

1- Les éléments du langage

```
a = 3 ;
b = "hello world" ;
c = 'hello from Mars' ;
a = b ;
a = 3*4*3+2 ;
```

```
+= y → x = x + y ;
x -= y → x = x - y ;
x++ → x = x + 1 ; x--
→ x = x - 1 ;
```

Opérations**2.2.1 Les opérateurs arithmétiques**

+ - * /

Exemples :

```
s = s + 1 ;
b = c - d ;
s = s ^ 5 ;
res = (3 + 5) * (23 / 4) - 3 ;
```

Autres opérateurs :

Opérateur	Exemple	signification
+=	i+=1	i=i+1
-=	i-=1	i=i-1
=	i=2	i=i*2
/=	i/=2	i=i/2

Les opérateurs de comparaison

Les opérateurs de comparaison servent à exprimer des conditions. Attention à ne pas confondre l'opérateur de comparaison == avec le signe = d'assignation.

égal à	==
différent de	!=
supérieur à	>
supérieur ou égal à	>=
inférieur à	<
inférieur ou égal à	<=

Les opérateurs logiques

ET logique	&&
OU logique	
NON logique	!

Les structures conditionnelles :

Choix avec une alternative

Formulation générale :	Exemple :
<pre>if (<condition>) { <action>; <action>; ... }</pre>	<pre>if (j == 5) { Somme = Somme +1; }</pre>

Choix avec deux alternatives

Formulation générale :	Exemple :
<pre>if (condition>) { <action>; <action>; ... } else { <action> <action> ... }</pre>	<pre>if (j == 5) { Somme = Somme + 1 ; } else { Somme = Somme - 1 ; }</pre>

Les boucles

For	
Formulation générale :	Exemple :
<pre>for (<expression de depart> ; <condition de continuation> ; <incrementation>) { <action> ... }</pre>	<pre>i = 0 ; for (i = 1 ; i <= 5 ; i = i + 1) { somme = somme + i ; }</pre>

Fonctions et procédures

syntaxe	Exemple :
<pre>function nom(param1,param2,...,paramN) { }</pre>	<pre>function carre(nombre) { resultat = nombre * nombre ; return(resultat) ; }</pre>

l'objet String

La syntaxe de la propriété length est la suivante :

```
x = nom_de_la_chaine.length;
```

Les méthodes de l'objet String

Méthode	description
Chaine.charAt(position)	Retourne le caractère situé à la position donnée en paramètre
concat(chaîne1 , chaîne2 [, ...])	Permet de concaténer les chaînes passées en paramètre, c'est-à-dire de les joindre bout à bout.
Chaine.indexOf(sous-chaîne , position)	Retourne la position d'une sous-chaîne (lettre ou groupe de lettres) dans une chaîne de caractère, en effectuant la recherche de gauche à droite, à partir de la position spécifiée en paramètre.
Chaine.lastIndexOf(sous-chaîne , position)	La méthode est similaire à <i>indexOf()</i> , à la différence que la recherche se fait de droite à gauche : Retourne la position d'une sous-chaîne (lettre ou groupe de lettres) dans une chaîne de caractère, en effectuant la recherche de droite à gauche, à partir de la position spécifiée en paramètre.
Chaine.substr(position1 , longueur)	La méthode retourne une sous-chaîne commençant à l'index dont la position est donnée en argument et de la longueur donnée en paramètre.
Chaine.substring(position1 , position2)	La méthode retourne la sous-chaîne (lettre ou groupe de lettres) comprise entre les positions 1 et 2 données en paramètre.
Chaine.toLowerCase()	Convertit tous les caractères d'une chaîne en minuscule.
Chaine.toUpperCase()	Convertit tous les caractères d'une chaîne en majuscule.

L'objet Math

Les particularités de l'objet Math

```
Math.cos(1);
```

Les méthodes et propriétés standards de l'objet Math

Méthode	Description	Exemples
abs()	Cette méthode renvoie la valeur absolue d'un nombre, il renvoie donc le nombre s'il est positif, son opposé (positif) s'il est négatif	<code>x = Math.abs(3.26);</code> donne <code>x = 3.26</code> <code>x = Math.abs(-3.26);</code> donne <code>x = 3.26</code>
ceil()	Renvoie le plus petit entier supérieur ou égal à la valeur donnée en paramètre	<code>x = Math.ceil(6.01);</code> donne <code>x = 7</code> <code>x = Math.ceil(3.99);</code> donne <code>x = 4</code>
floor()	La méthode <i>floor()</i> retourne le plus grand entier inférieur ou égal à la valeur donnée en paramètre.	<code>x = Math.floor(6.01);</code> donne <code>x = 6</code> <code>x = Math.floor(3.99);</code> donne <code>x = 3</code>
round()	Arrondit à l'entier le plus proche la valeur donnée en paramètre.	<code>x = Math.round(6.01);</code> // donne <code>x = 6</code> <code>x = Math.round(3.80);</code> // donne <code>x = 4</code>
max(Nombre1, Nombre2)	<i>max()</i> renvoie le plus grand des deux nombres donnés en paramètre	<code>X= Math.max(4,8)</code> // <code>x=8</code>
min(Nombre1, Nombre2)	<i>min()</i> renvoie le plus petit des deux nombres donnés en paramètre	<code>X= Math.min(4,8)</code> // <code>x=4</code>
pow(Valeur1, Valeur2)	Retourne le nombre <i>Valeur1</i> à la puissance <i>Valeur2</i>	<code>X= Math.pow(2,3)</code> // <code>x=8</code>
random()	La méthode <i>random()</i> renvoie un nombre pseudo-aléatoire compris entre 0 et 1.	
sqrt(Valeur)	Renvoie la racine carrée du nombre passé en paramètre	<code>X= Math.sqrt(25)</code> // <code>x=5</code>