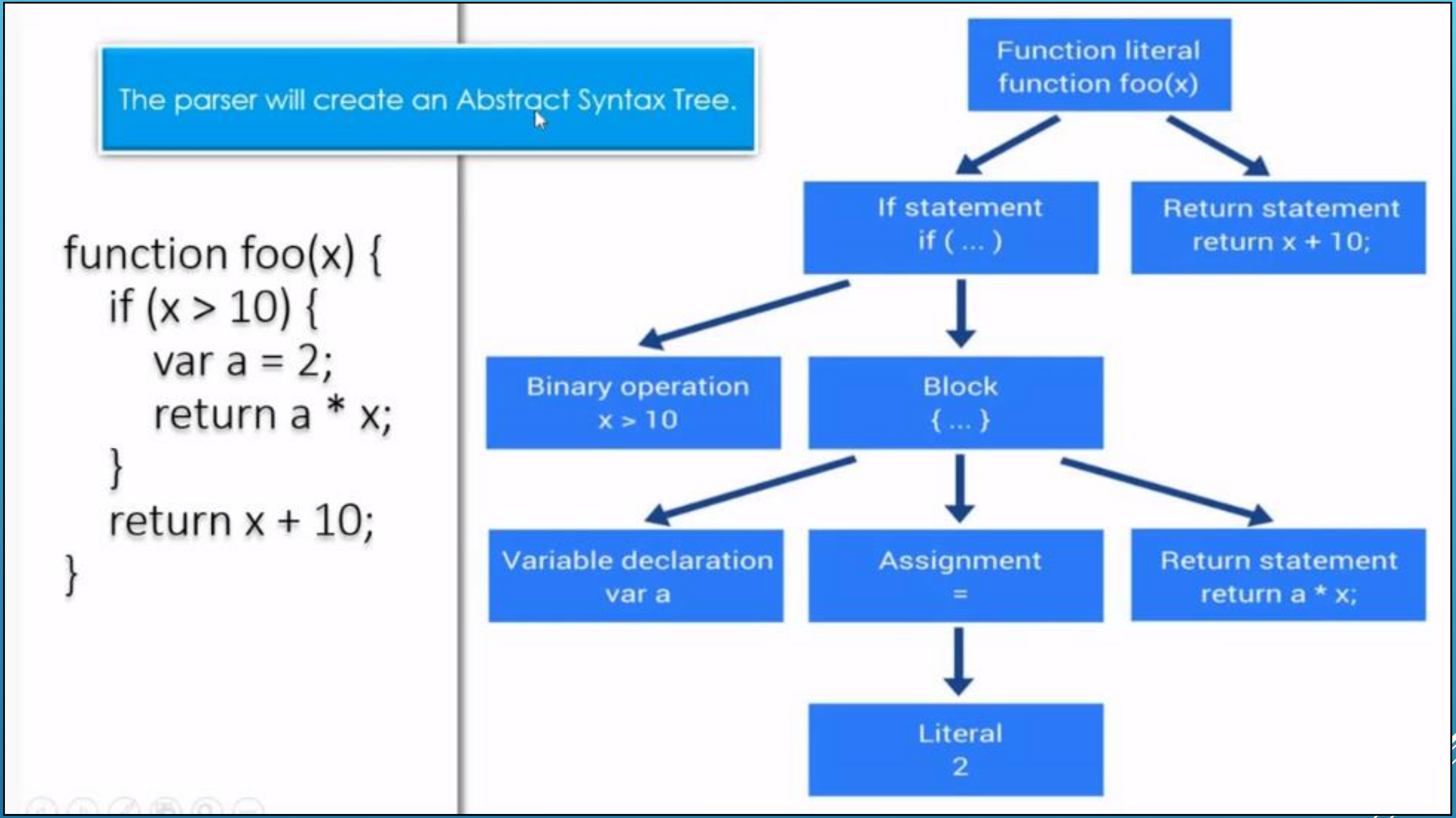
BROWSER SUPPORT FOR JAVASCRIPT

HOW JAVASCRIPT WORKS



AST--Abstract Syntax tree

HOW JAVASCRIPT WORKS



The parser will create an Abstract Syntax Tree.

```
function foo(x) {
   if (x > 10) {
      var a = 2;
      return a * x;
   }
   return x + 10;
}
```

Function literal
function foo(x)

If statement
if (...)

Return statement
return x + 10;

Binary operation
x > 10

Block
{ ... }

Variable declaration
var a

Assignment
=

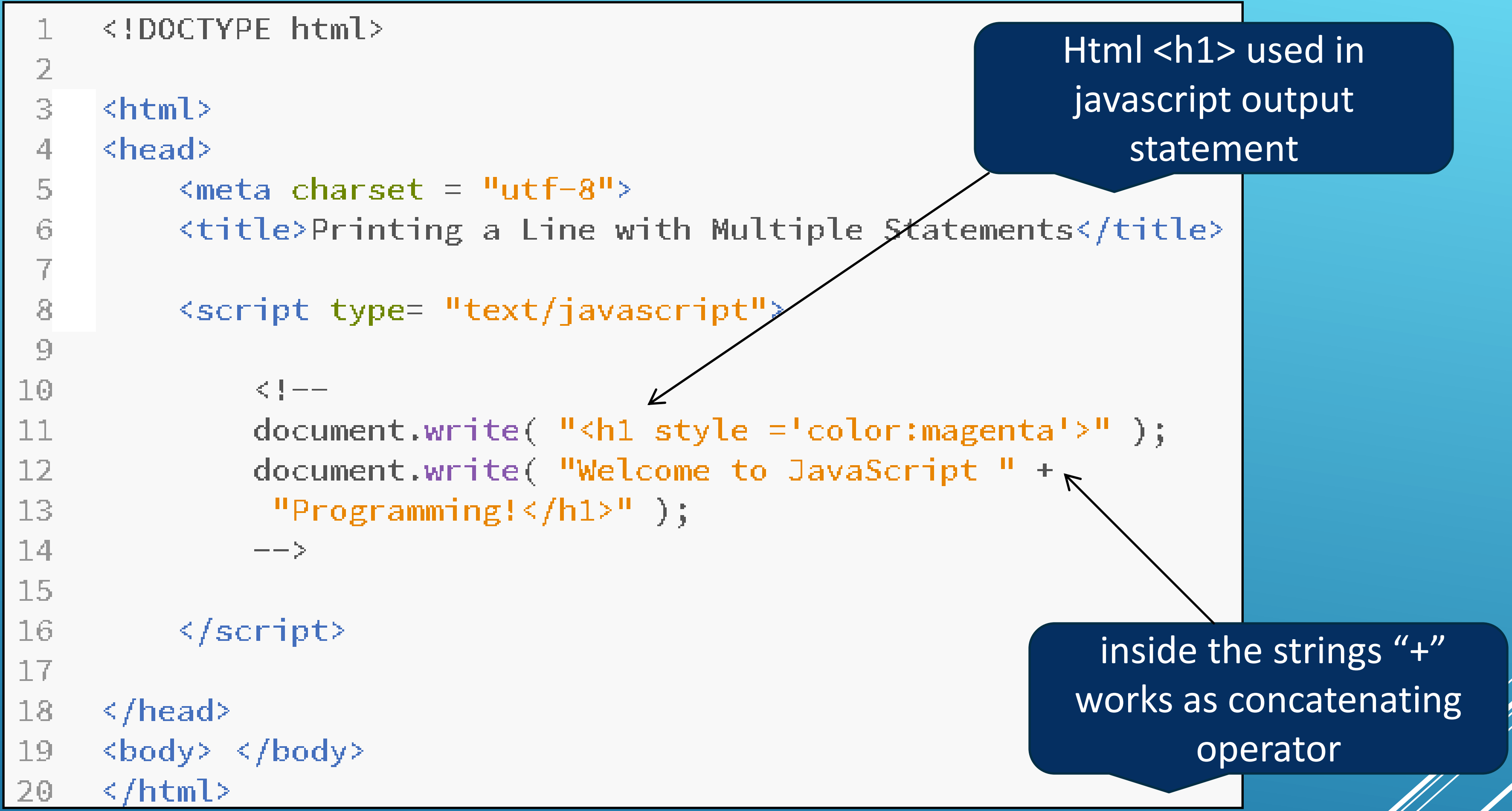
Return statement
return a * x;

Literal
2

MODIFY FIRST JAVASCRIPT PROGRAM

- A script can display Welcome to JavaScript Programming! in many ways.
- Figure 10.2 displays the text in magenta, using the CSS color property.
- Method write displays a string like writeln, but does not position the output cursor in the HTML5 document at the beginning of the next line after writing its argument

MODIFY FIRST JAVASCRIPT PROGRAM



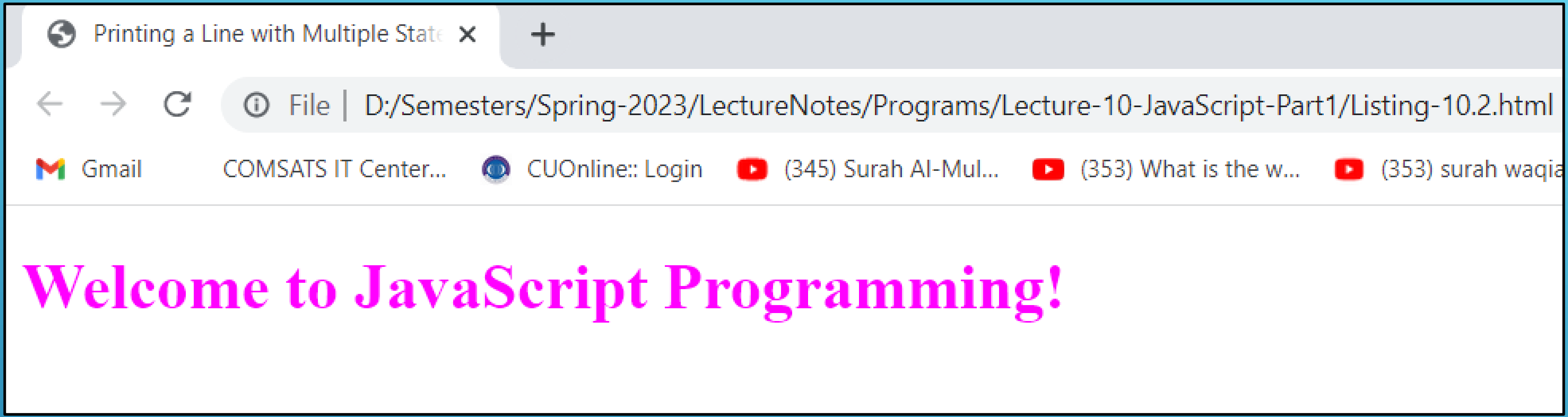
```
<!DOCTYPE html>

<html>
<head>
    <meta charset = "utf-8">
    <title>Printing a Line with Multiple Statements</title>

    <script type= "text/javascript">

        <!--
        document.write( "<h1 style ='color:magenta'>" );
        document.write( "Welcome to JavaScript " +
         "Programming!</h1>" );
        -->

    </script>

</head>
<body> </body>
</html>
```

Html <h1> used in javascript output statement

inside the strings “+” works as concatenating operator

Figure--10.2—use of html elements in Javascript

MODIFY FIRST JAVASCRIPT PROGRAM



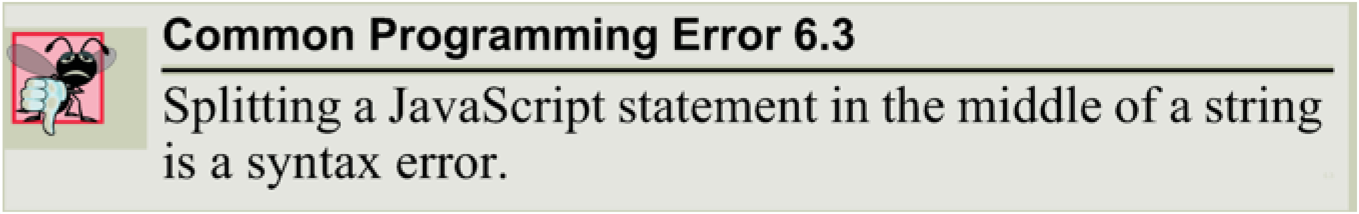
Printing a Line with Multiple State
File | D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.2.html
Gmail
COMSATS IT Center...
CUOnline:: Login
(345) Surah Al-Mul...
(353) What is the w...
(353) surah waqia

Welcome to JavaScript Programming!

Figure--10.2—Printing one line with separate statements

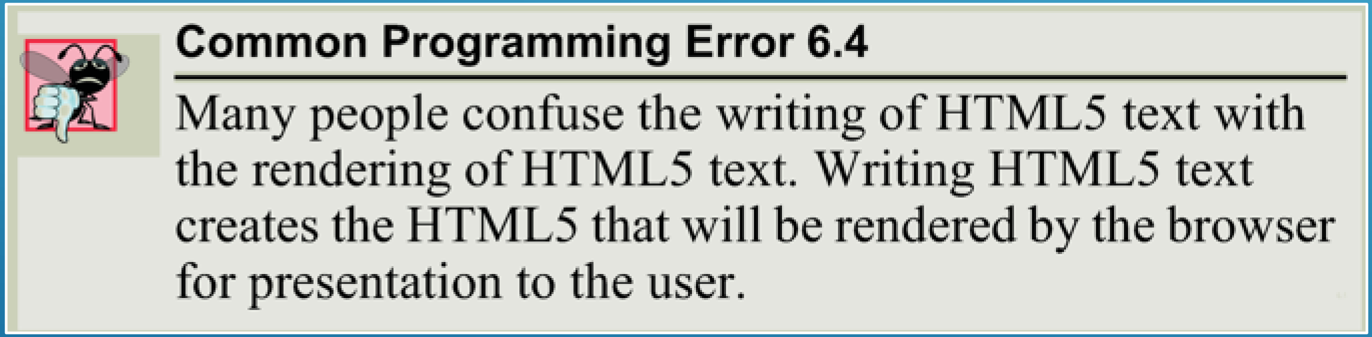
MODIFY FIRST JAVASCRIPT PROGRAM

- Line#12--The + operator (called the “concatenation operator” when used in this manner) joins two strings together



Common Programming Error 6.3

Splitting a JavaScript statement in the middle of a string is a syntax error.



Common Programming Error 6.4

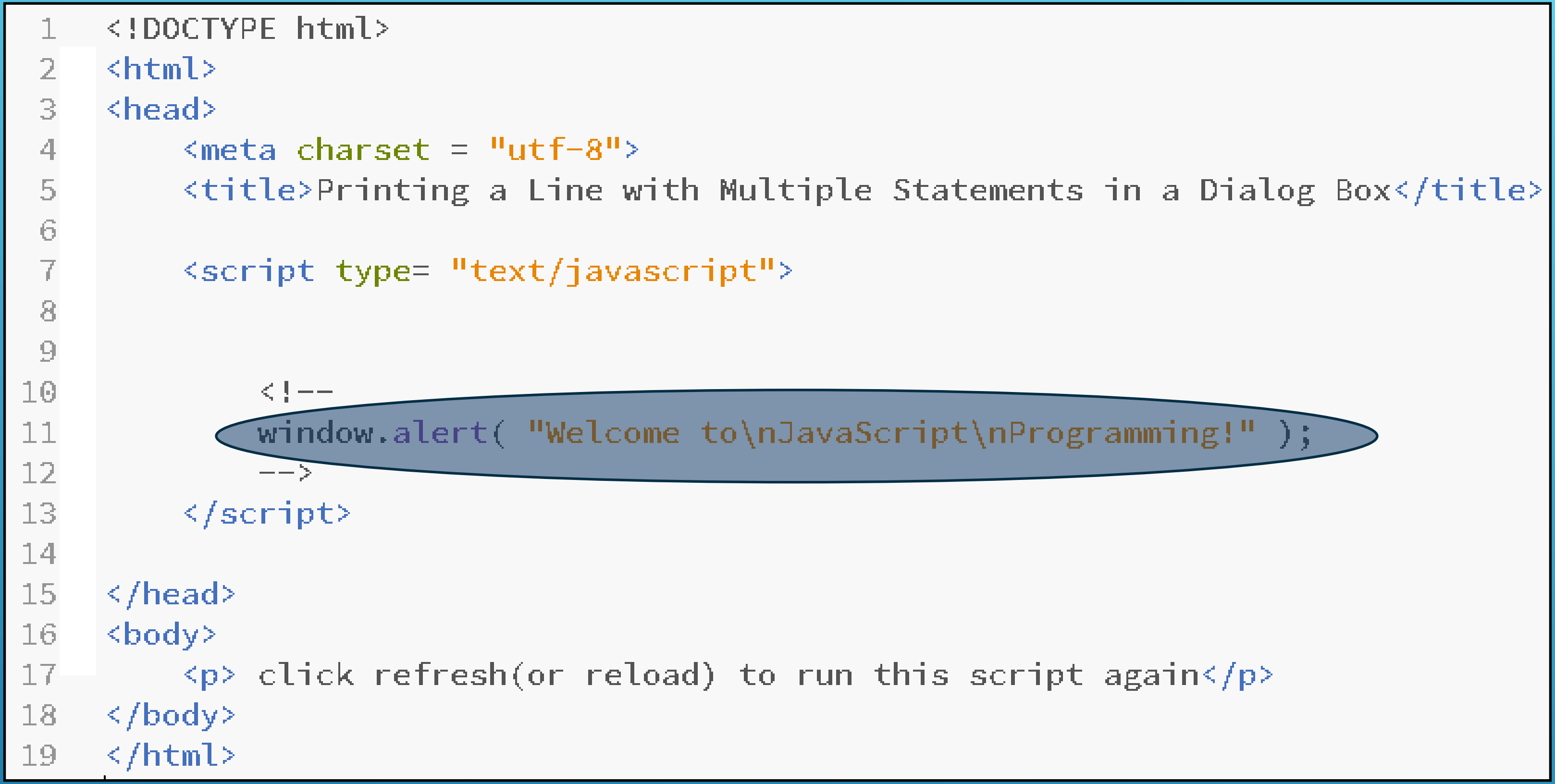
Many people confuse the writing of HTML5 text with the rendering of HTML5 text. Writing HTML5 text creates the HTML5 that will be rendered by the browser for presentation to the user.

MODIFY FIRST JAVASCRIPT PROGRAM

Displaying Text in an Alert Dialog

- Dialogs
  - Useful to display information in windows that “pop up” on the screen to grab the user’s attention
  - Typically used to display important messages to the user browsing the web page
  - Browser’s window object uses method alert to display an alert dialog
  - In method alert we requires a string as its argument to be displayed

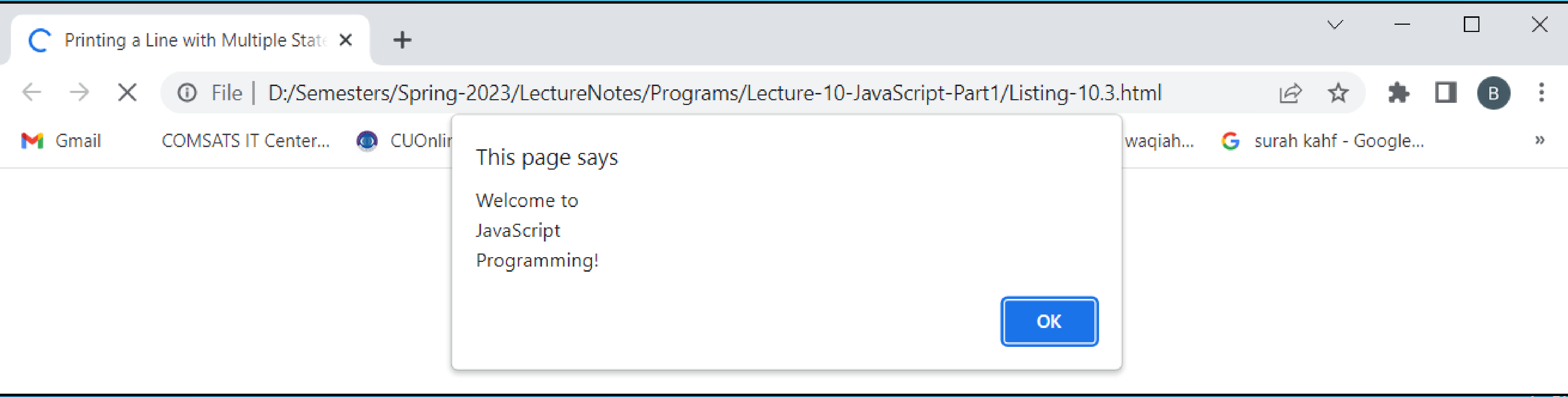
ALERT DIALOG IN JAVASCRIPT



```
<!DOCTYPE html>
<html>
<head>
    <meta charset = "utf-8">
    <title>Printing a Line with Multiple Statements in a Dialog Box</title>

    <script type= "text/javascript">


        <!--
        window.alert( "Welcome to\nJavaScript\nProgramming!" );
        -->
    </script>

</head>
<body>
    <p> click refresh(or reload) to run this script again</p>
</body>
</html>
```

Listing—10.3—Use of windows alert dialog using javascript

ALERT DIALOG IN JAVASCRIPT


Printing a Line with Multiple Stat
File
D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.3.html
Gmail
COMSATS IT Center...
CUOnlir
waqiah...
surah kahf - Google...
OK

This page says

Welcome to
JavaScript
Programming!

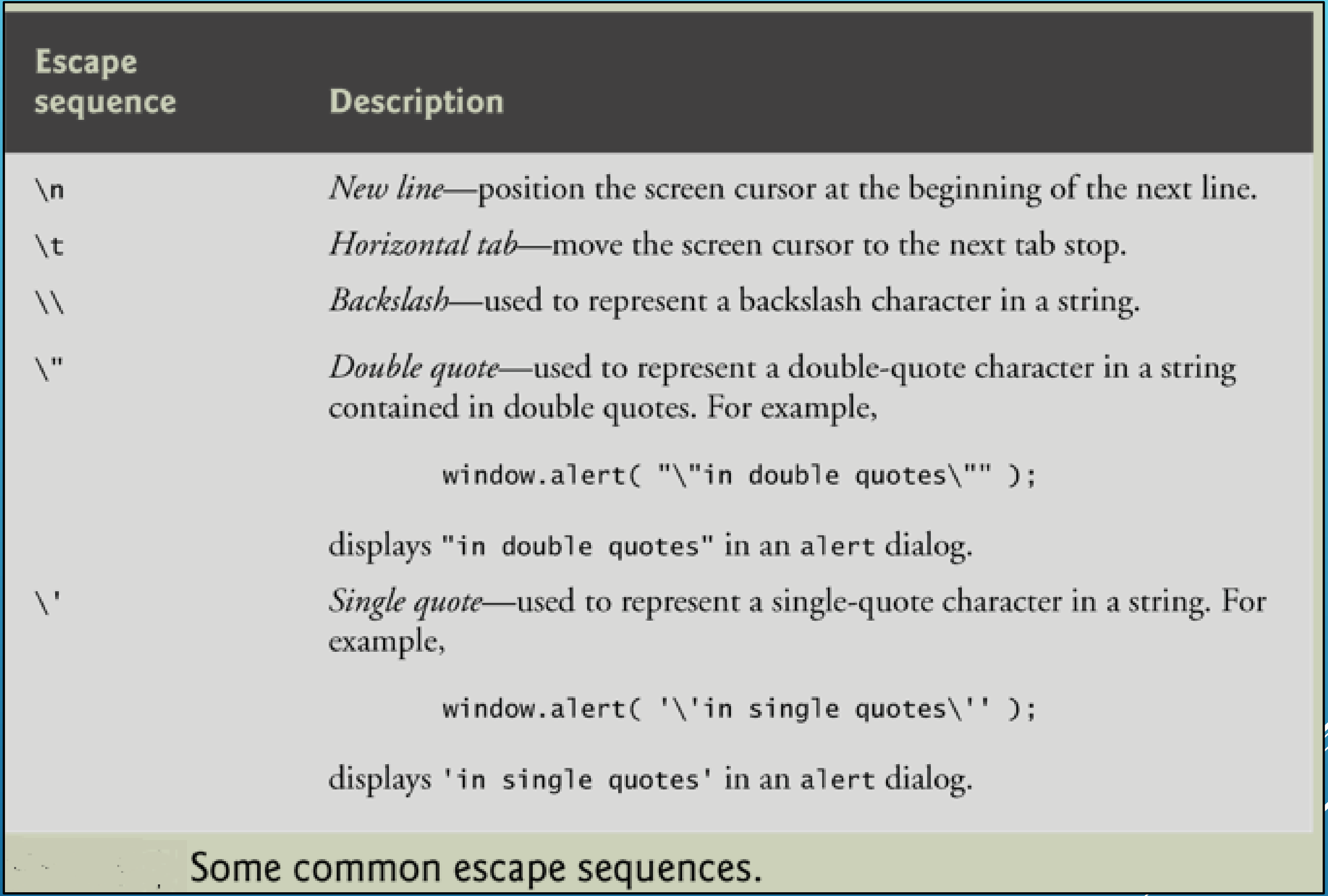
Figure--10.3—Use of windows alert dialog using javascript

ESCAPE SEQUENCE IN JAVASCRIPT

Escape Sequences

- In Listing-10.3 line#11 When a backslash is encountered in a string of characters, the next character is combined with the backslash to form an **escape sequence**. The escape sequence **\n** is the **newline character**. It causes the cursor in the HTML5 document to move to the beginning of the next line.

ESCAPE SEQUENCE IN JAVASCRIPT



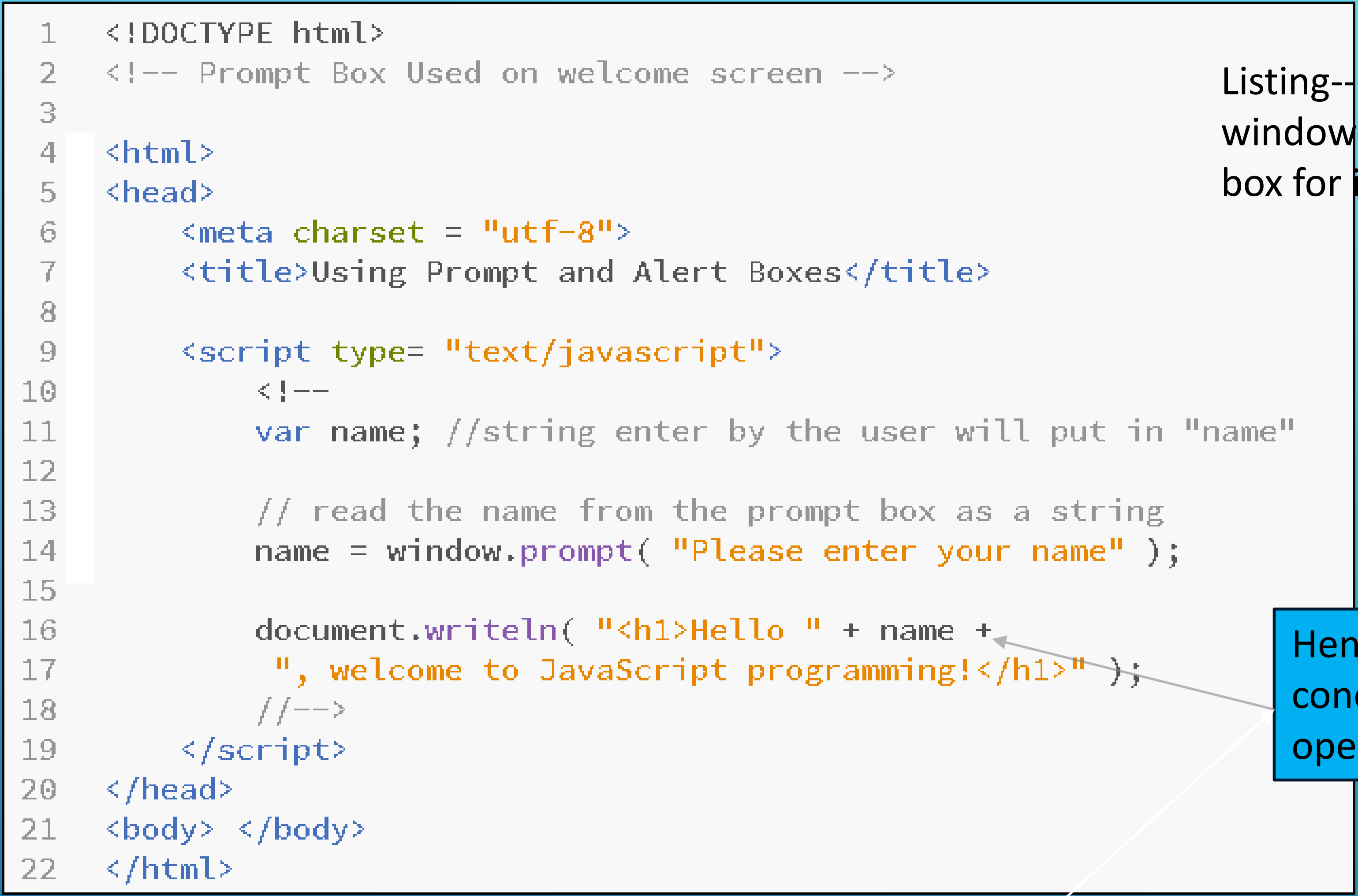
Escape sequence	Description
`\n`	*New line*—position the screen cursor at the beginning of the next line.
`\t`	*Horizontal tab*—move the screen cursor to the next tab stop.
`\\`	*Backslash*—used to represent a backslash character in a string.
`\"`	*Double quote*—used to represent a double-quote character in a string contained in double quotes. For example, `window.alert( "\"in double quotes\"" );` displays `"in double quotes"` in an `alert` dialog.
`\'`	*Single quote*—used to represent a single-quote character in a string. For example, `window.alert( '\'in single quotes\'' );` displays `'in single quotes'` in an `alert` dialog.

Some common escape sequences.

DYNAMIC WELCOME PAGE

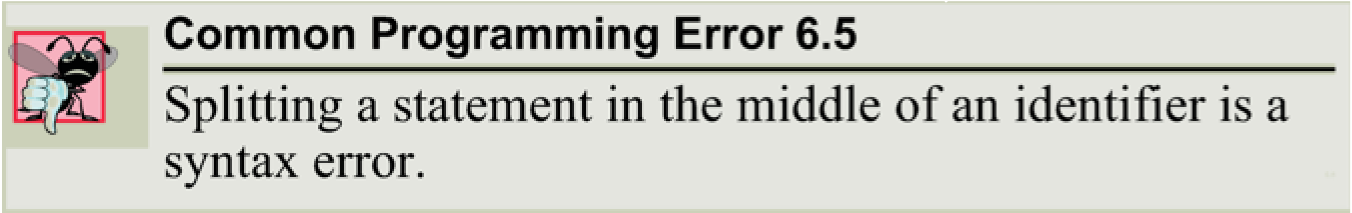
- The next script creates a dynamic welcome page that obtains the user’s name, then displays it on the page.
- The script uses another *predefined* dialog box from the window object—a **prompt** dialog—which allows the user to enter a value that the script can use.
- Listing 10.5 presents the script and sample output.

Listing--10.5—Use of windows prompt dialog box for input values



```html
<!DOCTYPE html>
<!-- Prompt Box Used on welcome screen -->

<html>
<head>
    <meta charset = "utf-8">
    <title>Using Prompt and Alert Boxes</title>

    <script type= "text/javascript">
        <!--
        var name; //string enter by the user will put in "name"

        // read the name from the prompt box as a string
        name = window.prompt( "Please enter your name" );

        document.writeln( "<h1>Hello " + name +
         ", welcome to JavaScript programming!</h1>" );
        //-->
    </script>
</head>
<body> </body>
</html>
```

Hence use concatenating + operator

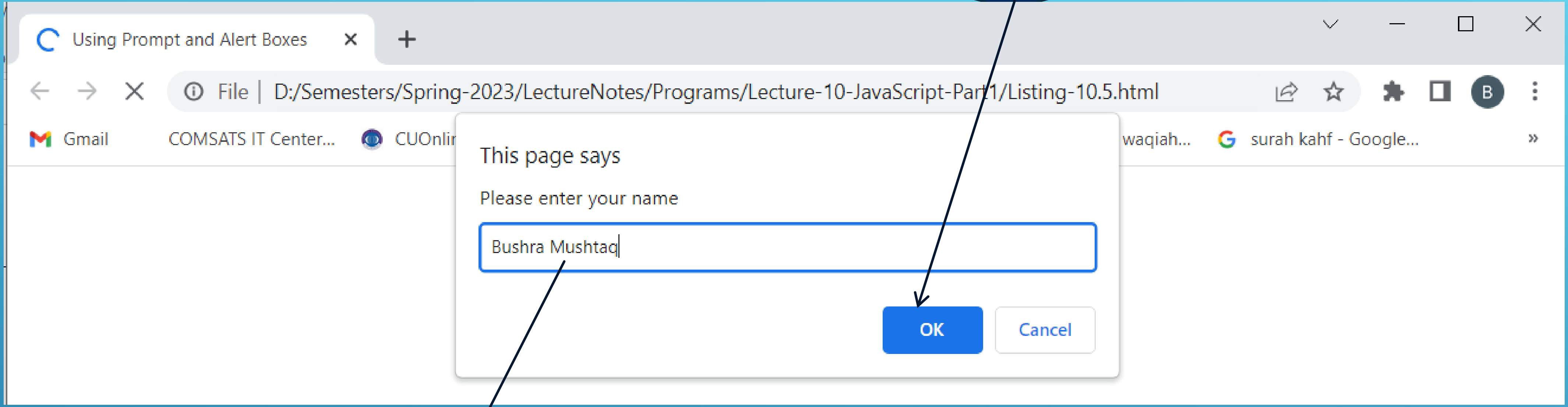


Common Programming Error 6.5

Splitting a statement in the middle of an identifier is a syntax error.

PROMPT DIALOG BOX

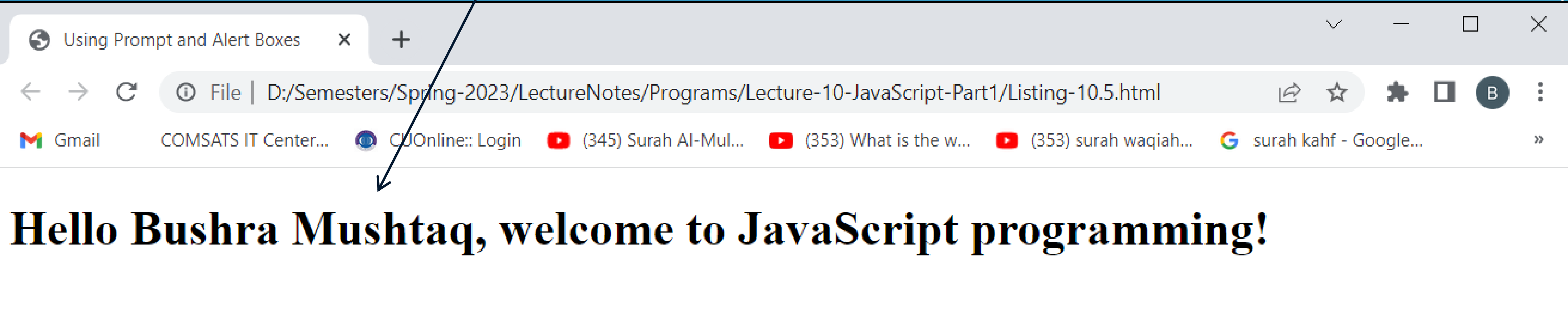
When “ok” clicked then message will appear with name as entered in dialog box


Using Prompt and Alert Boxes
File | D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.5.html
Gmail
COMSATS IT Center...
CUOnlir
waqiah...
surah kahf - Google...
OK
Cancel

This page says

Please enter your name

Bushra Mushtaq


Using Prompt and Alert Boxes
File | D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.5.html
Gmail
COMSATS IT Center...
CUOnline:: Login
(345) Surah Al-Mul...
(353) What is the w...
(353) surah waqiah...
surah kahf - Google...

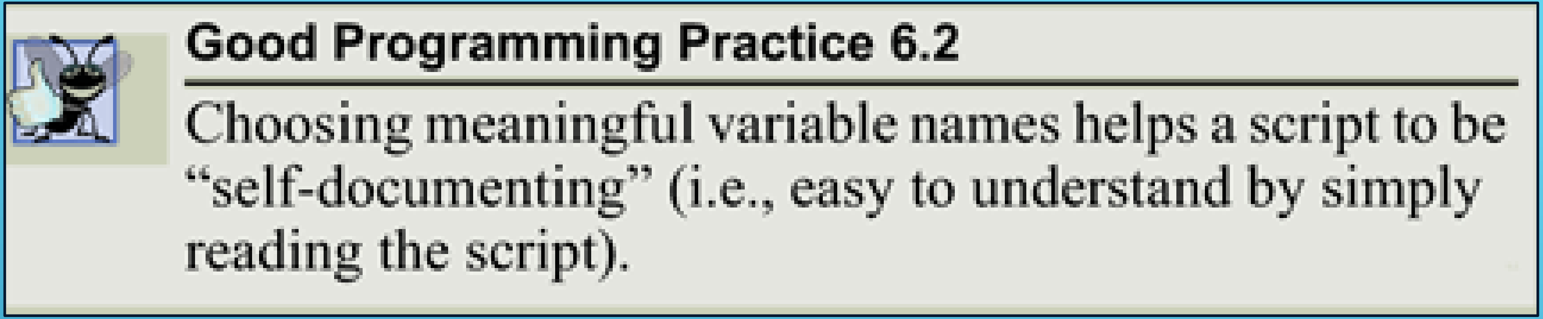
Hello Bushra Mushtaq, welcome to JavaScript programming!

Output--10.5—Use of windows prompt dialog box for input values

PROMPT DIALOG BOX

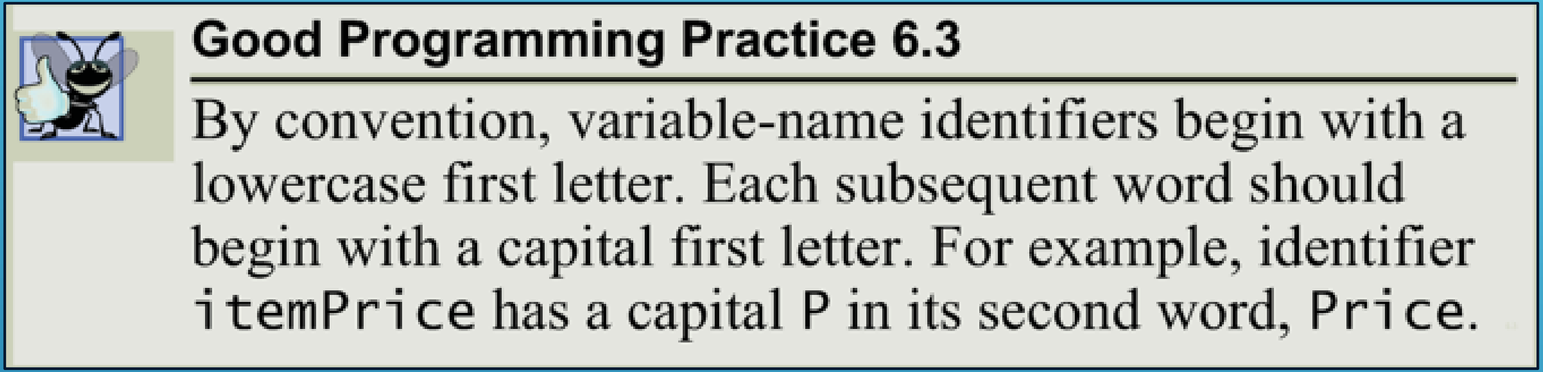
Listing—10.5

- **Keywords** are words with special meaning in JavaScript
- Keyword var at line#11
  - Used to declare the names of variables
  - A variable is a location in the computer's memory where a value can be stored for use by a script
  - All variables have a name, type and value, and should be declared with a var statement before they are used in a script
- A variable name can be any valid **identifier** consisting of letters, digits, underscores (_) and dollar signs ($) that does not begin with a digit and is not a reserved JavaScript keyword.



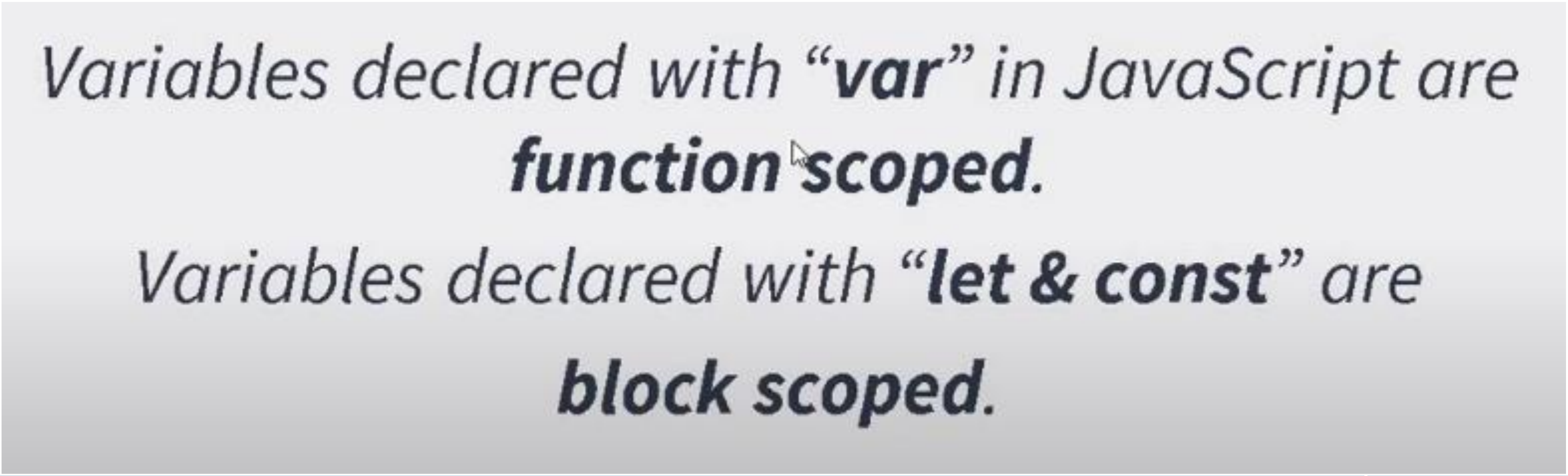
Good Programming Practice 6.2

Choosing meaningful variable names helps a script to be "self-documenting" (i.e., easy to understand by simply reading the script).



Good Programming Practice 6.3

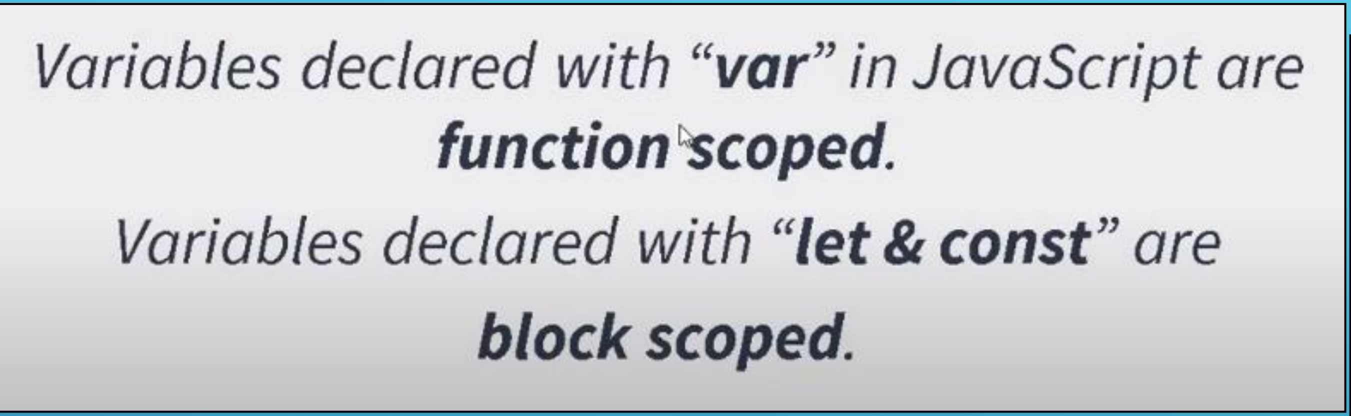
By convention, variable-name identifiers begin with a lowercase first letter. Each subsequent word should begin with a capital first letter. For example, identifier `itemPrice` has a capital `P` in its second word, `Price`.



*Variables declared with "**var**" in JavaScript are **function scoped**.*

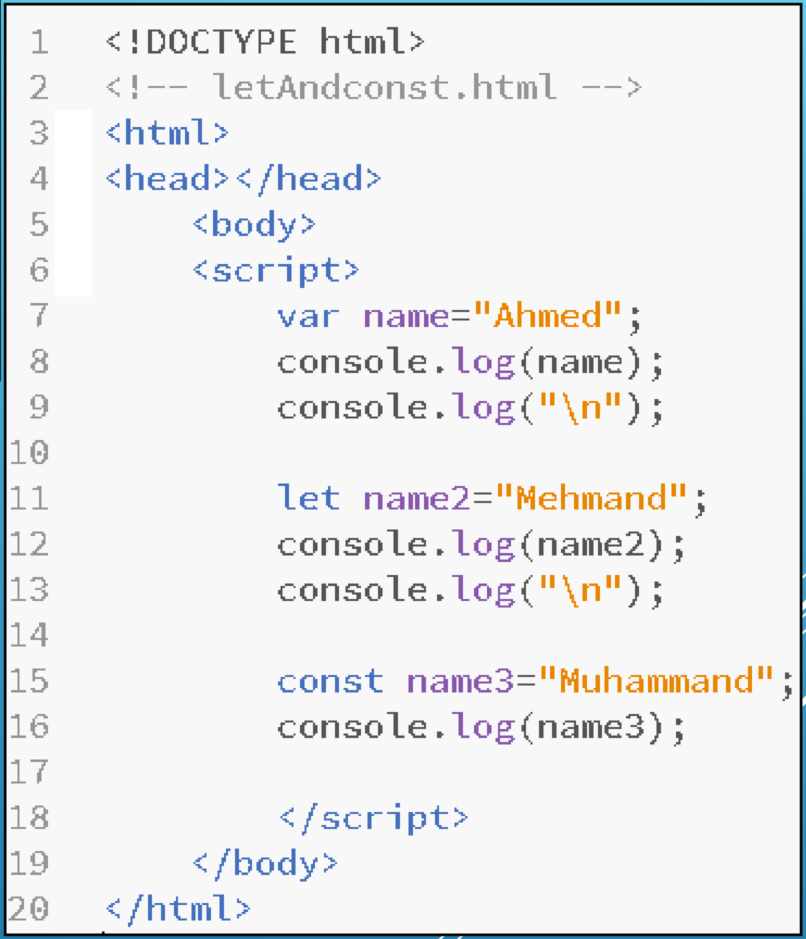
*Variables declared with "**let & const**" are **block scoped**.*

VAR, LET AND CONST



*Variables declared with “**var**” in JavaScript are **function scoped.***

*Variables declared with “**let & const**” are **block scoped.***

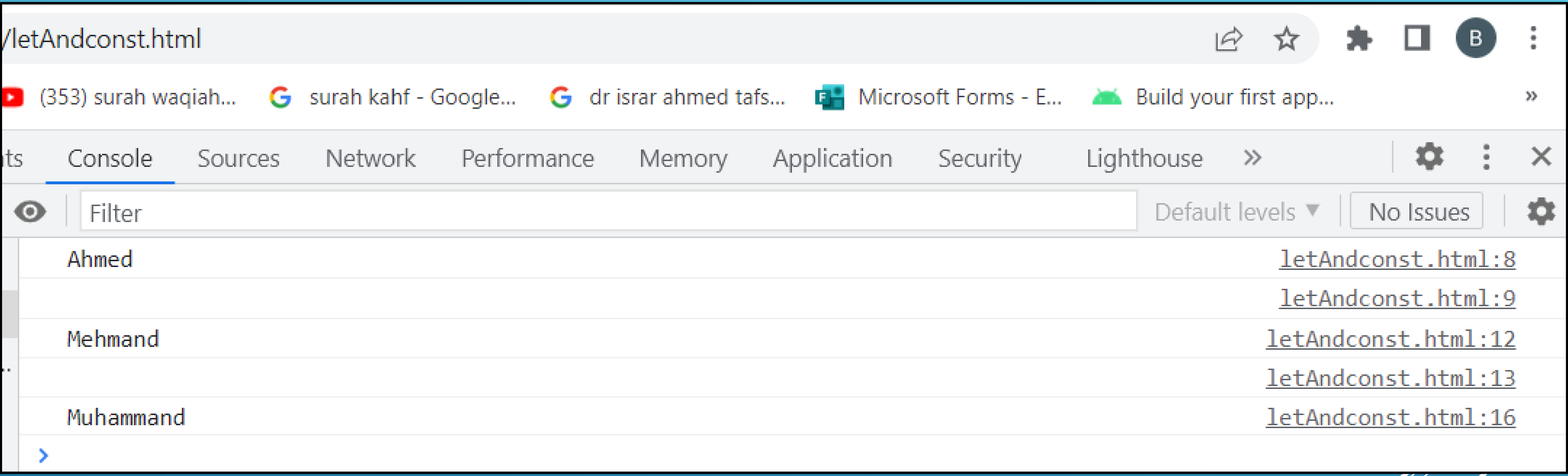


```
<!DOCTYPE html>
<!-- letAndconst.html -->
<html>
<head></head>
    <body>
    <script>
        var name="Ahmed";
        console.log(name);
        console.log("\n");

        let name2="Mehmand";
        console.log(name2);
        console.log("\n");

        const name3="Muhammand";
        console.log(name3);

        </script>
    </body>
</html>
```

letAndconst.html

COMSATS University Islamabad, Abbottabad Campus

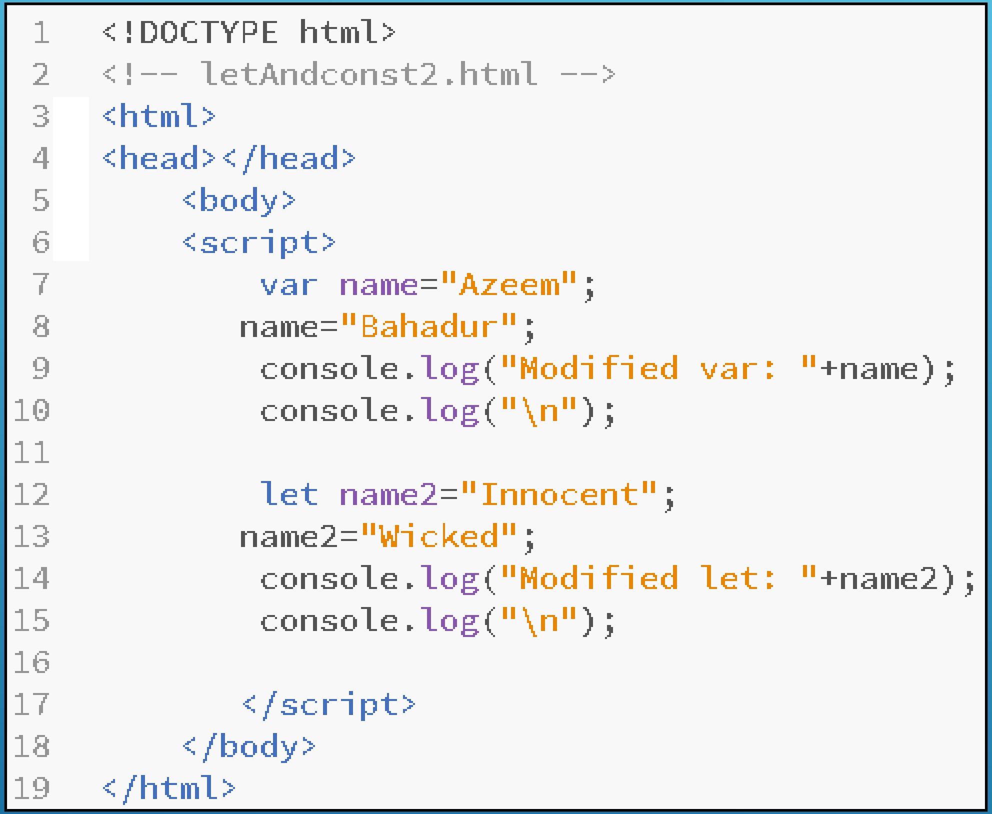
VAR, LET AND CONST



```
Ahmed                                                    letAndconst.html:8
                                                         letAndconst.html:9
Mehmand                                                  letAndconst.html:12
                                                         letAndconst.html:13
Muhammand                                                letAndconst.html:16
>
```

Output-letAndconst.html

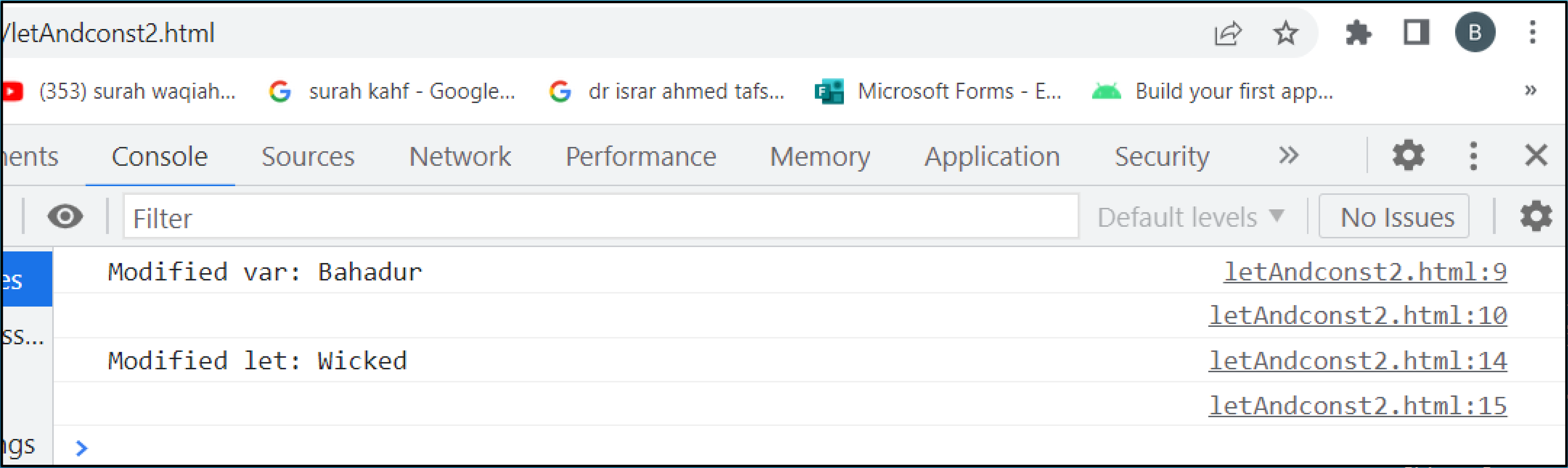
VAR, LET AND CONST….



```
<!DOCTYPE html>
<!-- letAndconst2.html -->
<html>
<head></head>
    <body>
    <script>
        var name="Azeem";
       name="Bahadur";
        console.log("Modified var: "+name);
        console.log("\n");

        let name2="Innocent";
       name2="Wicked";
        console.log("Modified let: "+name2);
        console.log("\n");

       </script>
    </body>
</html>
```

letAndconst2.html

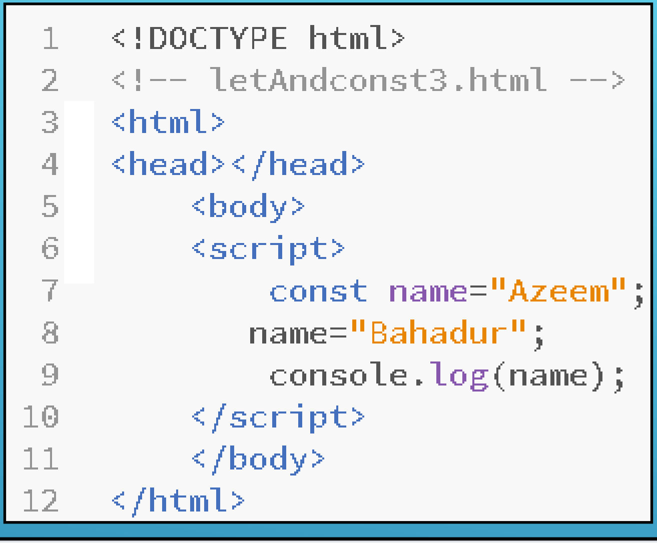
VAR, LET AND CONST….



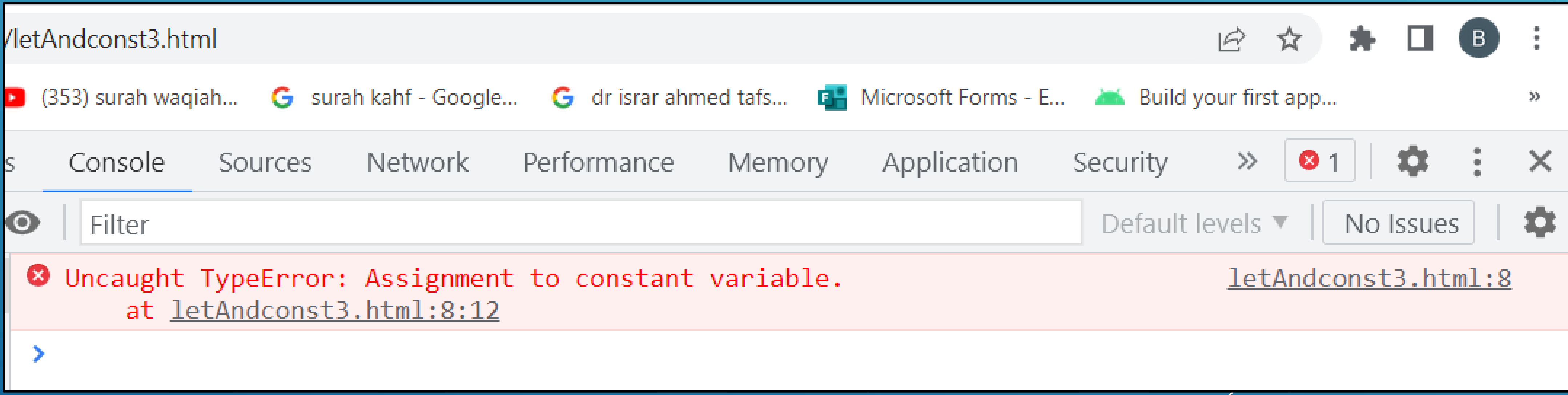
```
Modified var: Bahadur                         letAndconst2.html:9
                                              letAndconst2.html:10
Modified let: Wicked                          letAndconst2.html:14
                                              letAndconst2.html:15
>
```

Output-letAndconst2.html

VAR, LET AND CONST



```html
<!DOCTYPE html>
<!-- letAndconst3.html -->
<html>
<head></head>
    <body>
    <script>
        const name="Azeem";
       name="Bahadur";
        console.log(name);
    </script>
    </body>
</html>
```



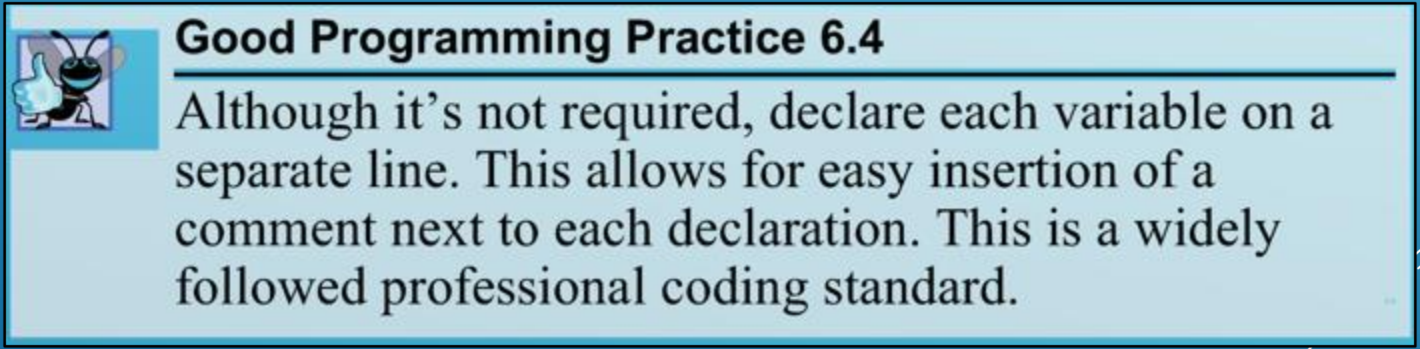
```
Uncaught TypeError: Assignment to constant variable.        letAndconst3.html:8
    at letAndconst3.html:8:12
>
```

Output-letAndconst3.html

COMSATS University Islamabad, Abbottabad Campus

COMMENTS IN JAVASCRIPT

- Declarations end with a semicolon (;) and can be split over several lines, with each variable in the declaration separated by a comma (forming a comma-separated list of variable names)
  - Several variables may be declared in one declaration or in multiple declarations.
- **Comments**
  - A single-line comment begins with the characters // and terminates at the end of the line
  - Comments do not cause the browser to perform any action when the script is interpreted; rather, comments are ignored by the JavaScript interpreter
  - Multiline comments begin with delimiter /* and end with delimiter */
  - All text between the delimiters of the comment is ignored by the interpreter.



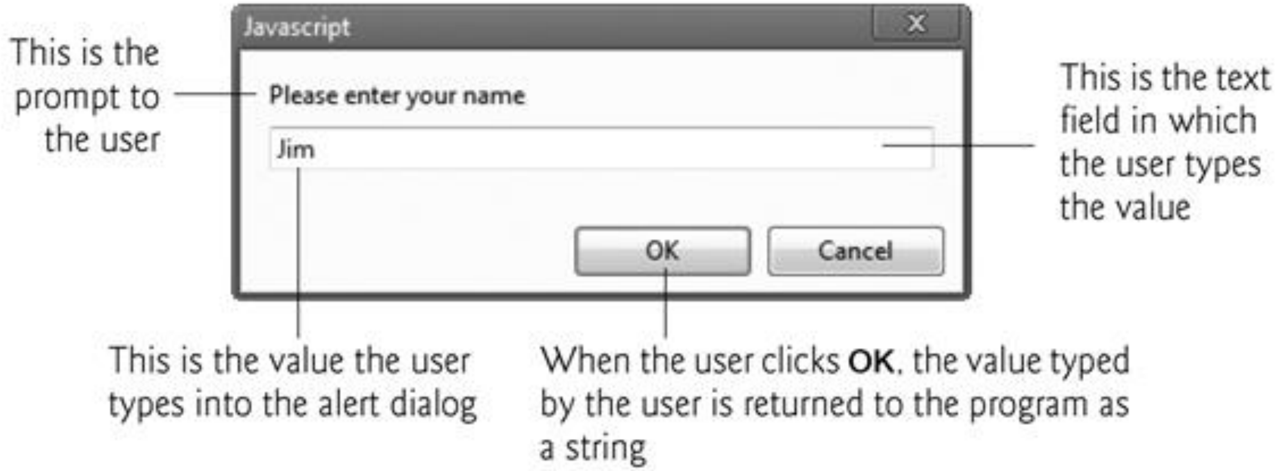
Good Programming Practice 6.4

Although it's not required, declare each variable on a separate line. This allows for easy insertion of a comment next to each declaration. This is a widely followed professional coding standard.

PROPMT DIALOG

- The window object's prompt method displays a dialog into which the user can type a value.
  - The first argument is a message (called a prompt) that directs the user to take a specific action.
  - The optional second argument is the default string to display in the text field.
- Script can then use the value that the user inputs.

PROMPT DIALOG..



Please enter your name

Jim

This is the prompt to the user

This is the text field in which the user types the value

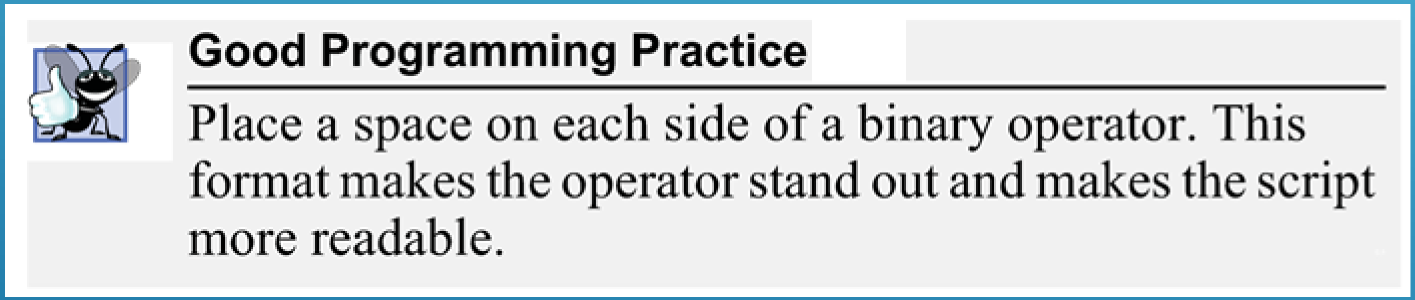
This is the value the user types into the alert dialog

When the user clicks **OK**, the value typed by the user is returned to the program as a string

Prompt dialog displayed by the `window` object's `prompt` method.

DYNAMIC JAVASCRIPT PROGRAM

- A variable is assigned a value with an assignment statement, using the assignment operator, =.
- The = operator is called a binary operator, because it has two operands.

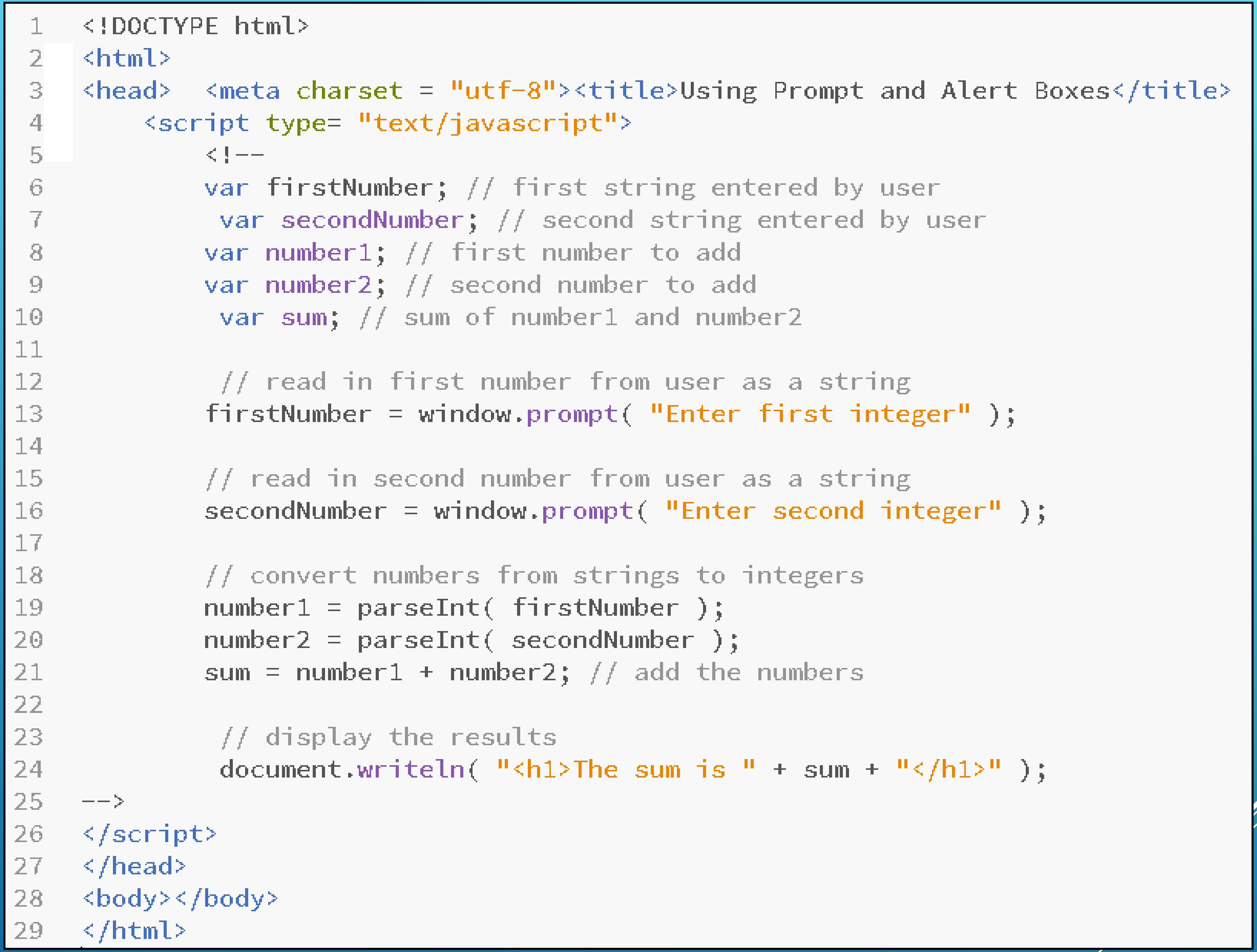


Good Programming Practice

Place a space on each side of a binary operator. This format makes the operator stand out and makes the script more readable.

PARSING

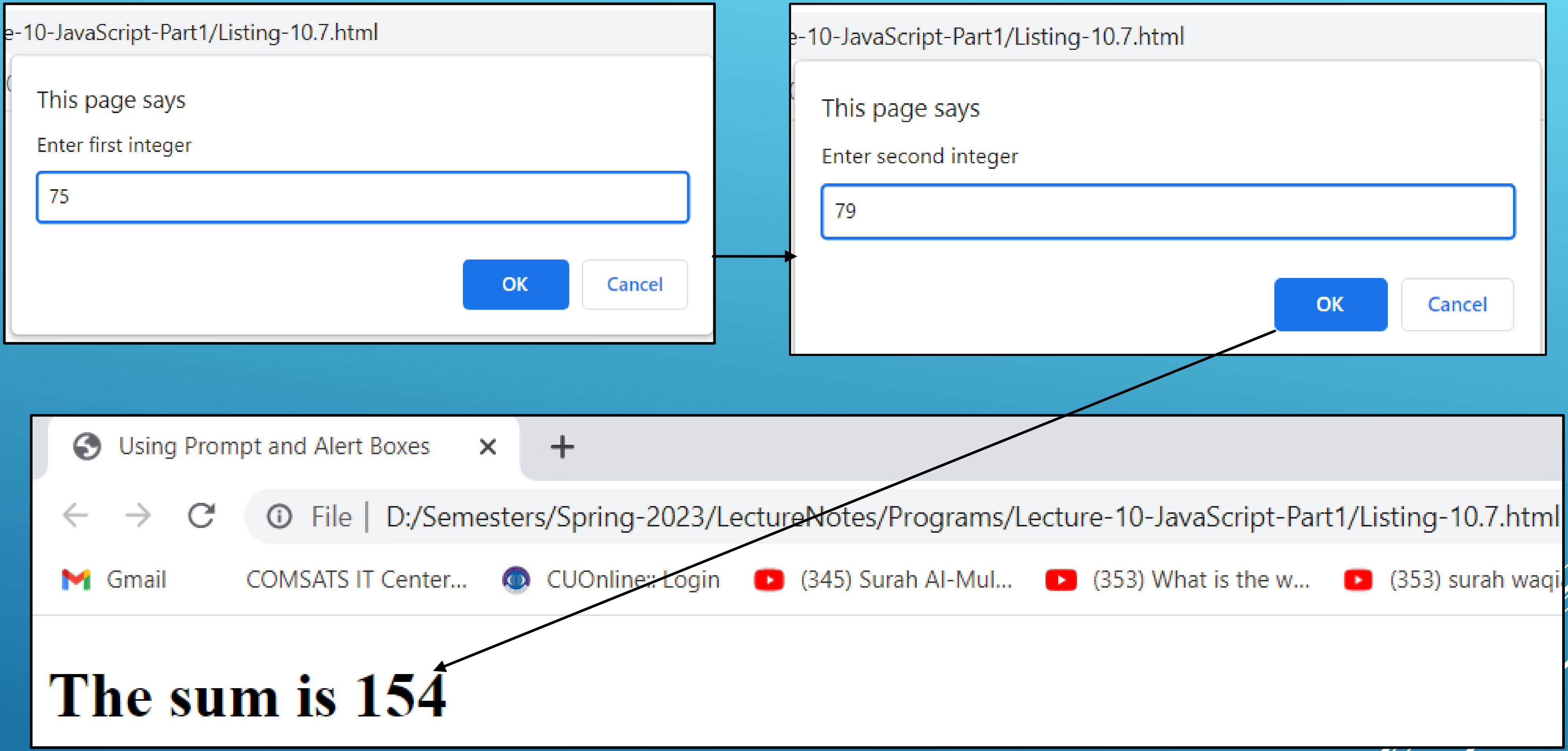
- null keyword
  - Signifies that a variable has no value
  - null is not a string literal, but rather a predefined term indicating the absence of value
  - Writing a null value to the document, however, displays the word “null”
- Function parseInt
  - converts its string argument to an integer
- JavaScript has a version of the + operator for string concatenation that enables a string and a value of another data type (including another string) to be concatenated
- Our next script illustrates another use of prompt dialogs to obtain input from the user.
- Figure 10.7 inputs two *integers* (whole numbers, such as 7, –11, 0 and 31914) typed by a user at the keyboard, computes the sum of the values and displays the result.



```
<!DOCTYPE html>
<html>
<head>  <meta charset = "utf-8"><title>Using Prompt and Alert Boxes</title>
    <script type= "text/javascript">
        <!--
        var firstNumber; // first string entered by user
         var secondNumber; // second string entered by user
        var number1; // first number to add
        var number2; // second number to add
         var sum; // sum of number1 and number2

         // read in first number from user as a string
        firstNumber = window.prompt( "Enter first integer" );

        // read in second number from user as a string
        secondNumber = window.prompt( "Enter second integer" );

        // convert numbers from strings to integers
        number1 = parseInt( firstNumber );
        number2 = parseInt( secondNumber );
        sum = number1 + number2; // add the numbers

         // display the results
         document.writeln( "<h1>The sum is " + sum + "</h1>" );
-->
</script>
</head>
<body></body>
</html>
```

Listing--10.7—Addition of numbers

MATHEMATICAL OPERATIONS IN JAVASCRIPT PROGRAM



e-10-JavaScript-Part1/Listing-10.7.html

This page says

Enter first integer

75

e-10-JavaScript-Part1/Listing-10.7.html

This page says

Enter second integer

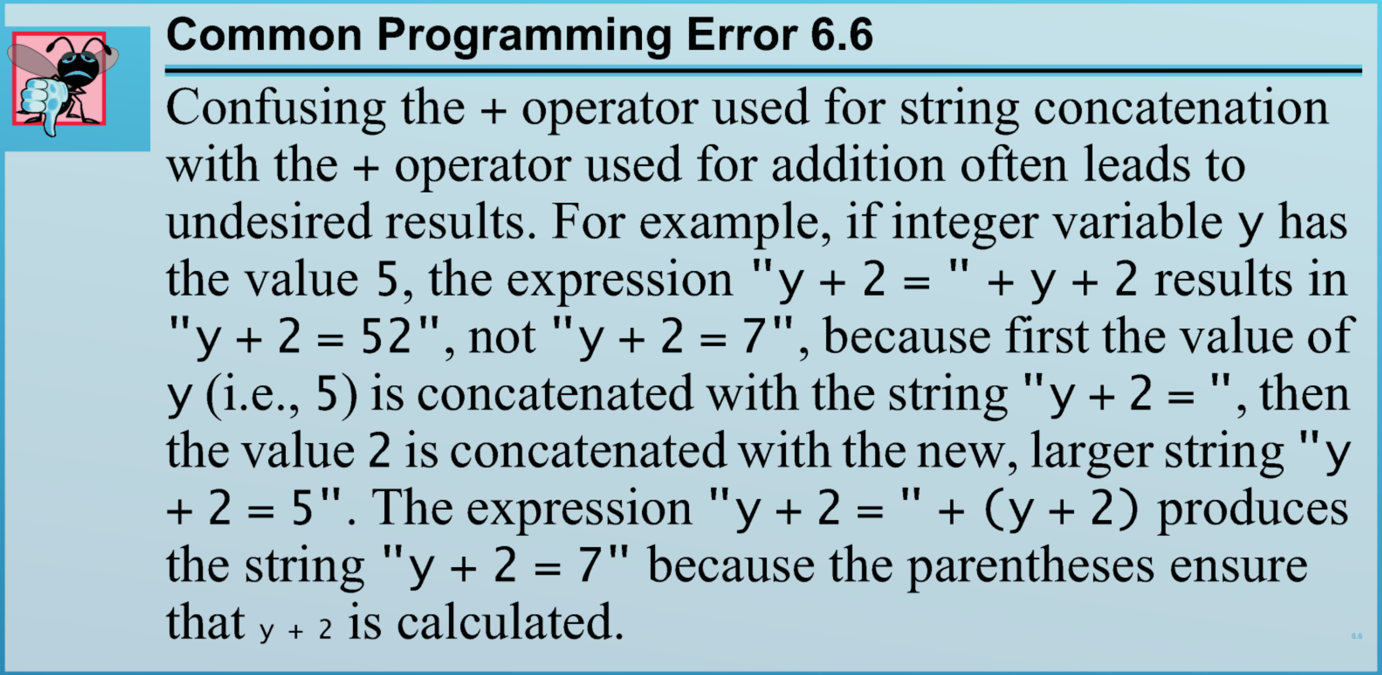
79

D:/Semesters/Spring-2023/LectureNotes/Programs/Lecture-10-JavaScript-Part1/Listing-10.7.html

The sum is 154

Output-Listing--10.7—Addition of numbers

MATHEMATICAL OPERATIONS IN JAVASCRIPT PROGRAM



Common Programming Error 6.6

Confusing the + operator used for string concatenation with the + operator used for addition often leads to undesired results. For example, if integer variable y has the value 5, the expression `"y + 2 = " + y + 2` results in `"y + 2 = 52"`, not `"y + 2 = 7"`, because first the value of y (i.e., 5) is concatenated with the string `"y + 2 = "`, then the value 2 is concatenated with the new, larger string `"y + 2 = 5"`. The expression `"y + 2 = " + (y + 2)` produces the string `"y + 2 = 7"` because the parentheses ensure that `y + 2` is calculated.

MEMORY CONCEPT IN JAVASCRIPT PROGRAM

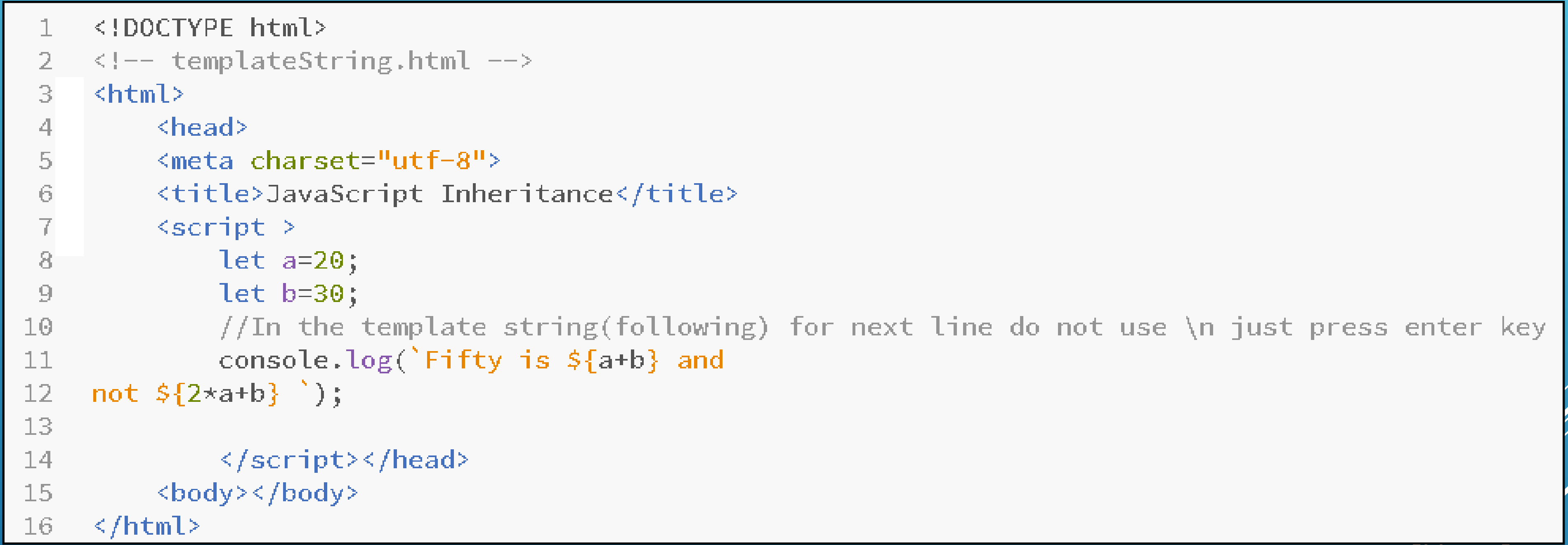
- Variable names correspond to locations in the computer’s memory.
- Every variable has a name, a type and a value.
- When a value is placed in a memory location, the value replaces the previous value in that location.
- When a value is read out of a memory location, the process is nondestructive.

MEMORY CONCEPT IN JAVASCRIPT PROGRAM

- JavaScript does not require variables to have a type before they can be used in a script
- A variable in JavaScript can contain a value of any data type, and in many situations, JavaScript automatically converts between values of different types for you
- JavaScript is referred to as a loosely typed language
- When a variable is declared in JavaScript, but is not given a value, it has an **undefined** value.
  - Attempting to use the value of such a variable is normally a logic error.
- When variables are declared, they are not assigned default values, unless specified otherwise by the programmer.
  - To indicate that a variable does not contain a value, you can assign the value **null** to it.

TEMPLATE STRING IN JAVASCRIPT PROGRAM

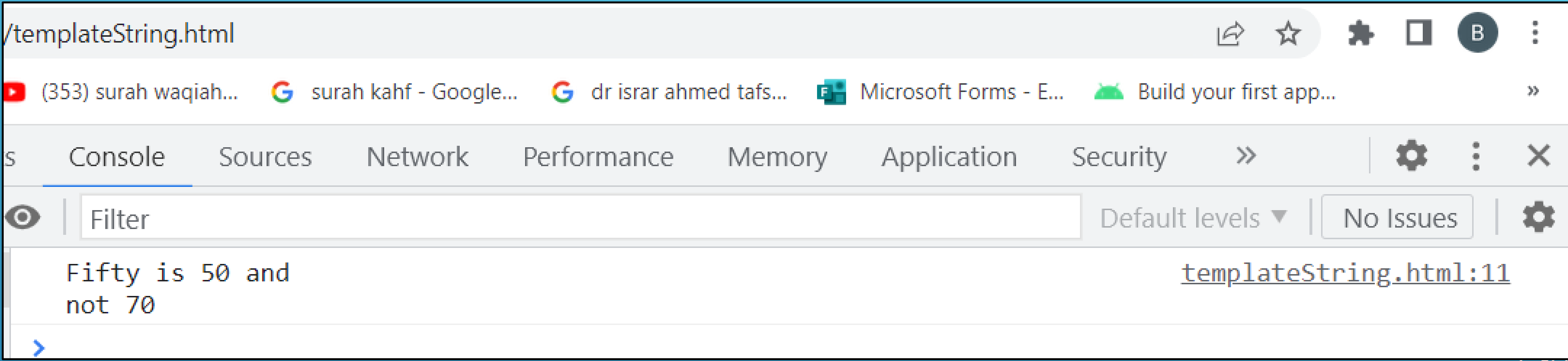
- In JavaScript ECMA6 we can use template string to display text and output of mathematical expression in a very simple way



```
<!DOCTYPE html>
<!-- templateString.html -->
<html>
    <head>
    <meta charset="utf-8">
    <title>JavaScript Inheritance</title>
    <script >
        let a=20;
        let b=30;
        //In the template string(following) for next line do not use \n just press enter key
        console.log(`Fifty is ${a+b} and
not ${2*a+b} `);

        </script></head>
    <body></body>
</html>
```

templateString.html

TEMPLATE STRING IN JAVASCRIPT PROGRAM



```
Fifty is 50 and
not 70
```

templateString.html:11

Output-templateString.html

COMSATS University Islamabad, Abbottabad Campus

OPERATORS IN JAVASCRIPT PROGRAM

- The operator itself is a keyword or symbol that does something to a value when used in an expression. For example, the arithmetic operator + adds two values together.
- The symbol is used in an expression with either one or two values and performs a calculation on the values to generate a result. For example, here is an expression that uses the x operator:

```
area = (width x height);
```

- An expression is just like a mathematical expression. The values are known as operands. Operators that require only one operand (or value) are sometimes referred to as unary operators, while those that require two values are sometimes known as binary operators.

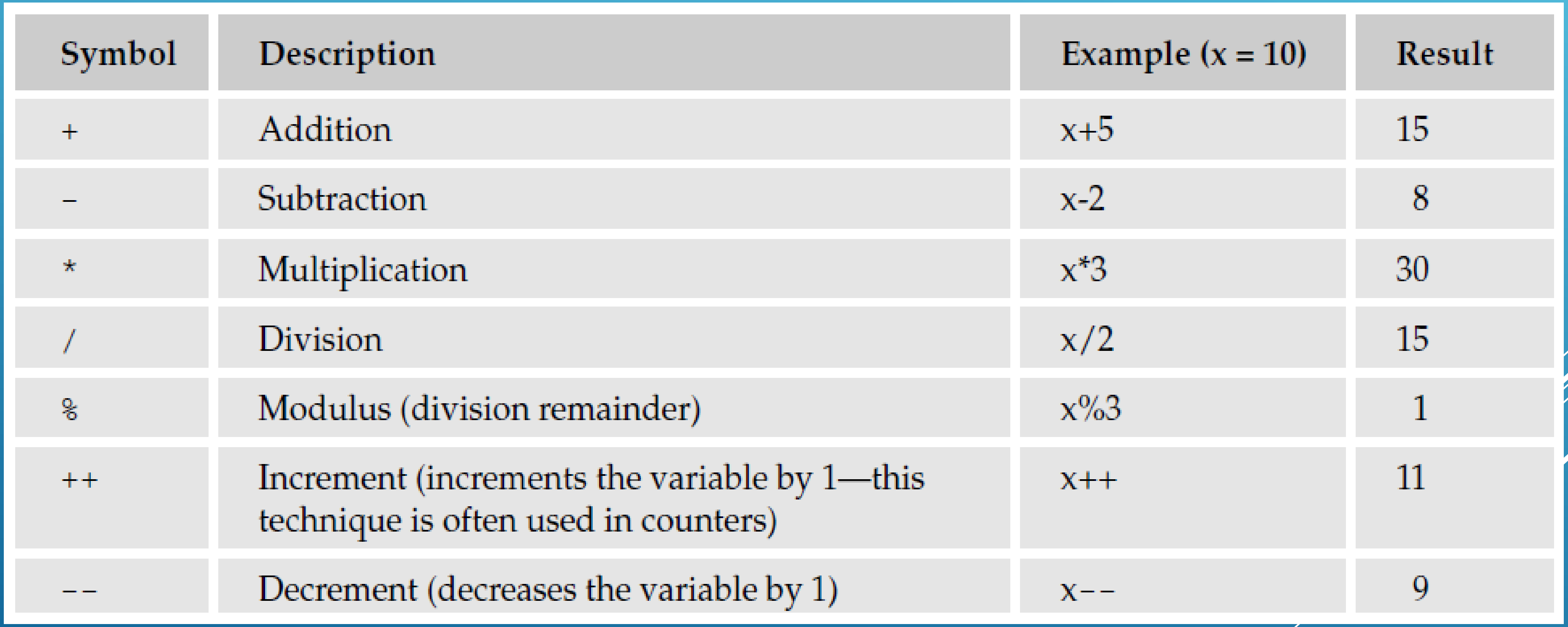
TYPES OF OPERATORS IN JAVASCRIPT

- ❑ Arithmetic operators
- ❑ Arithmetic Assignment operators
- ❑ Comparison operators
- ❑ Logical operators
- ❑ String operators

- Dear students: All these operators are applied in the same way as you already studied in compiled languages like C/C++, Java.
- Hence, I did not explaining them again here with details
- But there are some other operators in Comparison operators which you did not studied previously , hence here we will focus them.
- Also, we deeply study String operations too

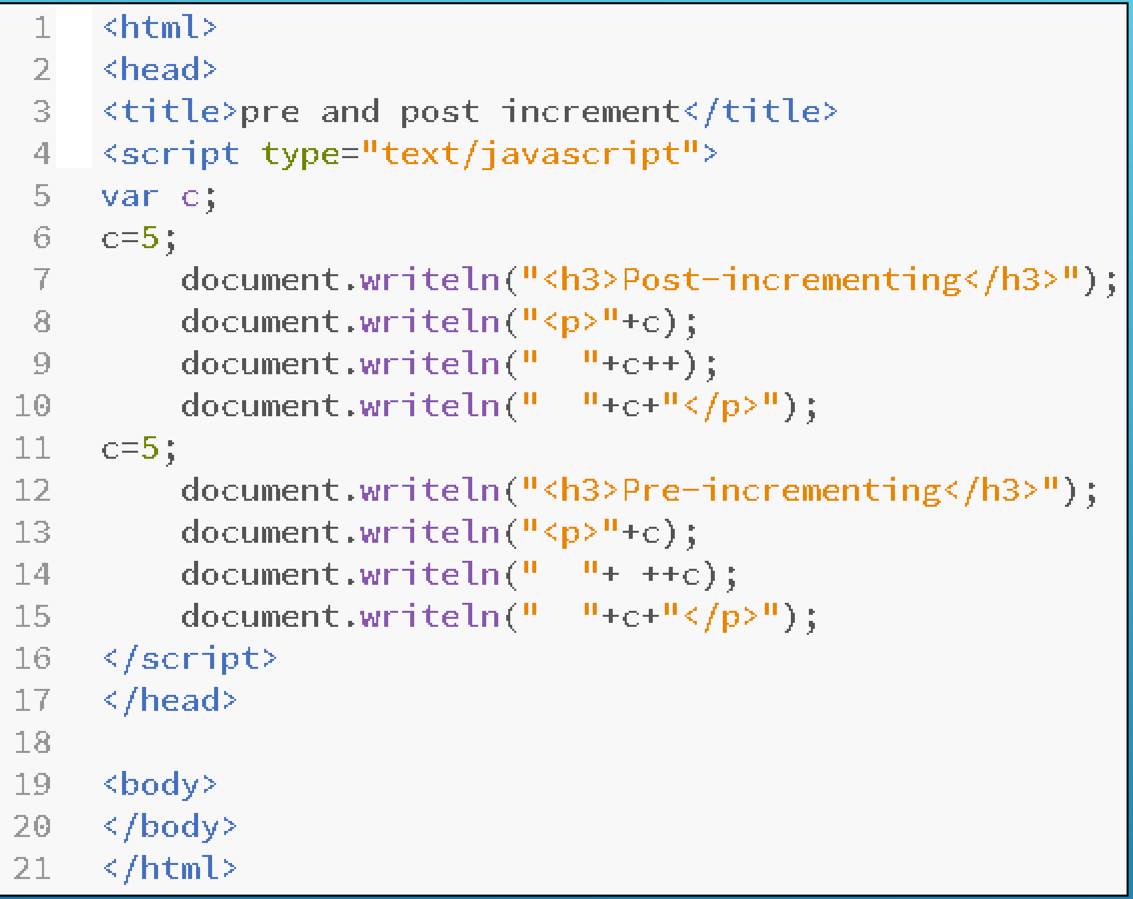
ARITHMETIC OPERATORS IN JAVASCRIPT

- Arithmetic operators perform arithmetic operations upon operands. (Note that in the examples in the following table x = 10.)

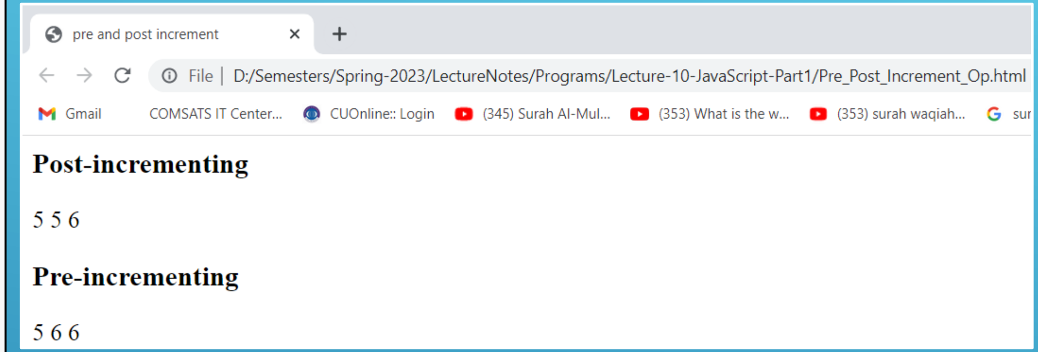


Symbol	Description	Example (x = 10)	Result
+	Addition	x+5	15
–	Subtraction	x-2	8
*	Multiplication	x*3	30
/	Division	x/2	15
%	Modulus (division remainder)	x%3	1
++	Increment (increments the variable by 1—this technique is often used in counters)	x++	11
--	Decrement (decreases the variable by 1)	x--	9

ARITHMETIC OPERATORS IN JAVASCRIPT



```
<html>
<head>
<title>pre and post increment</title>
<script type="text/javascript">
var c;
c=5;
    document.writeln("<h3>Post-incrementing</h3>");
    document.writeln("<p>"+c);
    document.writeln("  "+c++);
    document.writeln("  "+c+"</p>");
c=5;
    document.writeln("<h3>Pre-incrementing</h3>");
    document.writeln("<p>"+c);
    document.writeln("  "+ ++c);
    document.writeln("  "+c+"</p>");
</script>
</head>

<body>
</body>
</html>
```



```
Post-incrementing

5 5 6

Pre-incrementing

5 6 6
```

Listing-10.11-Pre_Post_Increment_Op.html

ARITHMETIC ASSIGNMENT OPERATORS IN JAVASCRIPT

- The basic assignment operator is the equal sign, but do not confuse this to mean that it checks whether two values are equal. Rather, it’s used to assign a value to the variable on the left of the equal sign.
- The assignment operator can be combined with several other operators to allow you to assign a value to a variable *and perform an operation in one step*. For example, with the arithmetic operators the assignment operators can be used to create shorthand versions of operators.

ARITHMETIC ASSIGNMENT OPERATORS IN JAVASCRIPT

- For example, take a look at the following statement:

total = total - profit

- This can be reduced to the following statement:

total -= profit

- While it might not look like much, this kind of shorthand can save a lot of code if you have a lot of calculations such as this to perform.

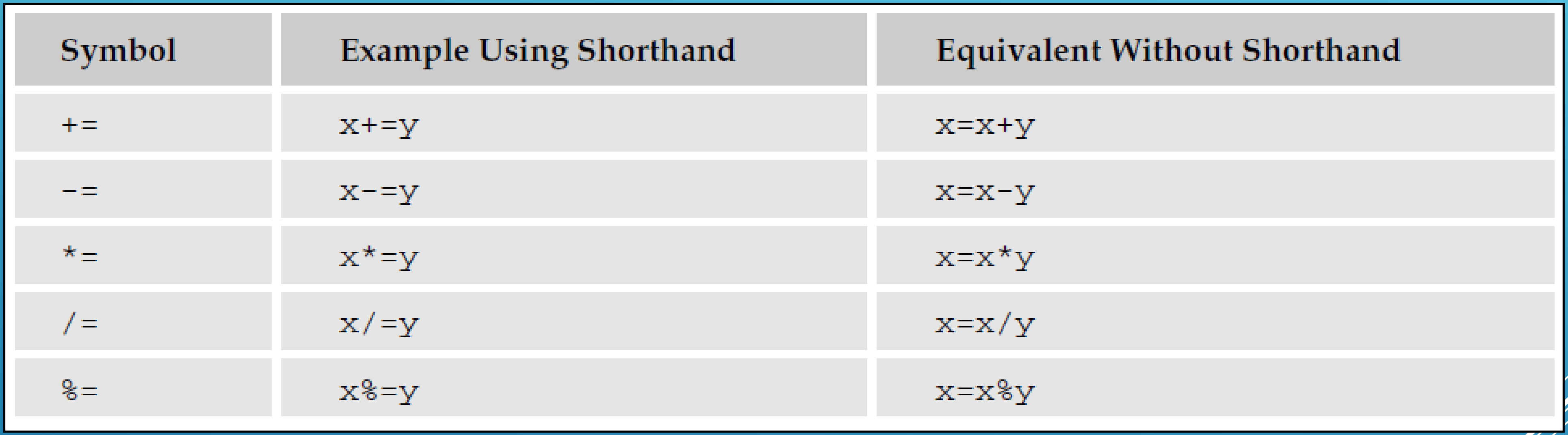
A=A+B

First add B into A

The answer will be stored in A

A+=B-----Arithmetic Assignment Op

ARITHMETIC ASSIGNMENT OPERATORS IN JAVASCRIPT

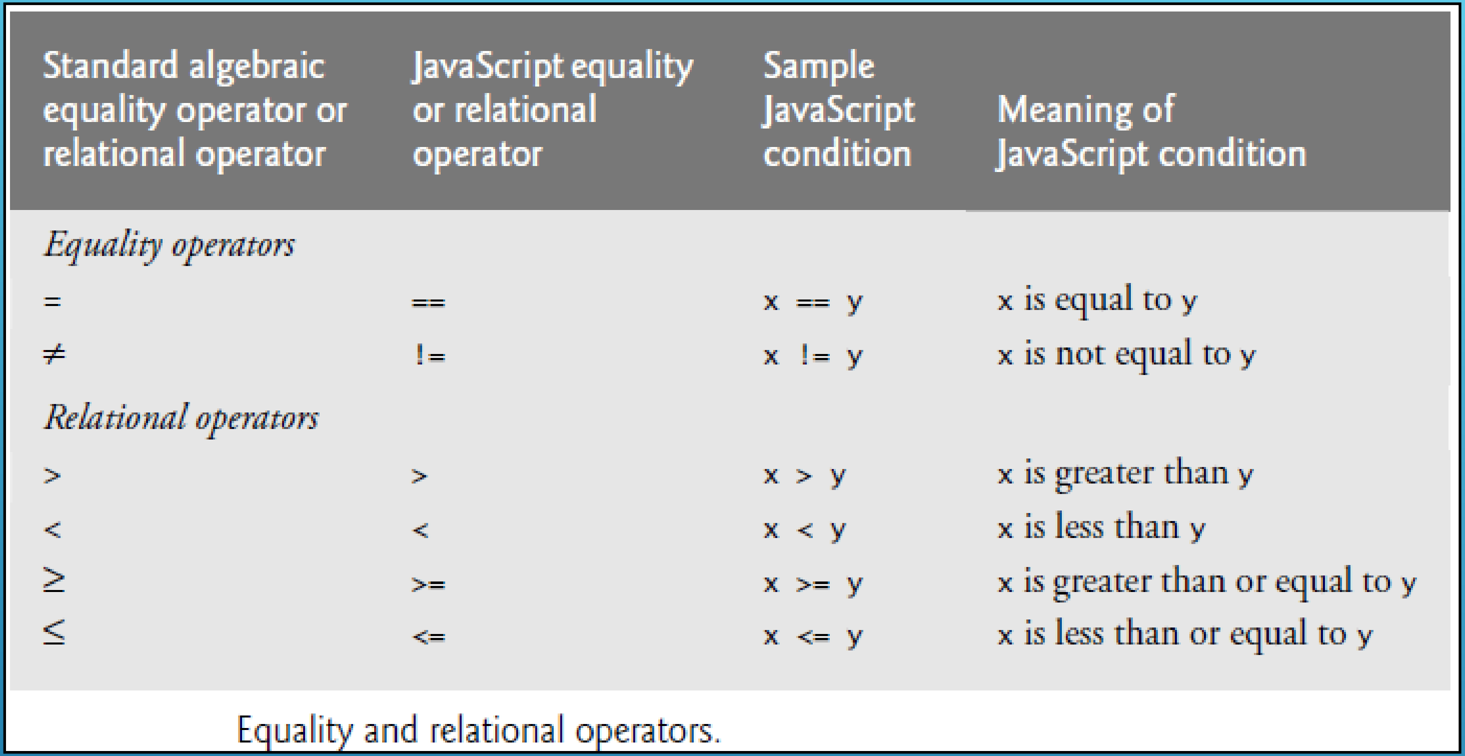


Symbol	Example Using Shorthand	Equivalent Without Shorthand
+=	x+=y	x=x+y
-=	x-=y	x=x-y
*=	x*=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

COMPARISON OPERATORS IN JAVASCRIPT

- As you can see in the table that follows, comparison operators compare two operands and then return either true or false based on whether the comparison is true or not.
- Note that the comparison for checking whether two operands are equal is two equal signs (a single equal sign would be an assignment operator).

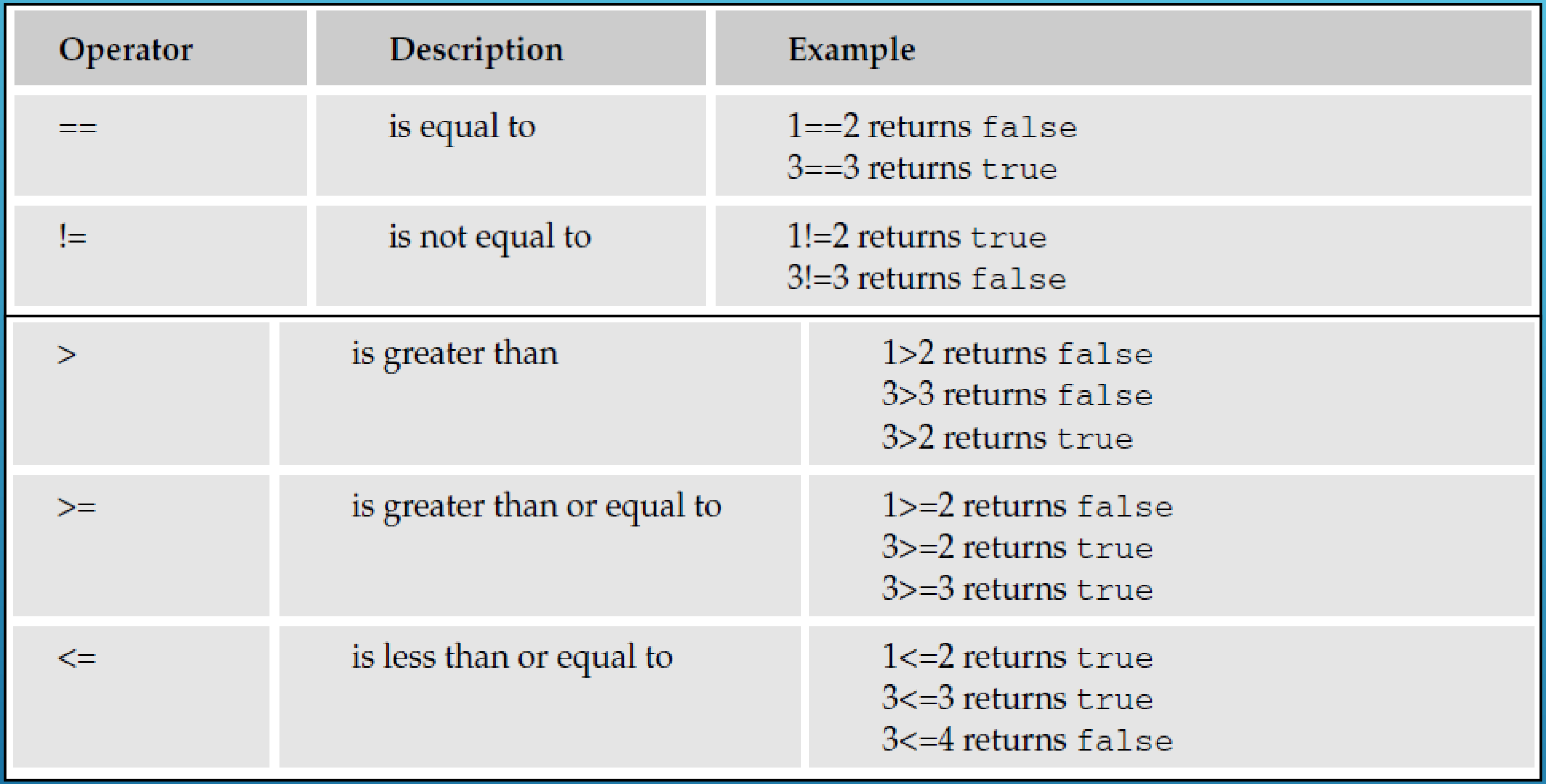
COMPARISON OPERATORS IN JAVASCRIPT



Standard algebraic equality operator or relational operator	JavaScript equality or relational operator	Sample JavaScript condition	Meaning of JavaScript condition
Equality operators			
=	==	x == y	x is equal to y
≠	!=	x != y	x is not equal to y
Relational operators			
>	>	x > y	x is greater than y
<	<	x < y	x is less than y
≥	>=	x >= y	x is greater than or equal to y
≤	<=	x <= y	x is less than or equal to y

Equality and relational operators.

COMPARISON OPERATORS IN JAVASCRIPT

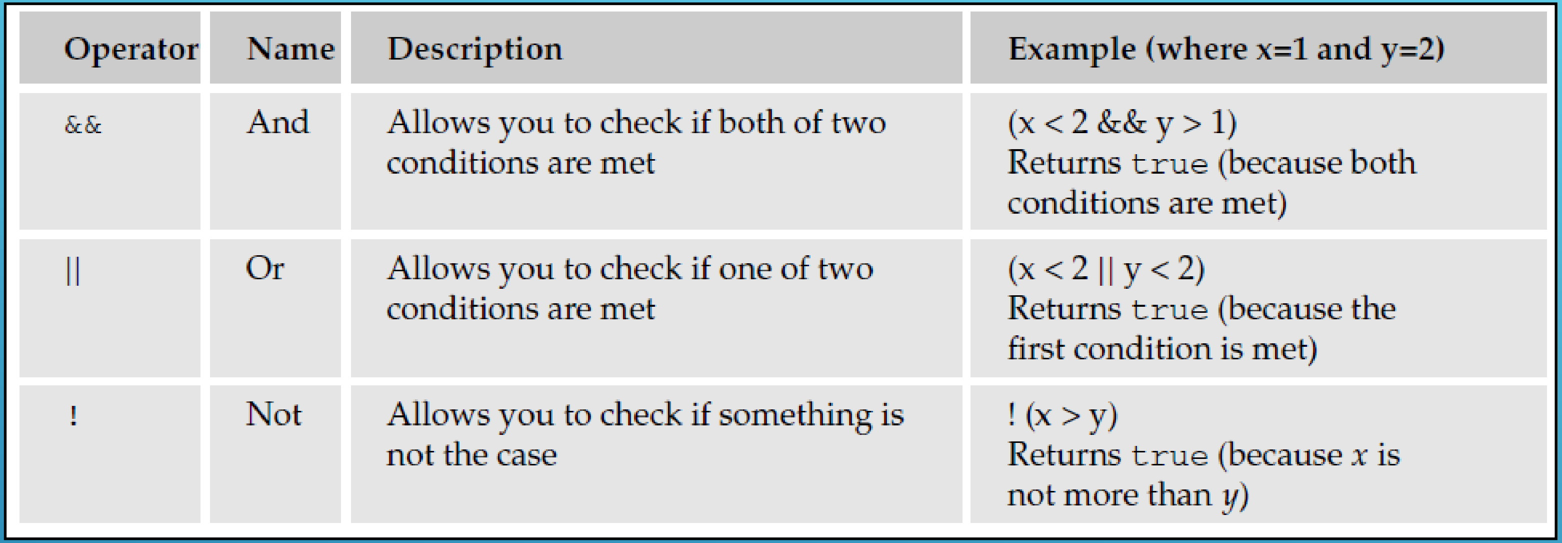


Operator	Description	Example
==	is equal to	1==2 returns false 3==3 returns true
!=	is not equal to	1!=2 returns true 3!=3 returns false
>	is greater than	1>2 returns false 3>3 returns false 3>2 returns true
>=	is greater than or equal to	1>=2 returns false 3>=2 returns true 3>=3 returns true
<=	is less than or equal to	1<=2 returns true 3<=3 returns true 3<=4 returns false

LOGICAL OPERATORS IN JAVASCRIPT

- Logical or Boolean operators return one of two values: true or false.
- They are particularly helpful because they allow you to evaluate more than one expression at a time. (Note that in the examples in the following table x=1 and y=2.)

LOGICAL OPERATORS IN JAVASCRIPT



Operator	Name	Description	Example (where x=1 and y=2)
&&	And	Allows you to check if both of two conditions are met	(x < 2 && y > 1) Returns `true` (because both conditions are met)
||	Or	Allows you to check if one of two conditions are met	(x < 2 || y < 2) Returns `true` (because the first condition is met)
!	Not	Allows you to check if something is not the case	! (x > y) Returns `true` (because *x* is not more than *y*)

STRING OPERATORS IN JAVASCRIPT

- + operator is also called concatenating operator when it used to merge(add) strings or strings with numbers.
- Here are some examples

<u>Example-1: concatenating two strings using + operator:</u>

```
firstName = “Ali”
lastName = “ Ahmed”
name = firstName + lastName
  document.writeln(name);
```

- The value of the name variable would now be Ali Ahmed. The process of adding two strings together is known as *concatenation*.

STRING OPERATORS IN JAVASCRIPT

Example-2: concatenating numbers with string using + operator:

```
document.writeln(6+9+”Total marks”);
```

- The output will be 15 Total Marks

How it solved?

- Step-1:The first + is between two numbers hence here addition will occur which results: 15
- Step-2: Second + is between 15 and “Total marks”, here it applied concatenation process because a non-text and text results into text
- Hence complete output is 15 Total Marks

STRING OPERATORS IN JAVASCRIPT

Example-3: concatenating string with numbers using + operator:

```
document.writeln(”Total marks are ”+6+9);
```

- The output will be Total Marks are 69------------(wrong addition)

How it solved?

- Step-1:The first + is between text and number: It will applied as concatenating operator which results in “Total Marks are 6”
- Step-2: Second + is between text results from step1 and 9: again concatenation applied
- Hence complete output is Total Marks are 69, which is wrong

STRING OPERATORS IN JAVASCRIPT

Example-4: Solution of Example-3 problem of wrong addition

```
document.writeln(”Total marks are ”+(6+9));
```

- The output will be Total Marks are 15------------(Correct addition)

How it solved?

- Step-1:The first () will solved and 15 will be result
- Step-2:+ is between text and number: It will applied as concatenating operator which results in “Total Marks are 15”
- You can also compare strings using the comparison operators you just met