



# JavaScript

SD BUT 2 R3-13 Technologies Web

IUT Lumière

---

# Plan

Introduction

HTML - CSS

**JavaScript**

Introduction

**JavaScript et HTML**

Les boîtes de messages de JavaScript

Variables

Opérateurs

Structures de contrôle

Évènements

JavaScript et les objets

Référentiel

---

# Intérêt

Enrichir le HTML et le rendre **interactif**

# Comment ?

## Exemple

```
<html>  
  
    <head>  
        <title>Page statique</title>  
    </head>  
  
    <body>  
        <div>  
            Nous sommes le 13/12/2022  
        </div>  
    </body>  
  
</html>
```

## Exemple

```
<html>  
  
    <head>  
        <title>Page dynamique</title>  
    </head>  
  
    <body>  
        <script type = "text/javascript">  
            var date = new Date();  
            document.write("Nous sommes le ", date);  
        </script>  
    </body>  
  
</html>
```

---

# Script

- Portion de code qui vient **s'insérer** dans une page HTML
- Le code du script n'est toutefois pas visible dans la fenêtre du navigateur car il est compris entre des balises spécifiques qui signalent au navigateur qu'il s'agit d'un script écrit en langage JavaScript
- Interprété du **coté client** (par le navigateur)

## Interne

```
<script type = "text/javascript">  
    code javascript  
</script>
```

## Externe

```
<script type = "text/javaScript" src = "url/script.js">  
</script>
```

---

## En-tête vs Corps

- Les éléments situés dans le **corps** du fichier html s'exécutent au fur et à mesure du chargement de la page
- Les éléments situés dans l'**en-tête** se comportent comme des déclarations; ils ne s'exécutent pas directement

# JavaScript n'est pas Java

Java développé par Sun en 1995

(Détenu maintenant par Oracle)



Langage complexe

Ecriture de logiciels portables

Code source compilé avant son exécution

JavaScript développé par Netscape en 1995

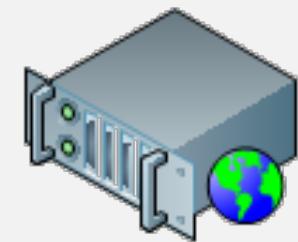
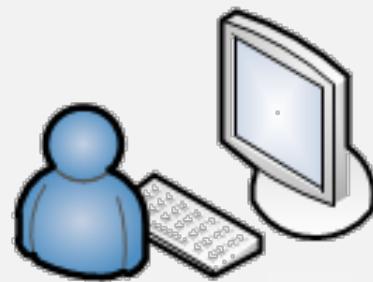
Langage simple, applications limitées

JavaScript est interprété



---

# Vue d'ensemble : *Rappel + JS*



---

## Notions d'objet et d'évènement

Langage de **script** orienté **objet** et **événementiel**

---

# Ce que permet JavaScript

Vérification de saisie dans des formulaires  
(*Adresse, ...*)

Calculs simples suite à des saisies dans les formulaires  
(*Tarifs, Calculatrice*)

Gestion des dates et des heures  
(*Date du jour, Date de mise à jour, Calendriers*)

...

# Ce que permet JavaScript

## *Exemple* Vérification de saisie

The screenshot shows a web form with fields for First Name, Last Name, EMail, Phone, Address, and Country. A dropdown menu for Country is set to "[choose yours]". An "OK" button is at the bottom. A modal dialog box titled "Windows Internet Explorer" displays three error messages: "Please enter your First Name", "Please enter your Last Name", and "Email : Required Field".

First Name

Last Name

EMail

Phone

Address

Country

Submit

Please enter your First Name  
Please enter your Last Name  
Email : Required Field

OK

# Ce que permet JavaScript

*Exemple* Calculs suite à des saisies

## Calcul de moyenne 1

Note du DEVOIR 1 :

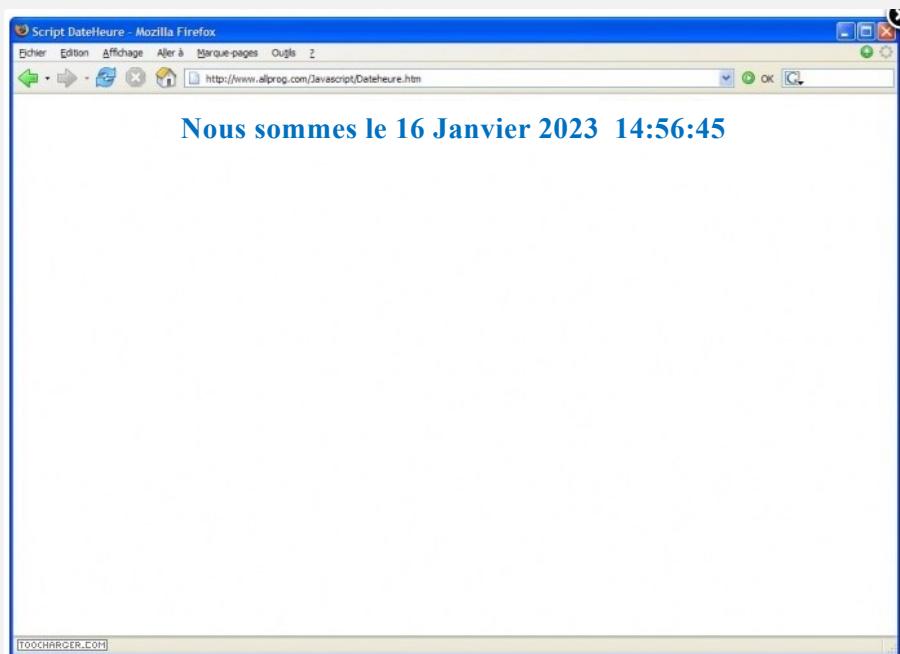
Note du DEVOIR 2 :

Note du DEVOIR 3 :

Moyenne obtenue :

# Ce que permet JavaScript

*Exemple* Dates et heures



---

# Plan

Introduction

HTML - CSS

**JavaScript**

Introduction

**JavaScript et HTML**

Les boîtes de messages de JavaScript

Variables

Opérateurs

Structures de contrôle

Évènements

JavaScript et les objets

Référentiel

# Positionnement dans le code HTML

Deux balises indispensables

```
<script>  
...  
</script>
```

Deux modes d'insertion

- Insertion directe dans la page
- Insertion comme un module externe

## Insertion directe dans le corps du fichier HTML

### Positionnement dans HTML

Mon premier script qui écrit la date

Hello monde, cliquer sur OK pour voir la date du jour!

OK



Mon premier script qui écrit la date

Sun Jan 15 2023 21:04:43 GMT+0100 (heure normale d'Europe centrale)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Mon premier script</title>
  </head>
  <body>
    <h1 align="center">Mon premier script qui écrit la date</h1>
    <script type="text/JavaScript">
      alert("Hello monde, cliquer sur OK pour voir la date du jour!");
      document.write("Aujourd'hui nous sommes le: "+Date());
    </script>
  </body>
</html>
```

# Positionnement dans HTML

```
document.write("Aujourd'hui nous sommes le: "+Date());
```

```
-8" />  
script</title>
```



affiche\_date.js

```
<body>  
  <h1>Mon second script qui écrit la date</h1>  
  <script type="text/javascript">  
    src="affiche_date.js">  
  </script>  
</body>  
</html>
```

**Mon premier script qui écrit la date**

Sun Jan 15 2023 21:04:43 GMT+0100 (heure normale d'Europe centrale)

# Positionnement dans HTML

Mon troisième script qui écrit la date

Afficher la date

Mon Feb 10 2014 12:01:50 GMT+0100

```
<!DOCTYPE html>
<html>
  <head> <meta charset="UTF-8" />
    <title>Mon premier script</title>
    <script type="text/javascript">
      function afficheDate() {
        document.write(Date());
      }
    </script>
  </head>
  <body>
    <h1>Mon troisième script qui écrit la date</h1>
    <button type="button"
      onclick="afficheDate()">Afficher la date</button>
  </body>
</html>
```

---

# Récapitulatif

Utilisation de la balise `<script>...</script>`:

- déclaration de fonctions dans l'en-tête HTML (entre `<head>` et `</head>`)
- appel d'une fonction ou exécution d'une commande JavaScript dans `<body>...</body>`
- insertion d'un fichier externe ('.js')

---

# Plan

Introduction

HTML - CSS

**JavaScript**

Introduction  
JavaScript et HTML  
**Les boîtes de messages de JavaScript**  
Variables  
Opérateurs  
Structures de contrôle  
Évènements  
JavaScript et les objets  
Référentiel

---

# Les boîtes de messages

Trois types de fenêtre avec pour rôle :

- D'**avertir** l'utilisateur
- De le confronter à un choix, **choisir**
- De lui **demander** de compléter un champ

# Avertir l'utilisateur

## Alerte : Méthode `alert()`

- Afficher à l'utilisateur des informations simples
- Une fois que ce dernier a lu le message, il doit cliquer sur OK pour faire disparaître la boîte

```
...
<body>
    <script type="text/JavaScript">
        alert("Hello monde, cliquer sur OK pour voir la date du jour!");
        document.write(Date());
    ... </script>
</body>
```



# Confronter l'utilisateur à un choix

## Confirmation : Méthode `confirm()`

Permet à l'utilisateur de choisir entre les boutons OK et Annuler.

```
...
<body>
    <h3>Voir la date (ou non!) avec le popup de la méthode confirm()</h3>
    <script type="text/JavaScript">
        var c;
        c = confirm("Voulez-vous voir la date du jour ?");
        if (c==true)
            document.write(Date());
        else
            document.write("Vous avez choisi de ne pas voir la date !!!");
    </script>
</body>
...
```

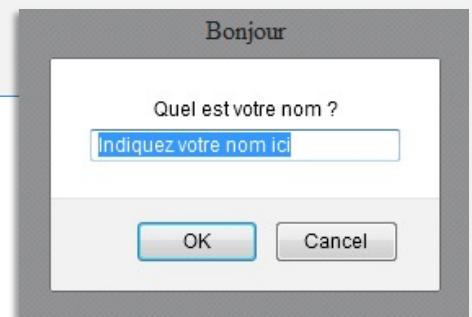


# Demande de saisie

## Demande de saisie : Méthode `prompt()`

Permet à l'utilisateur de taper son propre message en réponse à la question posée

```
...
<body>
Bonjour
    <script type="text/JavaScript">
        var p;
        p = prompt("Quel est votre nom ?","Indiquez votre nom ici");
        document.write("Bonjour" + p + ". Comment allez-vous aujourd'hui ?");
    </script>
</body>
...
```



Bonjour Chantal. Comment allez-vous aujourd'hui ?

---

# Plan

Introduction

HTML - CSS

**JavaScript**

Introduction  
JavaScript et HTML  
Les boîtes de messages de JavaScript

**Variables  
Opérateurs**

Structures de contrôle  
Évènements  
JavaScript et les objets  
Référentiel

# Les variables

```
...
<body>
    <script type="text/JavaScript">
        var nom;
        prenom = "Mickaël";
        var Prenom;
        document.write("1. " + prenom + "<br/>");
        prenom = 55;
        document.write("2. " + prenom + "<br/>");
        document.write("3. " + Prenom + "<br/>");
        document.write("4. " + prenom + "<br/>");
        /*Je me demande
        ce que cela va donner :*/
        //Pas vous ?
    </script>
</body>
...
```

1. Mickaël
2. 55
3. undefined
4. 55

**js7\_Variables\_Cha.html**

# Les opérateurs

- de calcul
  - +, -, \*, /, %, =
- de comparaison
  - ==, <, <=, >, >=, !=
- associatifs
  - +=, -=, \*=, /=
- logiques
  - &&, ||
- incrémentation
  - ++, --

Assignment operators assign values to JavaScript variables.

Operator	Example	Same As
=	<code>x = y</code>	<code>x = y</code>
<code>+=</code>	<code>x += y</code>	<code>x = x + y</code>
<code>-=</code>	<code>x -= y</code>	<code>x = x - y</code>
<code>*=</code>	<code>x *= y</code>	<code>x = x * y</code>
<code>/=</code>	<code>x /= y</code>	<code>x = x / y</code>
<code>%=</code>	<code>x %= y</code>	<code>x = x % y</code>

Comparison operators are used in logical statements to determine equality or difference between variables or values.

Given that  $x = 5$ , the table below explains the comparison operators:

Operator	Description	Comparing	Returns
==	equal to	$x == 8$	false
		$x == 5$	true
		$x == "5"$	true
!=	not equal	$x != 8$	true
>	greater than	$x > 8$	false
<	less than	$x < 8$	true
>=	greater than or equal to	$x >= 8$	false
<=	less than or equal to	$x <= 8$	true

# Logical Operators

Logical operators are used to determine the logic between variables or values.

Given that **x = 6** and **y = 3**, the table below explains the logical operators:

Operator	Description	Example	Try it
<code>&amp;&amp;</code>	and	<code>(x &lt; 10 &amp;&amp; y &gt; 1)</code> is true	<a href="#">Try it »</a>
<code>  </code>	or	<code>(x == 5    y == 5)</code> is false	<a href="#">Try it »</a>
<code>!</code>	not	<code>!(x == y)</code> is true	<a href="#">Try it »</a>

# Plan

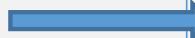
---

- Introduction
- HTML - CSS
- **JavaScript**

Introduction  
JavaScript et HTML  
Les boîtes de messages de JavaScript  
Variables  
Opérateurs

## **Structures de contrôle**

Evènements  
JavaScript et les objets  
Référentiel



---

# Structures de contrôle

- for, while, if else, switch case

## for Répéter des instructions un nombre déterminé de fois - Ex1

Ci-dessous la suite de 0 à 5 grâce à le boucle for !  
012345

```
...
<head>
...
</head>
<body>
    <h1> Ci-dessous la suite de 0 à 5 grâce à le boucle for !</h1><br />
    <script>
        for (i=0;i<=5;i++) document.write(i);
    </script>
</body>
...
```

## for Répéter des instructions un nombre déterminé de fois – Ex2

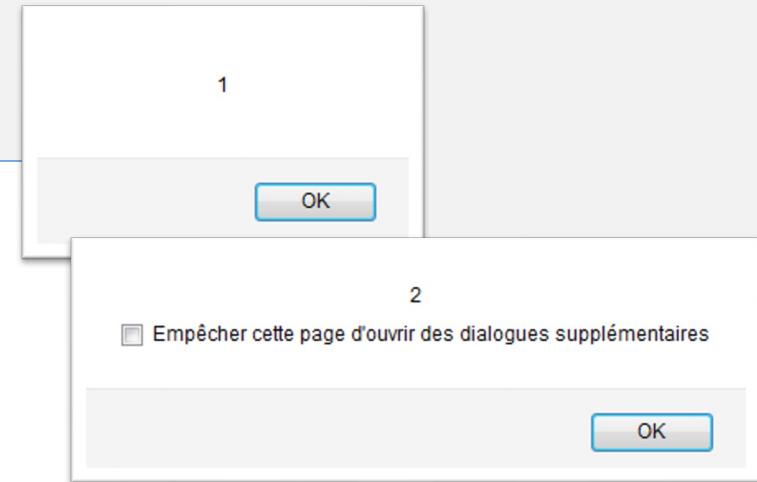
```
<head>
    <meta charset="UTF-8" />
    <script>
        function boucleFor() {
            for (i=0;i<=5;i++) document.write(i);
        }
    </script>
</head>
<body>
    <h1>Suite de 0 à 5 avec boucle for dans une fonction et
    évènement onclick sur un bouton !</h1>
    <input type="button" onclick="boucleFor()" value="Test" />
</body>
```

## for Répéter des instructions un nombre déterminé de fois – Ex3

```
...
<head>
    <script type="text/javascript">
        function showAlert(i) {
            alert("Je suis la popUp numero " + i);
        }
        function testLoop() {
            var i=0;
            for (i=0;i<=5;i++) showAlert(i);
        }
    </script>
</head>
<body>
    <h1>Mon serial-PopUp</h1>
    <input type="button" onclick="testLoop()" value="Test"/>
</body>
...
```

## while Répéter des instructions tant qu'une condition est VRAIE

```
...
<body>
    <script>
        var i = 1;
        while(i < 10) {
            alert(i);
            i++;
        }
    </script>
</body>
...
```



## if ... else Diriger l'exécution du script selon des conditions



```
...
<body>
    <script>
        var age = prompt("Quel âge avez-vous ? (en années)");
        if(age >= 18)
            alert("Vous êtes majeur");
        else
            alert("Tu es mineur");
    </script>
<body>
...
```

## switch case Faire plusieurs tests de valeur sur le contenu d'une même variable

```
switch ( Variable ) {  
    case Valeur1:  
        Liste d'instructions;  
        break;  
    case Valeur2:  
        Liste d'instructions;  
        break;  
    case ValeurX:  
        Liste d'instructions;  
        break;  
    default:  
        Liste d'instructions;  
        break;  
}
```

- Les parenthèses qui suivent le mot clé **switch** indiquent une expression dont la valeur est testée successivement par chacun des cas
- Lorsque l'expression testée est égale à une des valeurs suivant un **case**, la liste d'instruction qui suit celui-ci est exécutée
- Le mot clé **break** indique la sortie de la structure conditionnelle
- Le mot clé **default** précède la liste d'instructions qui sera exécutée si l'expression n'est jamais égale à une des valeurs
- Opération possible mais compliquée avec des *if* imbriqués
- Simplifie le test de plusieurs valeurs d'une variable

---

# Plan

## Introduction HTML - CSS **JavaScript**

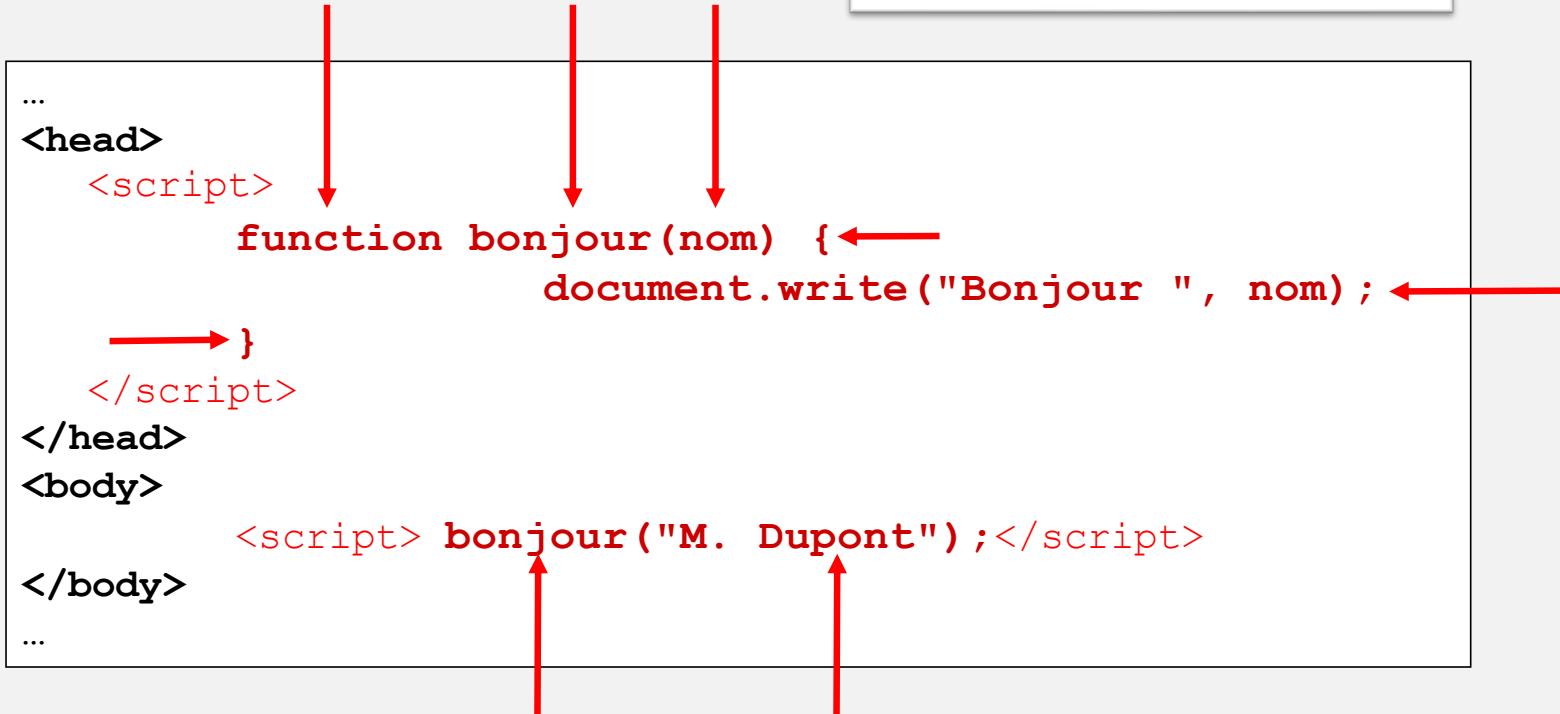


Introduction  
JavaScript et HTML  
Les boîtes de messages de JavaScript  
Variables  
Opérateurs  
Structures de contrôle  
**Les fonctions**  
Evènements  
JavaScript et les objets  
Référentiel

# Formalisons

Bonjour M. Dupont

```
...
<head>
  <script>
    function bonjour(nom) {←
      document.write("Bonjour ", nom); ←
    }
  </script>
</head>
<body>
  <script> bonjour("M. Dupont");</script>
</body>
...
```



# Formalisons

```
...
<head>
    <script>
        function somme(unNombre) {
            unNombre = unNombre + 12;
            → return unNombre;
        }
    </script>
</head>
<body>
    <script>
        var resultat;
        → resultat = somme(1);
        document.write(resultat);
    </script>
</body>
...
```

# Entrainement

```
<head>
    <script>
        function volumeSphere(rayon) {
            return 4/3 * Math.PI * Math.pow(rayon,3);
        }
    </script>
</head>
```

$$\frac{4}{3} \pi R^3$$

```
        function calculerPrix(PrixUnitaire, NbArticles) {
            return PrixUnitaire * NbArticles;
        }
    </script>
</body>
```

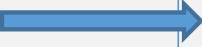
Tu dois payer 600 Euros.  
Le volume d'une sphère de rayon 1 est 4.1887902047863905

```
        <script>
            document.write("Tu dois payer ", calculerPrix(150,4)," Euros.<br>");
            document.write("Le volume d'une sphère de rayon 1 est ",
                           volumeSphere(1), "<br>" );
        </script>
</body>
```

js9\_fonctions\_5.html

# Plan

## Introduction HTML - CSS **JavaScript**

- 
- Introduction
  - JavaScript et HTML
  - Les boîtes de messages de JavaScript
  - Variables
  - Opérateurs
  - Structures de contrôle
  - Les fonctions
  - Evènements**
  - JavaScript et les objets
  - Référentiel

# Notion d'objet et d'évènement

## Evénement:

Action de l'utilisateur

L'événement par excellence: *le clic de souris*

Peut donner lieu à une interactivité

Grâce à JavaScript :

*Possible d'associer des fonctions, des méthodes à des événements*

Langage de **script orienté objet et évènementiel**

# La notion d'évènement

- Les **événements** permettent de déclencher une **fonction** selon qu'une **action** s'est produite ou non
- Par exemple, on peut faire apparaître une fenêtre **alert()** lorsque l'utilisateur **survole une zone** (un élément HTML) d'une page Web

Nom de l'événement	Action pour le déclencher
click	Cliquer (appuyer puis relâcher) sur l'élément
dblclick	Double-cliquer sur l'élément
mouseover	Faire entrer le curseur sur l'élément
mouseout	Faire sortir le curseur de l'élément

# Utiliser les évènements

```
...  
<head>  
...  
</head>  
  
<body>  
<input type="button" value="Cliquer ici" onClick="alert('Clic')" />  
  
<span style="color:blue" onclick="alert('Hello !')">Cliquez-moi !</span>  
</body>  
...  
Attribut de span | Valeur de l'attribut
```

---

# Plan

## Introduction HTML - CSS **JavaScript**

Introduction  
JavaScript et HTML  
Les boîtes de messages de JavaScript  
Variables  
Opérateurs  
Structures de contrôle  
Évènements  
**JavaScript et les objets**  
Référentiel



# Notion d'objet et d'évènement

**Elément/Entité qui:**

Possède des caractéristiques spécifiques  
Des manières différentes de les utiliser

**Des objets de base:** *Chaînes de caractères, Dates, Images*



Langage de **script** orienté **objet** et **évenementiel**

# La notion d'objet en JavaScript

Reprendons l'exemple  
de la diapo 17 :  
`document.write(...)`

Mon premier script qui écrit la date

Hello monde, cliquer sur OK pour voir la date du jour!

OK

écrire la date  
`document.write(Date()) ;`  
de mon document Web  
qui est un objet  
La fonction write()  
>  
  
charset="UTF-8" />  
Mon premier script</title>  
  
<h1 align="center">Mon premier script qui écrit la date</h1>  
<script type="text/JavaScript">  
alert("Hello monde, cliquer sur OK pour voir la date du jour!");  
**document.write(Date());**  
</script>  
</body>  
</html>

Mon premier script qui écrit la date

Sun Jan 15 2023 21:04:43 GMT+0100 (heure normale d'Europe centrale)

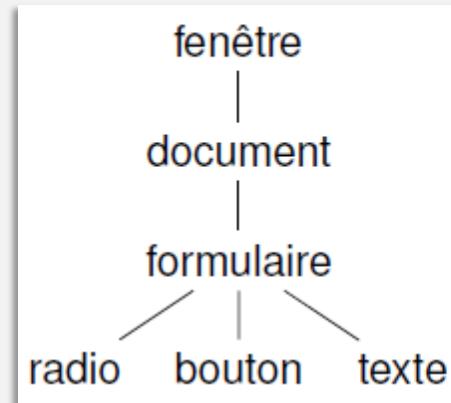
# Les objets du navigateur et leur hiérarchie

JavaScript divise **une page Web en objets**

JavaScript permet d'accéder à ces objets, d'en retirer des informations, de les manipuler

## Exemple de hiérarchie d'objets

- une fenêtre
- un document HTML
- un formulaire HTML
- des boutons radio, un bouton classique et une zone de texte



# Que contiennent les objets ?

- **Les propriétés**

Variables relatives l'objet

Contiennent des informations nécessaires au fonctionnement de l'objet, qui caractérisent l'objet

**Exemple:** longueur d'un **objet Chaîne de caractères**

- **Les méthodes** (*fonctions JavaScript*)

Permettent de réaliser des opérations sur l'objet

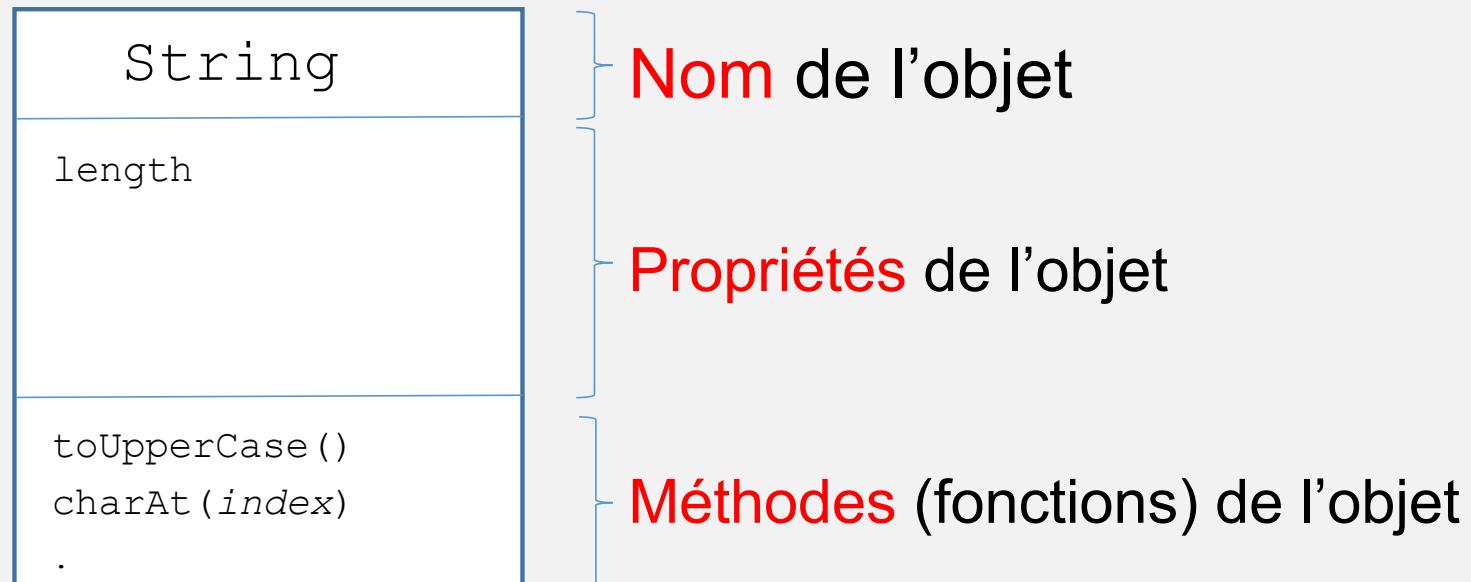
**Exemple** sur un **objet Chaîne de caractères**: une méthode permet de mettre une chaîne de caractères en majuscules

- **Le constructeur**

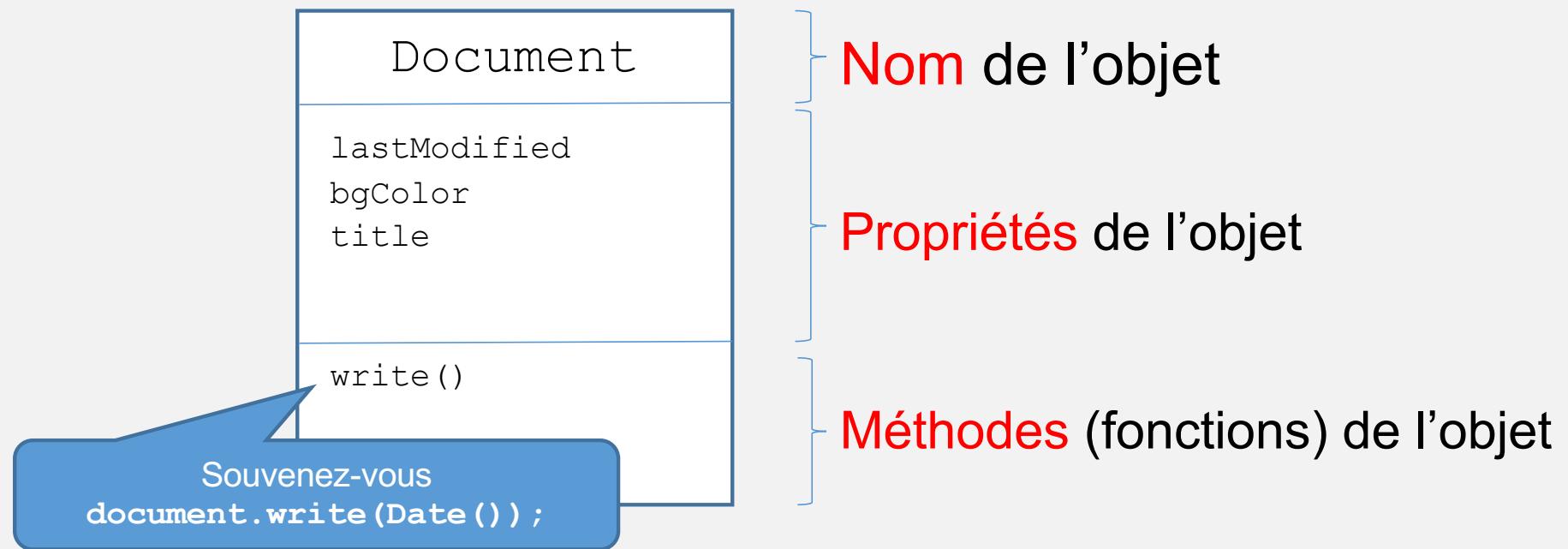
Fonction qui est exécutée pour créer un nouvel objet

# Représentation des objets

**Exemple :** objet Chaîne de caractères



# Représentation des objets



# Les objets et leurs propriétés

Une **propriété** est un attribut, une caractéristique, une description de l'objet

Les objets ont des propriétés personnalisées

Pour accéder aux propriétés, on utilise la syntaxe:

`nom-de-l'objet.nom-de-la-propriété`

Exemple: donner la couleur de fond d'un document

Exemple: Connaître le titre du document:

`document.bgColor='blue'`

Une méthode de  
l'objet document

`document.title`

Une propriété de  
l'objet document

`document.write(document.title);`

Et si je veux afficher  
le titre sur ma page ?

# Recherche de Propriétés

*Exemple avec l'objet String*

String

length

charAt(*index*)

.

.

.

```
var pays = new String('Vietnam');
```

Comment obtenir la longueur de la chaîne de caractère contenue dans la variable pays ?

pays.length

# Les méthodes des objets

*Ex. d'utilisation avec l'objet String*

String

length

charAt (*index*)

.

.

.

```
var pays = new String('Vietnam');
```

```
pays.charAt(1) ?
```

Retourne le caractère situé à la position *index*

# Création d'objets et initialisation

Avec *utilisation du constructeur*

Opérateur ***new*** + Nom du ***constructeur*** servant à initialiser l'objet

```
var maintenant = new Date ()
```

*représente la date et l'heure courante*

```
var nouvel_an = new Date (2023, 01, 16)
```

*représente le 16 Janvier 2023*

Propriétés de l'objet

# D'autres objets !



Nous avons vu : les  
objets du navigateur



Il y a aussi :  
les objets du noyau!

**Math** : Manipuler des fonctions mathématiques

**Date** : Manipuler des dates, Créer des dates

**Array** : Créer des tableaux, Manipuler des tableaux (*ajouter, supprimer, extraire des éléments d'un tableau, les trier*)

**String** : Créer des chaînes de caractères, Manipuler des chaînes de caractères

# Définition de Propriétés

//Créé un objet

```
var livre = new Object();
```

//Définit une propriété dans l'objet (en lui affectant une valeur)

```
livre.titre = "JavaScript - La référence";
```

//Définit d'autres propriétés. Notez l'emboîtement des objets

```
livre.chapitre1 = new Object();
livre.chapitre1.titre = "Introduction à JavaScript";
livre.chapitre1.pages = 19;
```

# Recherche de Propriétés

//Lit certaines valeurs de propriétés depuis l'objet

```
Alert("Plan:" + livre.titre + "\n" +
      "Chapitre 1" + livre.chapitre1.titre);
```

# Les constructeurs (1)

**Il existe:** Des objets prédéfinis avec des fonctions constructeur prédéfinies:  
Object(), Date(), String()

**Possibilité:** De définir ses propres objets

Pour cela: écrire des constructeurs

Un constructeur définit une classe d'objets

**Exemple**

```
// La classe Rectangle  
// Possède 2 propriétés  
// Voici son constructeur
```

```
function Rectangle ( L, l) {  
    this.largeur = l;  
    this.longueur = L ;  
}
```

# Les constructeurs (2)

## Création d'un nouvel objet Rectangle avec new + nom constructeur

```
// rect1 et rect2 : deux nouveaux objets Rectangle  
// rect1 et rect2 : sont des instances de la classe Rectangle  
// Les objets sont créés avec l'opérateur new  
// Les valeurs de leurs 2 propriétés sont spécifiées
```

```
var rect1 = new Rectangle(2 , 4);  
var rect2 = new Rectangle(8.5 , 11);
```

## Exemple d'utilisation

```
// Ecrire sur la page Web, la largeur de l'objet rect1 et la longueur de l'objet rect2  
  
document.write("largeur = "+ rect1.largeur);  
document.write("longueur = "+ rect2.longueur);
```

# Les méthodes des objets (1)

Définition d'une fonction *qui sera appelée par l'intermédiaire d'un objet*

// la fonction suivante calcule l'aire du rectangle

```
function calcule_aire () {  
    return this.largeur * this.longueur  
}
```

## Les méthodes des objets (2)

```
/*
```

*On peut se servir de la fonction calcule\_aire() pour définir une méthode par affectation de la fonction à une propriété de l'objet Rectangle*

```
*/
```

```
rect2.aire = calcule_aire ;
```

```
// invoque la nouvelle méthode comme ceci
```

```
var a2 = rect2.aire() ;
```

```
//on peut afficher la valeur de l'aire du rectangle:
```

```
document.write("L'aire du rectangle rect2 =" + a2);
```

---

## Exemple: Un objet Personne

*On veut créer un objet Personne*

*Une personne a un nom, un prénom, une adresse, un code postal, une ville, un numéro de téléphone, une année de naissance et un nombre d'enfant*

*Ecrivez le constructeur : Propriétés et Méthodes*

```
//déclaration des fonctions-méthodes  
function calculer_age() { ..... }  
  
// Constructeur de l'objet Personne  
function Personne(n, p, ad, cp, v, t, a, nb) {  
    // Definition des propriétés  
    this.nom = n ;  
    this.prenom = p ;  
    this.adresse=ad ;  
    this.codePostal=cp;  
    this.ville=v;  
    this.numTel=t;  
    this.annee_naissance = a;  
    this.nb_enfant = nb ;  
  
    // Définition des méthodes  
    this.age = calculer_age ; }
```

---

# Référentiel

<http://www.toutjavascript.com/main/index.php3>

<http://www.w3schools.com/js/>