



# JavaScript

## Éléments du langage

Ce cours s'est inspiré librement du cours de CHENU Guy



# Introduction

---

## ■ JavaScript

- Langage de script incorporé dans le HTML
- Historiquement, premier langage de script pour le Web
- Apporte des améliorations au HTML
  - HTML permet d'écrire
  - JavaScript permet de programmer, c'est-à-dire de gérer l'information

## ■ Qualités :

- Disponible sur les navigateurs actuels et gratuit

## ■ Défauts :

- Interprété et donc très lent, pas de débogueur



# Introduction

---

## ■ A quoi ressemble un script ?

- C'est une 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 (ou tags) spécifiques qui signalent au navigateur qu'il s'agit d'un script écrit en langage JavaScript
- Balises annonçant le code Javascript :

```
<SCRIPT language="Javascript">
```

**Placez ici le code de votre script**

```
</SCRIPT>
```



# Introduction

---

- **Code interprété ou compilé ?**
  - Dès que le navigateur rencontre la balise `<script>` il passe la main à l'interprète du langage appelé
  - Votre navigateur interprétera votre script, puis l'exécutera
- **Que mettre dans le script ?**
  - Des variables et instructions, organisées selon votre algorithme, c'est-à-dire selon le résultat que vous souhaitez obtenir
- **Où placer les scripts ?**
  - **Au début, dans le conteneur `<body>...</body>`**
    - Dès que vous serez capables de comprendre les fonctions, nous les placerons le plus souvent dans le conteneur `<head>...</head>`



# Introduction

---

- Masquage du script pour les anciens navigateurs
  - Attention :
    - Comme Javascript a été inventé après l'arrivée des navigateurs, ceux-ci peuvent ne pas le reconnaître
    - On place alors le code dans le body **en commentaire**

```
<SCRIPT language="Javascript">
```

```
<!--
```

```
    Placez ici le code de votre script
```

```
//-->
```

```
</SCRIPT>
```



# Éléments du langage

---

## ■ Les commentaires

- Pour mettre en commentaires toute une ligne, on utilise le double slash:

`//` Tous les caractères derrière le `//` sont ignorés

- Pour mettre en commentaire une partie du texte (éventuellement sur plusieurs lignes) on utilise le `/*` et le `*/`:

`/*` Toutes les lignes comprises entre ces repères sont ignorées par l'interpréteur de code `*/`



# Lire/Écrire

---

- **prompt()**

- ouvre une boîte de dialogue avec une zone de saisie et 2 boutons : OK et Annuler, rend l'information lue

- **alert ()**

- permet d'écrire un message dans une fenêtre
- Exemple : **lire-ecrire.html**

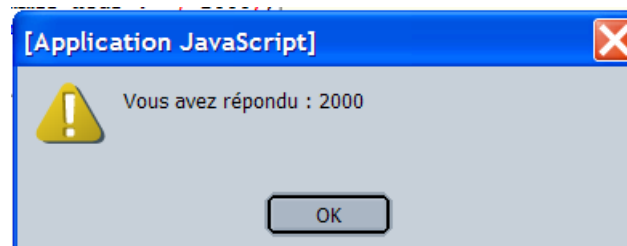
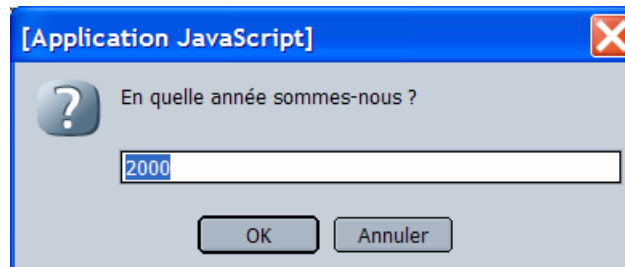
```
<html>
<head><title>Programme In1</title></head>
<body>
  <script language="JavaScript">
    <!--
      annee=prompt('En quelle année sommes-nous ? ', 2000);
      alert('Vous avez répondu : ' + annee);
    //-->
  </script>
</body>
</html>
```



# Lire/Ecrire

---

## ■ Résultat







# Lire/Écrire

---

- **document.write :**

- Permet d'écrire directement dans la fenêtre HTML
- Exemple : lire-ecrire2.html

```
<html>
  <head>
    <title>Programme Out1</title>
  </head>
  <body>
    <script language="JavaScript">
      <!--
        document.write('Vous avez le bonjour de
          JavaScript <br>');
      //-->
    </script>
  </body>
</html>
```



# Lire/Écrire

---

- `document.write` pour écrire le contenu de variable

Exemple : lire-ecrire-var.html

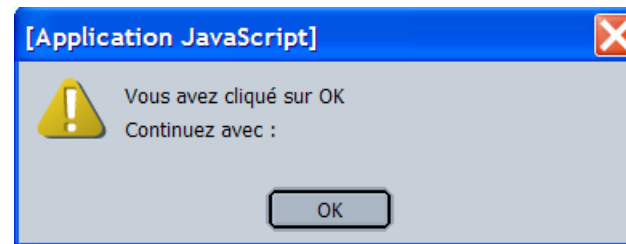
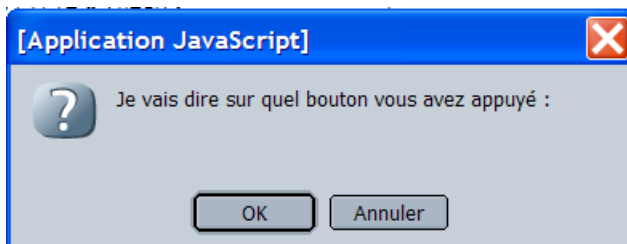
```
<html>
<head><title>Programme Out2</title></head>
<body>
<script language="JavaScript">
  <!--
    var jour = 21;
    var mois = 'juin';
    document.write(jour + ' ' + mois + ' : solstice');
  //-->
</script>
</body>
</html>
```

# Lire/Écrire

## ■ Confirm ()

- Cette méthode ouvre une boîte de dialogue avec 2 boutons : OK et Annuler. Elle permet :
  - d'envoyer une information, de recevoir un booléen
- Exemple : lire-ecrire-confirm.html

```
<script language="JavaScript">  
    if (confirm('Je vais dire sur quel bouton vous avez  
        appuyé : '))  
        {alert(' Sur OK \n Continuez avec :') }  
    else {alert(' Sur Annuler \n Sortez avec Ok !') };  
</script>
```



# Lire/Écrire

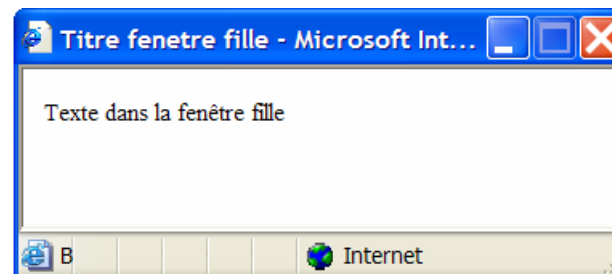
## ■ Écriture dans une fenêtre : `open()` + `document.write`

- Permet d'ouvrir une fenêtre et d'écrire dedans
- Exemple : lire-ecrire-open.html

```
<script>
```

```
filles=open("", "", 'height=50, width=300, status=yes');  
filles.document.write('<title>' + 'Titre fenetre fille' +  
'</title>');  
filles.document.write('Texte dans la fenetre fille');
```

```
</script>
```





# Variables

---

## ■ Déclaration

```
<script language="JavaScript">
```

```
  var date; // Déclaration sans affectation
```

```
  var compteur=0; // Déclaration avec affectation
```

```
  toto='coucou'; // Déclaration implicite par affectation
```

```
  var prem, second; // variables séparées par des virgules
```

```
</script>
```



# Types

---

## ■ Déclaration

```
<script language="JavaScript">  
    monNombre = new Number(); // Déclaration typée sans  
    affectation  
    e = new Number(2.71828); // Déclaration typée avec  
    affectation  
    var maChaine = new String(); //Déclaration de chaîne  
    var toto = new Boolean(true); //Déclaration de booléen  
</script>
```



# Types

## ■ Visibilité des variables :

- Bien que de même nom, les variables "x" sont indépendantes
- Exemple : variable-visibilite.html

```
<script language="JavaScript">
  var x = new String('abc');
  function commencer() {
    var x = new Number(123);
    x++;
    alert(x);
  };
  function suivre() {
    var x = new Boolean(true);
    alert(x);
  };
  x='ABC';
  commencer();
  suivre();
</script>
```



# Tableau

---

## ■ Array

- Le type des éléments : nombres, chaînes, booléens, ...
- La dimension 1, 2, ou 3, ... : tab(7) ; tab(x,y) ; tab(A,B,C) ; ...
- Les indices : souvent des nombres entiers
- Exemple : tableau.html

```
<script language=JavaScript>
```

```
// Tableau de chaînes, de dimension 1, indicé de 0 à 6 :
```

```
Jour=new Array(7);
```

```
Jour[0]='Dimanche' ;
```

```
Jour[1]='Lundi' ;
```

```
Jour[2]='Mardi' ; //...
```

```
Jour[6]='Samedi' ;
```

```
// En énumérant les éléments :
```

```
jour=new Array('dimanche','lundi','mardi', ...  
, 'vendredi','samedi');
```

```
document.write(Jour[1], ' ', Jour[0]); //affiche : Lundi  
dimanche
```

```
</script>
```





# Tableau

---

- **Array**

- Autre exemple : tableau2.html

```
<script language="JavaScript">
temp=new Array(30);
function relever() {
    for (i=1;i<=30;i++); // i commence à 1
    {
temp[i-1]=17+Math.floor(5*Math.random());
}; // i-1=0, donne le premier indice
};
function moyenner() {
    som=0;
    for (i=1;i<=30;i++) {
        som+=temp[i-1]; };
    return Math.round(10*som/30)/10;
};
relever();
alert('la moyenne du tableau est '+ moyenner());
</script>
```

## ■ Autre exemple (suite)

- Si on obtient NaN
  - Cela veut dire que la valeur n'est pas numérique
- Dans ce cas, utiliser :
  - `parseInt(moyenner())`
- Exemple : **affich-entier.html**

```
<html>
  <head>
    <title>Programme Out2</title>
  </head>
  <body>
    <script language="JavaScript">
      <!--
        function calcul() {
          var x=20; return x;
        };
        alert('la valeur de x =' + parseInt(calcul()));
      -->
    </script>
  </body>
</html>
```



# Opérateurs

## ■ Arithmétiques

### Binaires :

+	Addition	$2 + 3$ ; compteur + 1 . + opère aussi sur les chaînes. C'est un opérateur de concaténation : 'bon'+ 'jour' == 'bonjour'
-	Soustraction	$2 - 3$ ; rebours - 1
*	Multiplication	$2 * 3$ ; $a * b$
/	Division	$2 / 3$ ; $a / b$ Attention : nombre/0 rend null
%	Modulo	$13 \% 5$ ; Reste dans la division euclidienne $13 \% 5 = 3$

### Unaires :

-	Opposé	-monSolde
++	Incrémentation	$i++$ ; équivaut à $i = i + 1$ (Pratique) $++i$ ; l'incrément est fait avant d'utiliser la valeur de $i$
--	Décrément	$k--$ ; équivaut à $k = k - 1$ $--k$ ; la décrémentation est faite avant d'utiliser la valeur de $k$



# Opérateurs

## ■ De comparaison

- La comparaison se fait entre deux objets de même type
  - renvoie un booléen : true ou false

### **Binaires :**

<	Inférieur strict	<code>x &lt; 2 ;</code> <code>compteur &lt; 101</code> <code>'femme' &lt; 'homme'</code> renvoie true
>	Supérieur strict	<code>x &gt; 2 ;</code> <code>compteur &gt; x</code> <code>'chat' &gt; 'chien'</code> renvoie false
<=	Inférieur large	<code>x &lt;= 2 ;</code>
>=	Supérieur large	<code>x &gt;= 2 ;</code>
==	Egal ("identique à")	<code>x == 2 ;</code>
!=	Différent	<code>x != 2 ;</code>



# Opérateurs

## ■ Logique

- Opèrent sur des booléens et renvoient un booléen

### Binaires :

&&	ET	<code>bool1 &amp;&amp; bool2</code> true uniquement dans le cas : <code>true &amp;&amp; true</code>
	OU	<code>bool1    bool2</code> false uniquement dans le cas : <code>false    false</code>

### Unaire :

!	NON	<code>! bool1</code> <code>! true = false ; ! false = true</code>
---	-----	--



# Opérateurs

## ■ d'affectation

### Affectation simple :

=	Affectation simple	Total = HT+TVA ; JavaScript évalue HT+TVA puis le range dans Total Attention : $2 = \text{toto}+1$ n'est pas une affectation, et pire n'a pas de sens. Quant à $2 == \text{toto}$ c'est un booléen vrai ou faux
---	--------------------	---

### Affectation arithmétique :

+=	Addition	$x += y$ signifie $x = x + y$
-=	Soustraction	$x -= y$ signifie $x = x - y$
*=	Multiplication	$x *= y$ signifie $x = x * y$
/=	Division	$x /= y$ signifie $x = x / y$
%=	Modulo	$x \% = y$ signifie $x = x \% y$



# Conditionnelles

---

- **Syntaxe**

**if** (condition booléenne)

{Instructions à exécuter si "oui"}

**else**

{Instructions à exécuter si "non"};



# Conditionnelles

- Exemple : conditionnelle.html

```
<script language="JavaScript">
a = prompt('a=',2);
b = prompt('b=',6);
if (a==0)
    {if (b==0)
        {alert(a+'x='+b+ '\n' +'tout nombre est solution')}}
    else
        {alert(a+'x='+b+ '\n'+ '0x est différent de '+b +
'\n'+ 'Pas de solution') }
}
else
    {alert(a+'x='+b+ '\n' +'Une seule solution \n'+ 'x='+b/a) }
</script>
```





# Choix

---

- **Choix multiple**

```
switch(x) {  
    case 1 : instructions 1; break;  
    case 2 : instructions 2; break;  
    ...  
    case n : instructions 3; break;  
    default : instructions 4; break;  
};
```



# Choix

---

- Exemple : choix.html

```
function preferer(k) {  
    switch(k) {  
        case 1 : alert(1); break;  
        case 2 : alert(2); break;  
    };  
};  
preferer(2);
```



# Itération

---

- For

**for** (valeur de départ ; contrôle pour sortie ; progression )  
{ Instructions à itérer }

- Exemple

```
for (i=1;i<=5;i++) {  
    Instructions à répéter  
}
```



# Itération

---

- **Table de multiplication**

- Exemple : iteration.html

```
<html>
  <head><title>Programme iterer1 </title></head>
  <body>
    <script language="JavaScript">
      <!--
        var n=0;
        n=prompt('Quelle table ? ',7);
        for (i=1;i<10;i++) {
          document.write(i+' x '+n+' = '+n*i+'<br>');
        };
      //-->
    </script>
  </body>
</html>
```



# Itération sous contrôle

---

## ■ while

- Syntaxe :

```
while(condition) {  
    suite d'instructions;  
}
```

- Exemple :

```
i=0;  
while(i<=10) {  
    suite d'instructions;  
    i++;  
}
```



# Fonctions

---

- Syntaxe

```
function maFonction(x, toto) {  
    ... instructions; ...  
    return valeur du résultat;  
}
```

- Exemple

```
<script language="JavaScript">  
    function VolCylindre(r,h) {  
        pi=3.14159;  
        return pi*r*r*h;  
    }  
    document.write(VolCylindre(1,2));  
</script>
```



# Fonctions

---

- Autre exemple

```
<script language="JavaScript">  
var volume;  
function VolCylindre(r,h) {  
    pi=3.14159;  
    volume = pi*r*r*h;  
}  
VolCylindre(1,2);  
document.write(volume);  
</script>
```



# Fonctions prédéfinies

## ■ Dates

- Javascript peut gérer le temps :
- Aujourd'hui : **mardi 16 août 2005**, l'horloge de votre ordinateur donne **13h 24min**
- Il vous propose plusieurs fonctions prédéfinies

<u>Lecture Jour</u> <u>Lecture Heure</u>	<code>getDate()</code> , <code>getDay()</code> , <code>getMonth()</code> , <code>getFullYear()</code> <code>getHours()</code> , <code>getMinutes()</code> , <code>getSeconds()</code> <code>getTime()</code> , <code>getTimezoneOffset()</code>
<u>Écrire Jour</u> <u>Écrire Heure</u>	<code>setDate(j)</code> , <code>setMonth(m)</code> , <code>setYear(a)</code> <code>setHours(h)</code> , <code>setMinutes(min)</code> , <code>setSeconds(s)</code> <code>setTime(ms)</code>
<u>Conversion</u>	<code>parse()</code> , <code>toGMTString()</code> , <code>toLocaleString()</code>





# Fonctions prédéfinies

---

- Exemple : `date.html`

```
<html>
  <head> <title>Aujourd'hui</title> </head>
  <body> Date de votre machine :
  <script language="JavaScript">
    var aujourd_hui = new Date();
    document.write(aujourd_hui.getDate()+ ' / ');
    document.write(aujourd_hui.getMonth()+1+ ' / ');
    document.write(aujourd_hui.getFullYear());
  </script>
  </body>
</html>
```



# Fonctions prédéfinies

## ■ Mathématiques :

<u>Constantes</u>	PI , e , sqrt(2) , ... , ln(2) , ...
-------------------	--------------------------------------

<u>Trigonométrie</u>	cos() , sin() , tan() , acos() , asin() , atan()
<u>Algèbre</u>	abs() , random() , ceil() , floor() , max() , min() , round()
<u>Analyse</u>	sqrt() , pow() , log() , exp()



# Fonctions prédéfinies

## ■ mathématiques : utilisation

Math.PI	3.141592653589793
Math.SQRT2	1.4142135623730951
Math.SQRT1_2	0.7071067811865476
Math.E	2.718281828459045
Math.LN2	0.6931471805599453

	Méthode	Définition.	Exemple	Résultat
Math.LN10	Math.abs()	Valeur absolue. (Rend positif)	Math.abs(-2)	2
Math.LOG10E	Math.random()	Nombre pseudo-aléatoire entre 0 et 1.	Math.random()	0.4578877016100097
Math.LOG2E	Math.max()	Plus grand des deux.	Math.max(1,2)	2
	Math.min()	Plus petit des deux.	Math.min(1,2)	1
	Math.ceil()	Entier immédiatement supérieur.	Math.ceil(1.23)	2
	Math.floor()	Entier immédiatement inférieur.	Math.floor(1.23)	1
	Math.round()	Arrondi à l'entier le plus proche.	Math.round(1.23)	1



# Association avec un formulaire

---

- Utilisation dans un formulaire
  - Schéma d'utilisation
    - Body :
      - ❖ Contient la définition du formulaire
      - ❖ Il fait appel aux variables et fonctions définies dans le head
    - Head :
      - ❖ Contient les fonctions



# Association avec un formulaire

## Lire/Écrire : lire-ecrire-form0.html

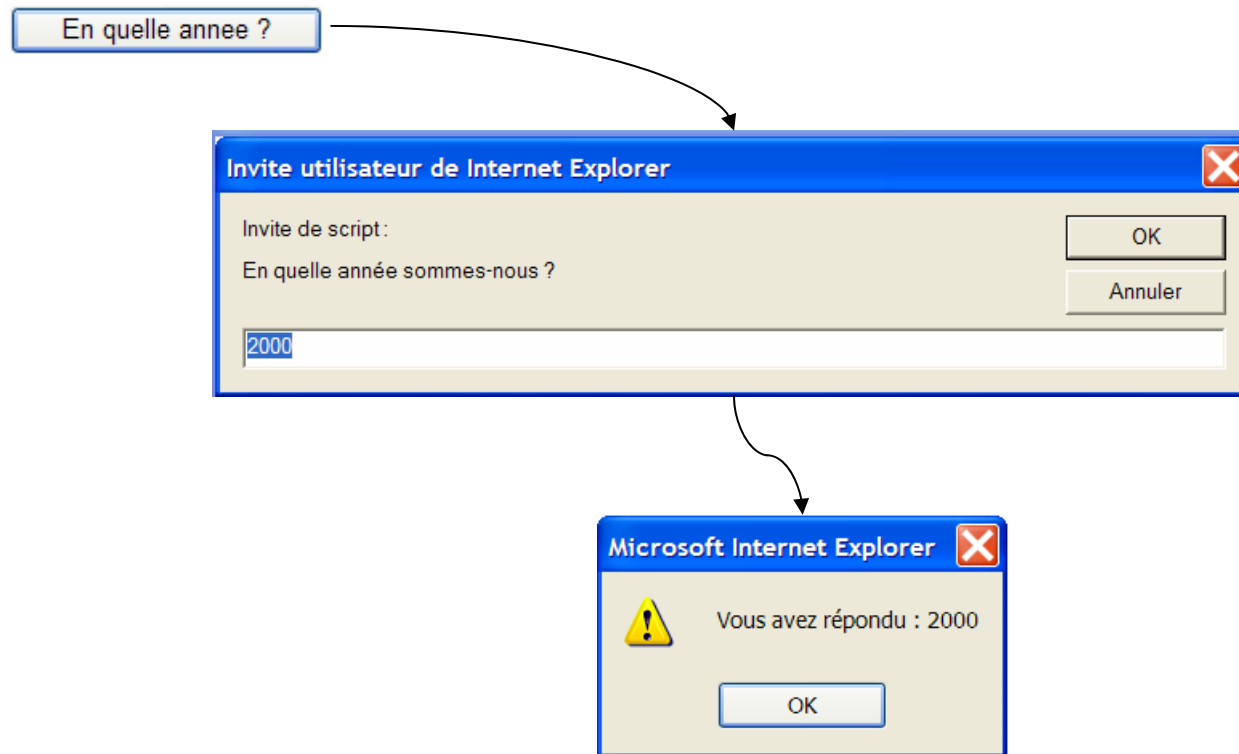
- **Exemple 1** : appel à une fonction de lecture en cliquant sur un bouton du formulaire

```
<html>
  <head>
    <title>Programme In2</title>
    <script language="JavaScript">
      function lireAnnee() {
        annee=prompt('En quelle année sommes-nous ? ', 2000);
        alert('Vous avez répondu : ' + annee) }
    </script>
  </head>
  <body>
    <form>
      <input type="button" value="En quelle année ? "
        onClick="lireAnnee()">
    </form>
  </body>
</html>
```

# Association avec un formulaire

Lire/Écrire : lire-ecrire-form0.html

## ■ Exemple 1 :





# Association avec un formulaire

- Lire un nombre et écrire son double
  - Exemple 2 : Lire-ecrire-form.html
    - Saisie du nombre :

```
<input type="text" name="nbre" size="3">
```
    - Traitement
      - ❖ `2*Number(document.lire.nbre.value);`

Entrez un nombre :   voici son double :

# Association avec un formulaire

## lire-ecrire-form.html

- Code source complet : ici le code js est dans le formulaire

```
<html>
<head><title>Programme In3</title></head>
<body>
  <form name="lire">
    Entrez un nombre :
    <input type="text" name="nbre" size="3">
    <input type="button" name="double" value="go"
    onClick="JavaScript:document.lire.aff.value=2*Num
    ber(document.lire.nbre.value);">
    voici son double : <input type="text" name="aff"
    size="8">
  </form>
</body>
</html>
```

Entrez un nombre :   voici son double :





# Association avec un formulaire

---

## ■ Exercice

- Mettez le code JavaScript dans une fonction que vous placez dans le Head

## ■ Solution

- `lire-ecrire-form-essai.html`

# Association avec un formulaire

## Lire/Écrire

### ■ Questionnaire à choix multiple

- Écrire un formulaire qui permet de réaliser un choix et d'afficher ce choix
- Exemple : lire-ecrire-form2.html

```
<head>
  <script language="JavaScript">
    var vote="";
    function enregistrer(x){vote=x}
    function depouiller()
    {document.election.resultat.value=vote}</script>
</head>
<body>
  <form name="election">
    Vous êtes pour l'informatique ?
    <input type="radio" name="vote" onClick="enregistrer('oui')">oui
    <input type="radio" name="vote" onClick="enregistrer('non')">non
    <input type="radio" name="vote" onClick="enregistrer('abstention')">
    abstention
    <input type="button" name="reponse" value="Résultat"
    onClick="depouiller()">
    <input type="text" name="resultat" value="">
  </form>
```



# Association avec un formulaire

## ■ Conditionnelle

- On cherche à contrôler la lecture des informations de type : nom, prénom et âge à l'enregistrement du formulaire de saisie
  - Le programme doit redemander l'entrée d'une information si elle est vide
- Le bouton **Annuler** servira à effacer les informations rentrées
- Exemple : conditionnelle-form.html

Nom :  Prénom :  Age :



# Association avec un formulaire

## Conditionnelle

---

### Solution

```
<body>
  <form name="fiche">
    Nom : <input type="text" name="nom" value="" size="24">
    Prénom : <input type="text" name="prenom" value=""
size="24">
    Age : <input type="text" name="age" value="" size="24">
    <input type="button" value="Enregister"
onClick="verifier()">
    <input type="reset" value="Annuler">
  </form>
</body>
```



# Association avec un formulaire

## Conditionnelle

---

```
<head><title> Si... Alors... Sinon... </title></head>
<script language="JavaScript">
function verifier()
{ if (document.fiche.nom.value=="")
  { alert('Saisissez le nom ! ');
  }
  else
  { if (document.fiche.prenom.value=="")
    { alert('Saisissez le prénom ! ');
    }
    else
    { if (document.fiche.age.value=="")
      { alert('Saisissez l'âge ! ');
      }
      else
      { alert('Vos trois champs ont été affectés');
      };};};
}
</script>
</head>
```



# Association avec un formulaire

---

- **Exercice**

- Vérifier la validité des noms par rapport à une liste de noms rangés dans un tableau

- **Solution**

- `verifier-saisie.html`

# Association avec un formulaire

## choix-form.html

- Autre exemple de lecture par formulaire HTML

- Choisir parmi une liste et afficher le choix

```
<script language="JavaScript">
function Preferer(x)
{ switch(x) {
    case 1 : alert('Votre choix : '+'classique');break;
    case 2 : alert('Votre choix : '+'jazz'); break;
    case 3 : alert('Votre choix : '+'rock'); break;
    case 4 : alert('Votre choix : '+'pop'); break; }; };
</script></head>
<body>
<form name="musique">
<select name="choix"
    onChange="Preferer(document.musique.choix.selectedIndex)">
    <option value = "choix">choisissez :
    <option value = "classique">classique
    <option value = "jazz">jazz
    <option value = "rock">rock
    <option value = "pop ">pop
</select>
</form>
```

# Association avec un formulaire

choix-form.html

- Résultat :

Entrée par formulaire : Vous préférez la musique :

classique

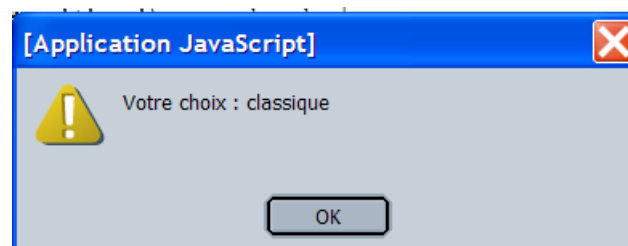
choisissez :

classique

jazz

rock

pop

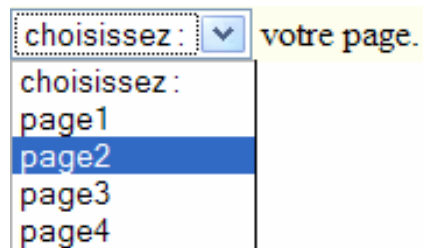




# Association avec un formulaire

## Lire/Écrire

- On voudrait créer le formulaire suivant
  - Entrée : on sélectionne une page en HTML



A screenshot of a web form. It features a label "choisissez :" followed by a dropdown menu. The dropdown menu is open, showing a list of options: "page1", "page2", "page3", and "page4". The option "page2" is currently selected and highlighted in blue. To the right of the dropdown menu, the text "votre page." is visible.

- Résultat : on l'affiche

**C'est la page 2.**



# Association avec un formulaire

## choix-form2.html, choix-form3.html

### ■ Le code

```
<html><head><script language="JavaScript">
function Envoyer(x)
{ switch(x) {
  case 1 : document.location.href='page1.htm';break;
  case 2 : document.location.href='page2.htm';break;
  case 3 : document.location.href='page3.htm';break;
  case 4 : document.location.href='page4.htm';break; };</script></head>
<body>
<form name="lecteur">
<select name="choix"
  onChange="Envoyer(document.lecteur.choix.selectedIndex)">
  <option>choisissez :
  <option>page1
  <option>page2
  <option>page3
  <option>page4
</select></form></body></html>
```

# Association avec un formulaire

## Lire/Écrire : cacher-montrer.html

- Autre exemple : cacher montrer une image
  - Créer un formulaire avec deux boutons : cacher, montrer



# Association avec un formulaire

## Lire/Écrire : cacher-montrer.html

### ■ Le code

```
<script>
  function montrer() {
    document.monImage.src='feuille.jpg'
  }
  function cacher() {
    document.monImage.src='vide.jpg'
  }
</script></head>
<body>
  <form>
    <input type="button" value="Montrer" onClick="montrer()">
    <input type="button" value="Cacher" onClick="cacher()">
  </form>
  //On initialise au début à une image vide
  
</body></html>
```

# Association avec un formulaire

## cache-monttrer2.html

- Autre variante
  - Ici, on clique sur une image pour avoir son nom inscrit



# Association avec un formulaire

## cache-montrer2.html

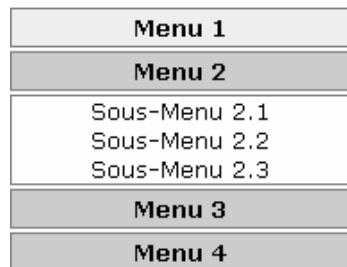
### ■ Le code

```
<html><head><script language=JavaScript>
function choisir1() {
    document.monForm.monChamp.value='Feuille'
}
function choisir2(){
document.monForm.monChamp.value='Coccinelle'
}
</script></head>
<body>
<a href="#" onClick="choisir1()"></a>
<a href="#" onClick="choisir2()"></a>
<form name="monForm">
<input type="text" name="monChamp" size="25" value="cliquez sur
    une image">
</form>
</body></html>
```

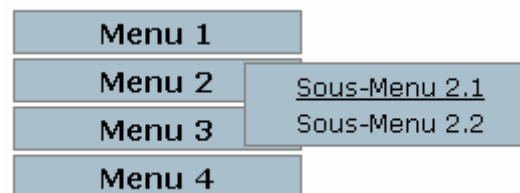
# Applications

## ■ Création de menus

- On cherche à créer des menus hiérarchiques, comme ci-dessous
- En passant sur un menu, il passe au gris clair
  - En cliquant sur Menu 2, des sous-menus apparaissent. Ils sont cliquables



[menu-vertical2.html](#)



variante : [menu-vertical2.html](#)



# Applications

---

- Pour lier les menus et les sous-menus : on utilise les listes de définition `<DL>`
  - Les listes de définitions comportent deux parties,
    - un **terme** et une **description**
  - Pour marquer une liste de définitions, il vous faut trois éléments HTML :
    - un `<dl>` conteneur,
    - un terme de définition `<dt>`,
    - et une description de définition `<dd>`
  - Par exemple :

```
<dl>  
  <dt>Grenouille</dt>  
  <dd>Truc vert humide</dd>  
  <dt>Lapin</dt>  
  <dd>Truc chaud et doux</dd>  
</dl>
```



```
<dl id="menu">
  <dt onclick="javascript:montre();"><a href="#">Menu1</a></dt>
  <dt onclick="javascript:montre('smenu2');">Menu2</dt>
    <dd id="smenu2">
      <ul>
        <li><a href="#">Sous-Menu 2.1</a></li>
        <li><a href="#">Sous-Menu 2.2</a></li>
        <li><a href="#">Sous-Menu 2.3</a></li>
      </ul>
    </dd>
  <dt onclick="javascript:montre('smenu3');">Menu3</dt>
    <dd id="smenu3">
      <ul>
        <li><a href="#">Sous-Menu 3.1</a></li>
        <li><a href="#">Sous-Menu 3.1</a></li>
        <li><a href="#">Sous-Menu 3.1</a></li>
        <li><a href="#">Sous-Menu 3.1</a></li>
        <li><a href="#">Sous-Menu 3.1</a></li>
        <li><a href="#">Sous-Menu 3.1</a></li>
      </ul>
    </dd>
  <dt onclick="javascript:montre('smenu4');">Menu4</dt>
    <dd id="smenu4">
      <ul>
        <li><a href="#">Sous-Menu 4.1</a></li>
        <li><a href="#">Sous-Menu 4.1</a></li>
      </ul>
    </dd>
</dl>
</body>
```

```
<script type="text/javascript">
<!--
window.onload=montre; //montre le menu principal au chargement
function montre(id) {
var d = document.getElementById(id);
    //On cache les sous-menus
    for (var i = 1; i<=10; i++) {
        if (document.getElementById('smenu'+i))
            {document.getElementById('smenu'+i).
                style.display='none';}
    }
    //On affiche le sous-menu
    if (d) {d.style.display='block';}
}
//-->
</script>
```

# Association avec un formulaire

- **Exercice : réaliser le formulaire suivant**
  - En cliquant sur le bouton Pays, on affiche une liste de noms de pays, en cliquant sur Capitales, on affiche une liste de capitales
- **Solution :**
  - pays\_capitales.html

Résultat :



A screenshot of a web form. On the left, there is a list box containing the names of four capital cities: Berlin, Vienne, Bruxelles, and Copenhague. To the right of the list box are two buttons: 'Pays' and 'Capitales'. The list box has a vertical scrollbar on its right side.

Déclaration :

```
E=new Array(2); // Pays Capitales
E[0]=new Array(15); // 15 Pays
E[1]=new Array(15); // 15 Capitales
```

Affectation :

```
E[0][0]='Allemagne';E[0][1]='Autriche';E[0][2]='Belgique'; E[0][3]='Dane...
E[1][0]='Berlin'; E[1][1]='Vienne'; E[1][2]='Bruxelles';E[1][3]='Cope...
```



# Association avec un formulaire

- **Exercice : réaliser le formulaire suivant**
  - Il s'agit de calculer la moyenne d'un élève ou d'une matière
- **Solution**
  - Moyennes\_notes.html

## Les données :

	Français	Math	Anglais	Histoire
Aurélie	12	11,5	13	12
Bastien	8,5	12	10	9,5
Naïma	10,5	10	14	8,5

## Les moyennes :

Bastien	▼	10	Français	▼	10.3333
---------	---	----	----------	---	---------