Scripting language I (LS1)

Course 2

Variables

Variables are an essential element of programming. They allow you to assign a value (string or number) to a word which will then be used as a referent in the code.

The procedure is simple. The variable must first be declared, and then a value must be assigned to it.

To declare a variable, use the term ***let*** followed by a space and the name of the variable you intend to use. Then, in a new instruction, write the name of your variable followed by the equal sign and the value you want to assign to it.

If the script was to be interpreted by browsers prior to version 11, consider using the term ***var*** instead of *let*.

```
<script>
        let myVariable;
        myVariable =  "Bob";
        alert(myVariable);
</script>
```

Variables names can be composed of letters, lowercases and uppercases, numbers and the underscore symbol. .

Warning : A variable name should never start with a number.

Some specific key-words reserved for JavaScript can' be used to declare a variable or a function.

Examples : *break, case, char, continue, delete, double, final, long, new, public* et *super.*

Shortcut :
It is also possible and just as efficient to declare a variable and assign it a value in one line only instead of two.

```
<script>
        let myVariable =  "Bob";
        alert(maVariable);
</script>
```

Using a user input as a value

It is possible to ask a user a question and to assign the user input as a value to a variable which can then be used in the program. To do so, the ***prompt*** function will be used, creating a message window followed by an input field and a confirmation button.

```
<script>
        let name = prompt("What is your name?");
        alert("Hello, " + name);
</script>
```

Note : Strings use quotes, but not variables as it is not the word « name » that has to be displayed, but the avlue it represents.

Just like it was possible to display many alert windows one after another, it is also possible to use multiple prompt windows to ask several questions to a user and, this way, to assign many values to as many variables.

```
<script>
        let name = prompt("What is your name?");
        let age = prompt("How old are you?");
        let city = prompt("In what city do you live?");
        alert("Hello, " + name + ".\n" + "You\'re" + age + " year old and
        you live in " + ville + ".");
</script>
```

Explications :

Les trois premières instruction demandent à l'utilisateur de saisir des valeurs qui sont associées aux variables *nom*, *age* et *ville* tandis que la dernière fait s'afficher une fenêtre dans laquelle s'affiche d'abord la chaîne «Bonjour, » placée entre guillemets. Le signe « + » est utilisé pour ajouter plus d'une valeur.

Note : The three first code lines declare variables and assign them values (user's inputs). In generating the output to the message window, you have to include all necessary spaces and line changes.

Also, in the case of the value assigned to age, although the answer is a number, it is still a string, not a number type value.

Here's another way of achieving the same result :

```
<script>
        let start = name, age, city, result;

        name = prompt('What is your name?');
        age = prompt('How old are you?');
        ville = prompt('In what city do you live?');

        result = "Hello, " + name + ".\n" + "You\'re" + age + " year old
        and you live in " + city + "."

        alert(result);
</script>
```

Explanation :

In the first code line, all variables are declared. The las variable (result) is used to gather all that will be displayed in the message window.

The three following code lines are used to assign values to the variables.

Assignment 2

Using variables as well as *prompt* and *alert* functions, code a program that will ask questions to the user in order to display the following informations in a message window :

FirstName LastName
1234, some street
City, Province
Postal code

EmailAddress

Number type values assigned to variables

The use of number type values assigned to variable is quite the same as what we just did with strings. Quite, but not exactly the same.

```
<script>
        let result = 3+2;
        alert(result);      // Displays: « 5 »
</script>
```

Mathematical operations using variables

Since it is possible to assign number type values to variables, it is then also possible to perform mathematical operations using variables instead of numbers.

```
<script>
        let a = 3, b = 2, result;
        result = a * b;
        alert(result);      // Displays : « 6 »
</script>
```

Mathematics using multiple operators

It is possible to go further by coding mathematical operations including several operators and variables. Note that the rules of parenthesis usage and the operations order are the same as for mathematics.

```
<script>
        result_1 =  (16 + 8 ) / 2 - 2  ;       // preliminary result : 10
        result = result_1/3;
        alert(result);                         // Displays : 3,33
</script>
```

Mathematical operation shortcuts

It is possible to simplify the code using a variable and a number type value.

```
<script>
        let number = 3;
        number = number + 5;
        alert(number);          // Displays : « 8 »
</script>
```

Even more simple, you can avoid repeating the variable simply by using **«+=»** operator:

```
<script>
        let number = 3;
        number += 5;
        alert(number);          // Displays : « 8 »
</script>
```

This last technique may also be used with strings:

```
<script>
        let text = "Hello ";
        text += "world";
        alert(text);            // Displays « Hello world ».
</script>
```

Note:
These doesn't only apply to addition but to all operators (+=, -=, *=, /=).

Number type values supplied by user

In the following example, the user is asked to supply two numbers and the program will display the sum. Since user inputs are strings, it is necessary to convert the user value in number type values.

```
<script>
        let a = number(prompt("Write a number : "));
        let b = number(prompt("Write another number : "));
        alert(a + " + " + b + " = "  + (a+b));
</script>
```

Explanation :

The two values supplied by the user has been assigned to variables ***a*** and ***b***.

In order to define the values as number type, the function ***number()*** has been used, so anything within its parenthesis is considered being of number type.

Converting string into number values

Instead of using ***number()*** function like we just did, it is even better and simpler to simply convert variables values using ***parseInt()*** function.

```
<script>
        let a, b, result;

        a = prompt("Write a number : ");
        b = prompt("Write another number : ");
        result = parseInt(a) + parseInt(b);

        alert(a + " + " + b + " = " + result);
</script>
```

Converting number type values in strings (method 1)

We have seen that concatenation is possible between a string and a number. But, it is impossible to concatenate two number; the addition operator would generate the result instead of displaying the two numbers. The two values must then be converted. To do so, simply add a string in between the two number type variables :

```
<script>
        let text, number1 = 4, number2 = 2;
        text = number1 + "" + number2;
        alert(text);                        // Displays : « 42 »
</script>
```

Exercise 3

Using variables as well as *prompt* and *alert* functions, code a program that will ask the user to supply a number that will be put to square.

Font instance, if the user supplies « 3 » the result displayed would be 3 squared, « 9 ».

Assignment 3

Using variables as well as *prompt* and *alert* functions, code a program asking his dining table's diameter.

Using this information, the message window should display the table circumference and surface area.

The formulas used to obtain the results are the followings :

Circumference :

$$2\pi r$$

2 x 3,14 x (diameter divided by 2)

Surface area :

$$\pi r^2$$

3,14 x (result of diameter divided by 2 multiplied by itself)