3. JavaScript Keywords, Variables, and Operators

JavaScript keywords are reserved identifiers for the following purposes:

- Define user data and/or object types (var)
- Create and delete objects (new, delete)
- Define user needed function or module (function, void, return)
- Perform program flow controls (if, else, for, while, do, switch, case, default, continue, break)
- Access and validate object (this, typeof)
- Access constants and values (true, false, void)
- Import and export object propertied and methods in the current window or execution context (import, export)

JavaScript Keywords

break	case	continue	default	delete
do	else	export	false	for
function	if	import	in	new
null	return	switch	this	true
typeof	var	void	while	with

Reserved Keywords for future extensions

catch	class	const	debugger	enum
extends	finally	super	throw	try

Variables and Values

Value Types

Type	Description	Example
Number	Any numeric value	10.145
String	Character inside quotation marks	"Web Programming"
	Single quotes or double quotes	'testing'
Boolean	True or False	false
NULL	Empty value	
Objects	Compound data types that contains methods and properties	

Examples of Supported Number Systems

```
10          // Integer
3.1416      // Float
0.003       // Float
6.20E3      // Float in exponential form
0377        // Octal number; or 255 in decimal
0xFF        // Hex number; or 255 in decimal
```

Examples of String

```
'ECET Dept'
"3.1416"            // ASCII Number string
"me = ' thisform' "
```

Variable Declaration

```
var n, m;       // uninitialized
var x = 100, message = 'Hello you';
```

Escape Sequence (control characters and special characters)

\b	Backspace
\f	Form feed
\n	New line
\r	Carriage return
\t	Tab
\'	Apostrophe or single quote
\"	Double quote
\\	Back slash
\ddd	Three octal digits (d: 0,1, through 7)
\xdd	Two hexadecimal digits (d: 0, 1,2,..,9, A,B,C,D,E,F)
\udddd	Unicode encoding by four hex digits

Arrays

```
var n = new Array();          // n is declared as an Empty array
var b = new Array(20);        // b is declared as an array of 20 elements
var k = new Array(1, 2, 3);  // k is declared and initialized with three elements
var k = [1,  2, 3];
var anyobjet = [1, true, [1,2], {x:1, y:2}, "Hello"];
```

Null and Undefined

null - no value

Conditional Operator (? :)

Math.abs(x) or

x < 0? -x: x

typeof operator

Evaluate the type of object: "number", "string", or "boolean"

new operator

Creating new objects

delete operator

Delete existing objects

Math Operators

+	x + y	Add operation if x and y are both numeric types
+	x + y	String concatenation
-	x - y	Subtraction
*	x * y	Multiplication
/	x / y	Division
%	x%y	Modulus of x and y (remainder operation)
++	x++, ++x	Increment (post, pre)
--	x--, --x	Decrement (post, pre)
-	-x	Negate (change sign)

Bitwise Operators

&	x & y	bitwise AND; both x and y are unsigned integer
|	x | y	bitwise OR; both x and y are unsigned integer
^	x ^ y	bitwise XOR; both x and y are unsigned integer
~	~x	bitwise NOT (complement)
<<	x << 2	Shift left 2 bit (times 4)
>>	x >> 2	Shift right 2 bit (div by 4)
>>>	x >>> 2	Shift right zero fill (div 4)

Assignment Operators

=	x = y	Assignment, copy value of y to x
+=	x += y	Add and assign; x = x +y
-=	x -= y	Sub and assign; x = x -y
*=	x *= y	Multiply and assign; x = x * y
/=	x /= y	Divide and assign; x = x / y
%=	x %= y	Modulus and assign; x = x % y
<<=	a <<= b	Shift a to left for b bit positions; a = a << b
>>=	a >>= b	Shift a to right for b bit positions; a = a >> b
>>>=	a >>>= b	Shift a to right (zero filled) for b bit positions; a >>>= b
&=	a & b	a = a BIT-WISE AND b
|=	a |= b	a = a BIT-WISE OR| b
^=	a ^= b	a = a BIT-WISE EXCLUSIVE-OR b

Relational Operators (Numeric types)

==	x == y	Equal to; return true if x and y are equal
!=	x != y	Not equal; return true if x and y are not equal
>	x > y	Greater than; return true if x is greater than y
>=	x >= y	Greater than or equal to ; return true if x is greater than or equal to y
<	x < y	Less than; return true if x is less than y
<=	x <= y	Less than or equal to; return true if x is less than or equal to y
&&	x && y	Logical AND; Return true if both x and y are true
||	x || y	Logical OR; return true if either x or y is true
!	!x	Logical NOT; return true if x is false

Global Constant

Infinity

NaN - not a number

Global Functions

escape(s)
- Encode a string for transmission; RFC 1783

unescape(s)
- Decode an escaped string

eval(code)
- Execute or evaluate JavaScript expression code from a string and return a value

getClass(javaobj)
- Return JavaClass of a JavaObject

isFinite(n)

- Determine if a number is finite; return true or false

isNaN(x)

- Determine if a number is defined

parseFloat(s)

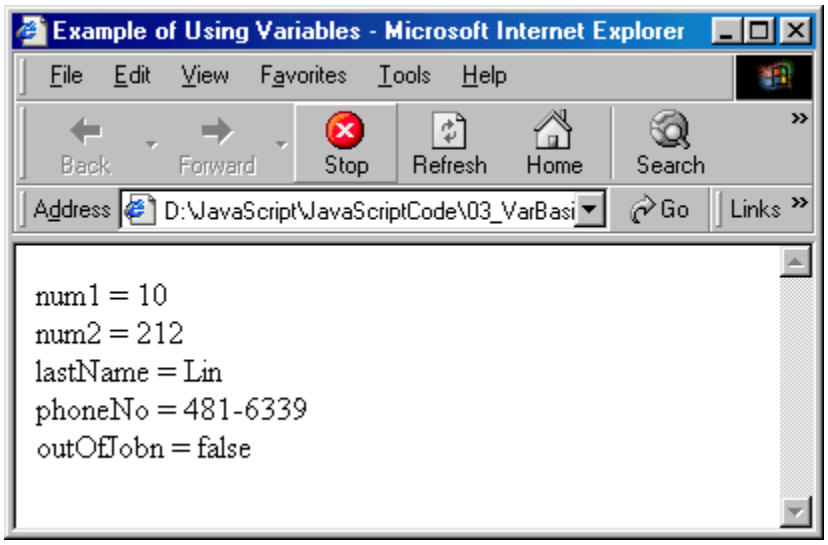
- Convert a string to a floating point number

parseInt(s, radix)

- Convert a string to an integer

Example 3-1: Declare and use JavaScript variables.

```
<html>
<!-- variable.html
  -->
<head>
<title>Example of Using Variables</title>
<script language=JavaScript>
var num1 = 10;    // a number variable
var num2 = 212.00;// a float number
var lastName ="Lin"; // a string variable
var phoneNo ='481-6339';  // a string variable
var outOfJob = false;  // a boolean variable

document.write("num1          = "+ num1 + "<BR>");
document.write("num2          = "+ num2 + "<BR>");
document.write("lastName      = "+ lastName + "<BR>");
document.write("phoneNo       = "+ phoneNo  + "<BR>");
document.write("outOfJobn     = "+ outOfJob + "<BR>");
</script>

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

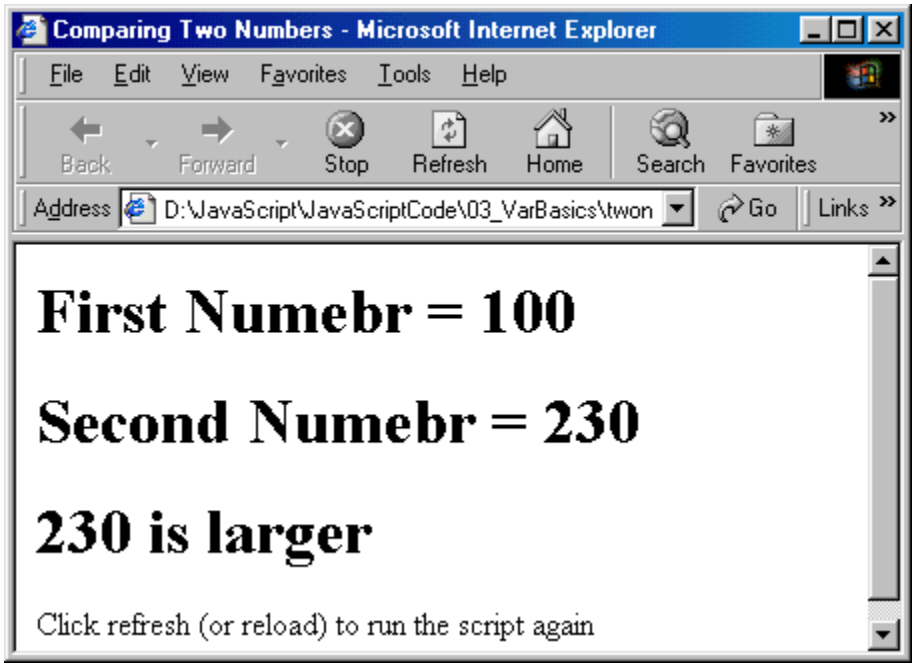


Example 3-2: Enter two numbers and make a comparison.

```
<html>
<!-- twonums.htm -->
<Head>
<Title>Comparing Two Numbers</title>
<script language="JavaScript">
<!--
var num1Str, num2Str;   // Define variables
var num1Int;  //first integer value
var num2Int;  //second integer value

//read first number from user as string
num1Str = window.prompt ("Enter first integer","0");
num1Int = parseInt(num1Str);   //Convert string to integer

//read second number from user as string
num2Str = window.prompt ("Enter second integer","0");
num2Int = parseInt(num2Str); //Convert string to integer

document.writeln("<H1>"+ "First  Numebr = " + num1Str +"</H1>");
document.writeln("<H1>"+ "Second Numebr = " + num2Str +"</H1>");
// Comparing two numbers
if (num1Int > num2Int)
        document.writeln("<H1>"+ num1Str + " is larger</H1>");
if (num1Int < num2Int)
        document.writeln("<H1>"+ num2Str+ " is larger</H1>");
if (num1Int == num2Int)
document.writeln("<H1>Two numbers are equal</H1>");
//-->
</script>
</head>
<body>
<p>Click refresh (or reload) to run the script again</p>
</body>
</html>
```

Output:

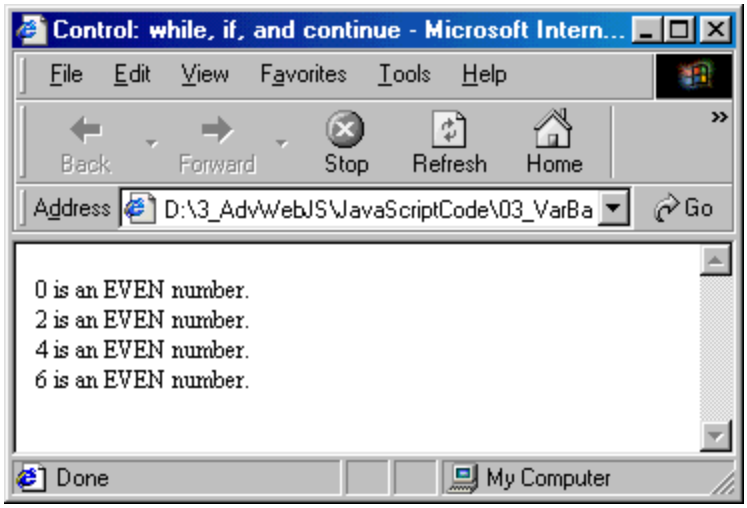

Explorer User Prompt
JavaScript Prompt:
Enter second integer
OK
Cancel
230


Comparing Two Numbers - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites
Address D:\JavaScript\JavaScriptCode\03_VarBasics\twon
Go Links
First Numebr = 100
Second Numebr = 230
230 is larger
Click refresh (or reload) to run the script again

Example 3-3: Use a while loop to generate a series of numbers and determine the even or odd number properties using decision making statement - if.

```
<html>
<!-- while_if.html
 -->
<head>
<title>Control: while, if, and continue</title>
</head>
<body>
<script language="JavaScript">
<!--
var numVar = 0;

while (numVar < 7)
 {
  if ((numVar % 2)== 0 )
   {
    document.write(numVar + " is an EVEN number.<BR>");

   }
  numVar = numVar + 1;

  }
-->
</script>
</body>
</html>
```

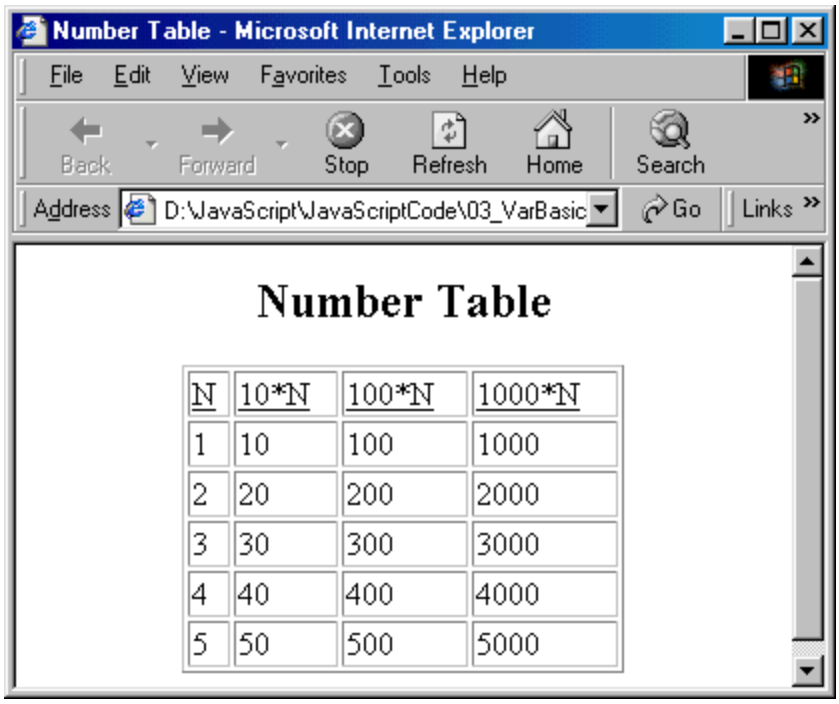
Output:



Example 3-4: Create a table of 4 columns and 5 rows.

```
<html>
<head>
<!-- numtable.html -->

<title>Number Table</title>
<script language="JavaScript">
// Define variables
var num = 0;           //num variable
var num10 = 0;         //10* num
var num100 = 0;        //100* num
var num1000 = 0;       //1000*num
//create a table and print headings
document.writeln("<h2 align= 'center'>Number Table</h2>");
document.writeln("<table border='1' align='center' width='60%'>");
document.writeln("<tr><td><u>N</u></td><td><u>10*N</u></td><td><u>100*N
</u></td><td><u>
  1000*N</u></td></tr>");num =1;       //initialize number to its first
value in the
while (num <=5)
{
        num10   = 10 * num;
        num100  = 100 * num;
        num1000 = 1000 * num;
        document.writeln("<tr><td>"+ num + "</td><td>" + num10 + "
</td><td>" + num100 +  " </td><td>" + num1000 + "</td></tr>");
//print entries
        num++;
}
document.writeln("</table>");
</script> </head>
<body></body>
</html>
```

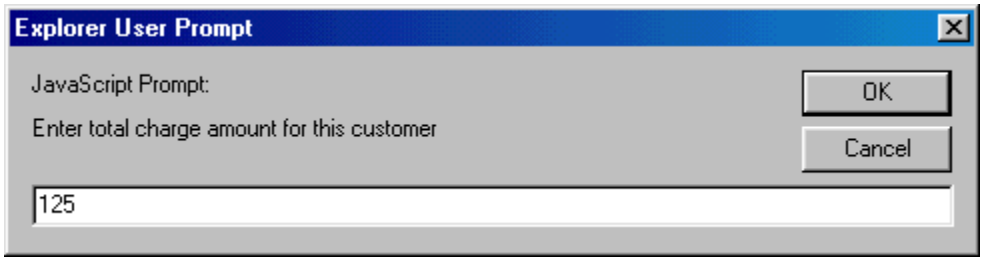
Output:

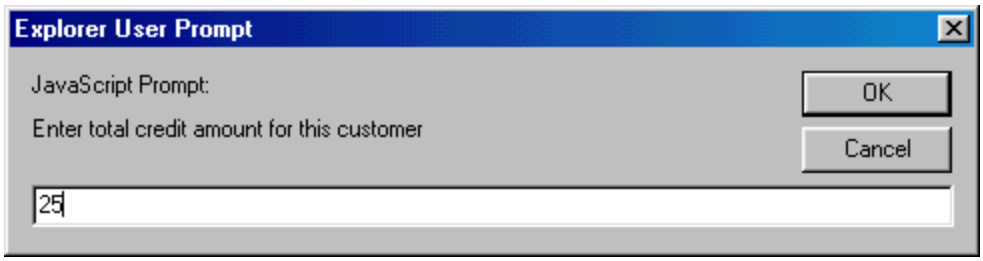


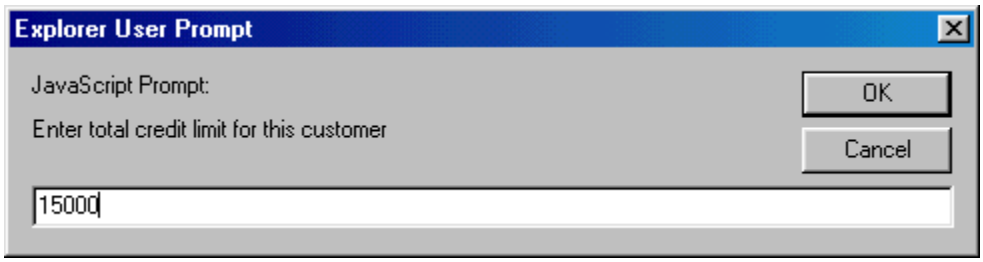
Example 3-5: Customer account and credit evaluation example.

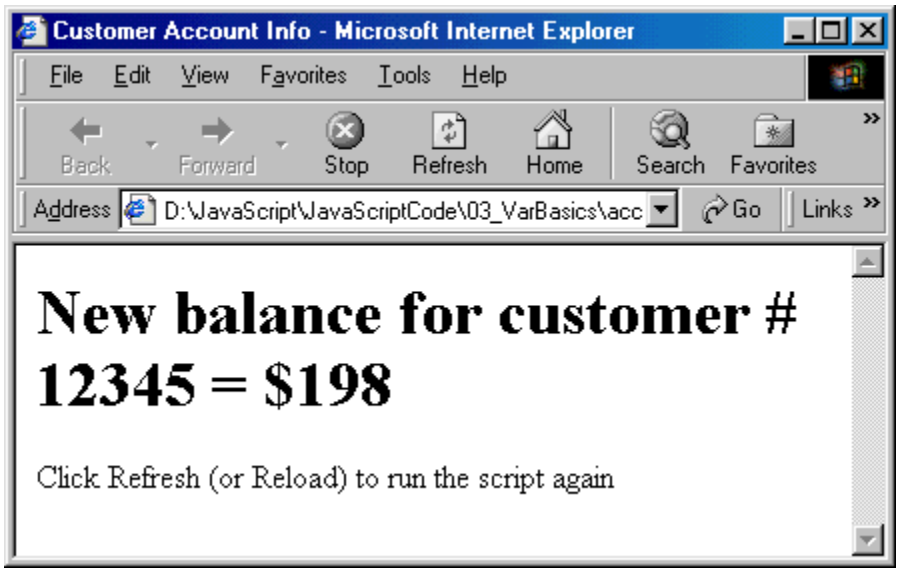
```
<html>
<!-- account.html -->
<head>

<title>Customer Account Info</title>

<script language="JavaScript">
// Define string variables
var acctNumber;      //customer's account number
var balance;
var charge;          // total of all items charged by this customer
                     // for this month
var credit;          //total of all credits applied to this customer
var creditLimit;

//Define variables for holding numerical value
var valBalance;      // balance as a Number
var valCharge;
var valCredit;
var valCreditLimit;

// Read inputs
acctNumber=window.prompt("Enter customers account number","0");
balance=window.prompt("Enter beginning balance","0");
valBalance=parseInt(balance);
charge=window.prompt("Enter total charge amount for this
customer","0");
valCharge=parseInt(charge);
credit=window.prompt("Enter total credit amount for this
customer","0");
```

```
valCredit=parseInt(credit);
creditLimit=window.prompt("Enter total credit limit for this
customer","0");
valCreditLimit=parseInt(creditLimit);
valBalance += (valCharge - valCredit);

document.writeln("<h1>New balance for customer #  " + acctNumber+ " =
$"
      + valBalance + "</h1>");

if (valBalance > valCreditLimit)
        document.writeln("<h1>Credit Limit Exceeded</h1>");
</script>
</head>
<body>
<p>Click Refresh (or Reload) to run the script again</p>
</body>
</html>
```









Example 3-6: Using a for loop control structure to change the appearance and size of a graphic display.

```
<html>
<!-- changeimagesize.html -->
<head>
<title>Image Size </title>
</head>
<body>
<img id="ImgCircle" src="circle.gif" width="37" height="37">
<script language="JavaScript">
var w = ImgCircle.width
var h = ImgCircle.height
```

```
var deltaW = 40
var deltaH = 40
alert("The circle has width "+ w + "and height "+ h)
for (n = 0; h <= 200 && w <= 400 ; n++)
{
        w += deltaW;    h += deltaH;
      ImgCircle.height= w;
      ImgCircle.width= h;
      alert("Image zize is now width "+w+" and height "+h)
}
</script>
</body>
</html>
```

Output: