ESI
مدرسة علوم المعلومات
ⵜⵉⵎⵔⵔⴰⵜ ⵏ ⵜⵓⵙⵙⵏⵉⵡⵉⵏ ⵏ ⵜⵣⵔⴼⵜ
ECOLE DES SCIENCES DE L'INFORMATION

Technologies Web

Pr. Fatima MOURCHID

DOM (Document Object Model)

Introduction

- Modèle Objet de Document: un outil permettant l'accès aux documents HTML et XML
  - interface de programmation normalisée par le W3C, qui permet à des scripts d'examiner et de modifier le contenu d'un document balisé, notamment une page Web
  - fournit une représentation structurée du document
  - codifie la manière dont un script peut accéder à cette structure

  =>lier une page Web à un langage de programmation ou de script
  - Bibliothèques de scripts comme jQuery, qui offrent des solutions « clé en main » pour lé développement d'interfaces évoluées
  - Ecrire des scripts en JavaScript, qui utiliseront le DOM pour manipuler dynamiquement les pages Web où ils se trouvent

Introduction

- Spécifications W3C DOM sont organisés en "niveaux"
- Level 0
  - Implémentations variés non standardisés, comme le modèle d'objets DHTML de Microsoft, le DOM de Netscape 4.x, etc.
- Level 1 (octobre 1998)
  - DOM Level 1 Core: définit la navigation et la manipulation de documents HTML et XML
  - DOM Level 1 HTML: extensions spécifiques à HTML
- Level 2 (2001)
  - 6 spécifications, ajoute XML namespace support, filtered views and events.
    - DOM Level 2 Core Specification (étend DOM Level 1 Core)
    - DOM Level 2 Views Specification
    - DOM Level 2 Events Specification: gestion d'événements (mais pas du clavier)
    - DOM Level 2 Style Specification: lecture et modification du CSS
    - DOM Level 2 Traversal and Range Specification
    - DOM Level 2 HTML Specification: extensions pour HTML

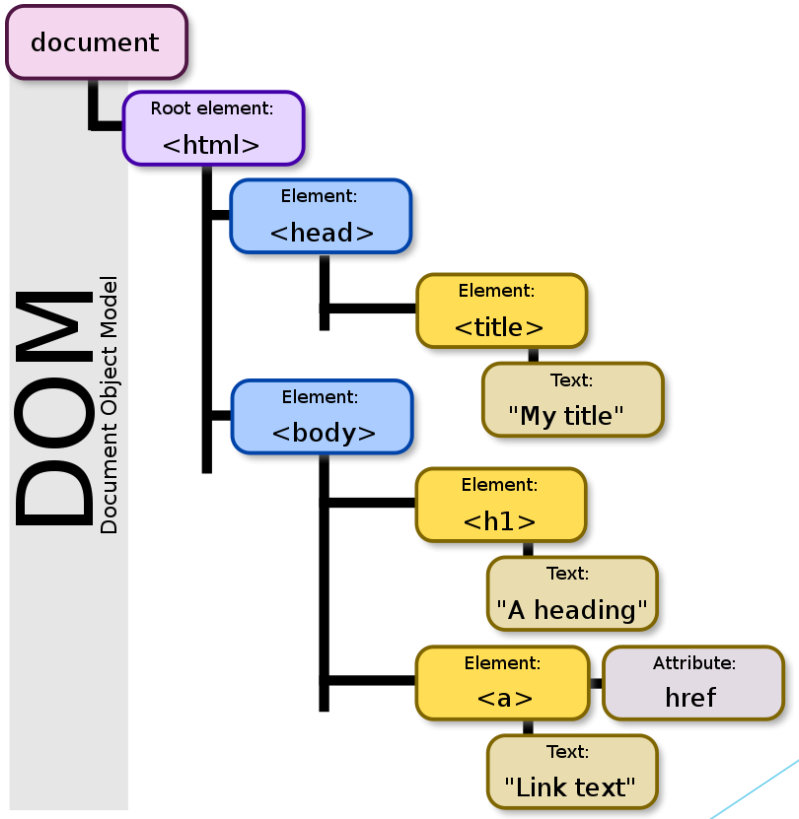
Introduction

- Level 3: (avril 2004)
  - contient 6 spécifications
    - DOM Level 3 Core
    - DOM Level 3 Load and Save
    - DOM Level 3 XPath
    - DOM Level 3 Views and Formatting
    - DOM Level 3 Requirements
    - DOM Level 3 Validation
- Level 4: (décembre 2015)
  - Dernière mise à jour date de décembre 2020

Conception de l'arborescence

- Un langage de marquage comme HTML peut être schématisé comme une arborescence hiérarchisée

- Types de nœuds dans un document HTML: nœuds-élément, nœuds-attribut, nœuds-texte, …



https://fr.wikipedia.org/wiki/Document_Object_Model

Conception de l’arborescence

Tag	Description
<!DOCTYPE>	Defines the document type
<html>	Defines an HTML document
<head>	Contains metadata/information for the document
<title>	Defines a title for the document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings
<p>	Defines a paragraph
 	Inserts a single line break
<hr>	Defines a thematic change in the content
<!--...-->	Defines a comment

Conception de l’arborescence

Attribute	Belongs to	Description
accept	<input>	Specifies the types of files that the server accepts (only for type="file")
accept-charset	<form>	Specifies the character encodings that are to be used for the form submission
accesskey	Global Attributes	Specifies a shortcut key to activate/focus an element
action	<form>	Specifies where to send the form-data when a form is submitted
alt	<area>, <img>, <input>	Specifies an alternate text when the original element fails to display
id	Global Attributes	Specifies a unique id for an element

Conception de l’arborescence

- Informations sur la nature d'un nœud déterminé

Méthode	Rôle
nodeName	renvoie le nom du nœud courant
nodeType	renvoie au type de nœud du nœud courant

Type de nœud	Valeur retournée par nodeType
element_node	1
attribute_node	2
text_node	3
…	…

Accès aux éléments

- Accès direct
  - à partir de l'ensemble du document

Méthode	Rôle
getElementById	sélectionner un élément d'identifiant donné dans une page
getElementsByName	sélectionner les éléments portant un nom donné dans une page
getElementsByTagName	sélectionner les éléments portant un nom de balise donné dans une page

Accès aux éléments

- à partir d'un élément quelconque d'un document

Méthode	Rôle
getElementsByTagName()	sélectionner les éléments portant un nom de balise donné dans une page
getAttribute(nom_d_attribut)	sélectionner un attribut particulier d'un élément déterminé

```
parag1=document.getElementById('monparagraphe') ;
liens=parag1.getElementsByTagName('a') ;
```

Accès aux éléments

- Accès relatif
  - Accès direct nécessite une connaissance soit de l'identifiant, soit du nom exact du nœud cherché

=>accéder à des collections de nœuds dont on ne connaît pas ces caractéristiques a priori

<img src="oiseau.gif" alt="Un oiseau sur un arbre" width="45" height="60">

elt=parag1.getElementsByTagName('img') ;

elt.attibutes() ;// affiche la collection des attributs de l'élément
// img: src, alt, width et height

Accès aux éléments

- Accès par collections
  - window.frames: permet d'accéder à la liste des frames du document courant
  - document.forms, document.images et document.links: collections des éléments de formulaire, d'images et de liens du document

Ajout de contenu dans un document

Méthode	Rôle
createElement	créer un élément
createTextNode	créer un nœud de type texte
cloneNode(deep)	créer une copie de l'élément courant
appendChild()	affecter un nœud à un nœud déjà existant

Ajout de contenu dans un document

```
liste=document.getElementById("maliste");
nouveauli=document.createElement("li");
nouveauli.appendChild(document.createTextNode("suite de la liste"));
liste.appendChild(nouveauli);
```

Modification du contenu dans un document

- Modification d'attributs

```
elt= getElementById('maliste');
elt.style.fontSize='1.4em'; //taille de la police de l'élément
                            //sélectionné à 140% de sa taille par
                            //défaut
elt.setAttribute("align", "center") // modifier l'attribut align de
                                    //l'élément courant en centré
```

- Modification d'éléments

```
elt.replaceChild(newChild, oldChild)
```

Suppression du contenu dans un document

Méthode	Rôle
removeChild	Suppression d'un nœud quelconque
removeAttribute	Suppression d'un attribut
removeAttributeNode	

```
elt=document.getElementById("div1") ;
elt_inclus=document.getElementById("para1") ;
elt_rejete=elt.removeChild(elt_inclus) ;   // supprime le nœud para1
```

```
elt=document.getElementById(« para1") ;
attr_align=elt.getAttributeNode("align") ;
elt.removeAttributeNode(attr_align) ; // supprime l'attribut align du nœud « para1 »
```

Gestion des évènements et DOM

- Comment appeler un événement?
  - Par un attribut HTML
  - Par l'utilisation d'une propriété d'un élément
  - Par des « event listeners »: utiliser une propriété pour affecter un événement à un élément
    - noeud.addEventListener(eventType, fonction, useCapture)
    - noeud.removeEventListener(eventType, fonction, useCapture)

Gestion des évènements et DOM

```
<body onload="VerifierCookies()">
…
<script>
function VerifierCookies() {
        var text = "";
        if (navigator.cookieEnabled == true) {    text = "Cookies sont activées.";   }
        else {     text = "Cookies sont désactivées.";   }
        document.getElementById("demo").innerHTML  = text;
}
</script>
```

Gestion des évènements et DOM

```
<span id="exemple1">Exemple avec un événement souris</span>
…
<script>
    document.getElementById('exemple1').onmouseover = miseEnGras ;
    document.getElementById('exemple1').onmouseout = normal ;
    function miseEnGras(event){
        this.style.fontWeight="bold" ;
        this.style.color="red" ; }
    function normal(event){
        this.style.fontWeight="normal" ;
        this.style.color="" ;}
</script>
```

Gestion des évènements et DOM

```
var el=.document. getElementById("exemple1") ;
el.addEventListener("mouseover", miseEnGras, false) ;
el.addEventListener("mouseout", normal, false) ;
```

Gestion des évènements et DOM

- Evènements page et fenêtre

Evènement	Rôle
onabort	s'il y a une interruption de chargement
onerror	en cas d'erreur pendant le chargement de la page
onload	après la fin du chargement de la page
onbeforeunload	se produit juste avant de décharger la page en cours (par changement de page, en quittant)
Onunload	se produit lors du déchargement de la page (par changement de page, en quittant)
onresize	quand la fenêtre est redimensionnée

Gestion des évènements et DOM

- Événements souris

Evènement	Rôle
onclick	sur un simple clic
ondblclick	sur un double clic
onscroll	lorsque le scroll de la souris est utilisé

Gestion des évènements et DOM

- Événements clavier

Evènement	Rôle
onkeydown	lorsqu'une touche est enfoncée
onkeypress	lorsqu'une touche est pressée et relâchée
onkeyup	lorsqu'une touche est relâchée

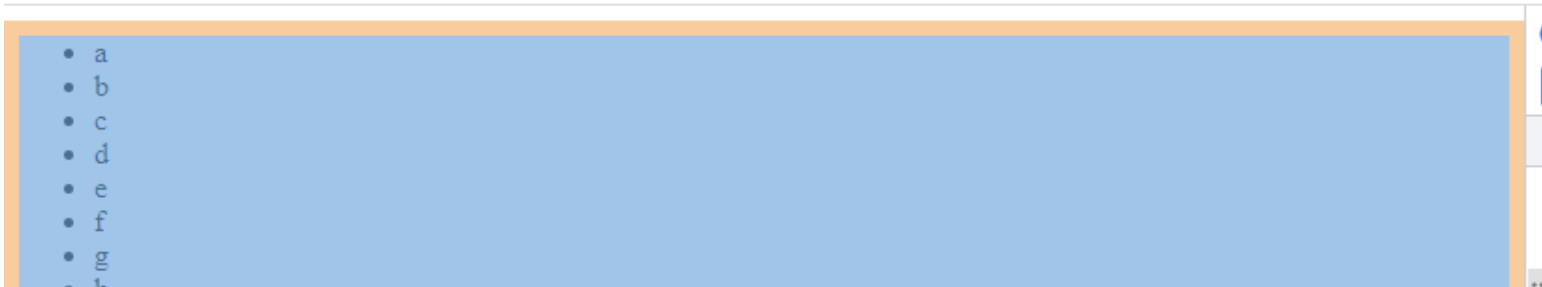
Gestion des évènements et DOM

- Événements formulaire

Evènement	Rôle
onfocus	lorsque l'élément obtient le focus (ou devient actif)
onselect	quand du texte est sélectionné
onsubmit	quand le formulaire est validé via un bouton ou une fonction submit()

TPs: enoncés

- TP6: Écrire les fonctions permettant d'avancer et de reculer dans l'historique du navigateur
- TP7:
  - Afficher une liste d’éléments
  - Récupérer l'objet de liste à l'aide d'une méthode getElementsByTagName
  - Récupérer la collection d'items de liste à l'aide d'une méthode getElementsByTagName
  - Changer la valeur de l'attribut de type de la liste en "circle"
  - Pour chaque élément de liste, écrire le texte en majuscule, et affecter à l'attribut onclick un appel à une boîte d'alerte avec un « Ceci est un exercice sur DOM»



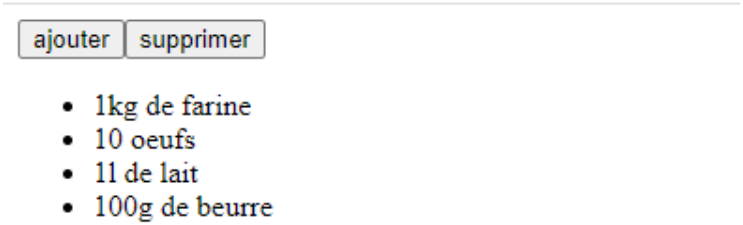
TPs: enoncés

- TP8:
  - Créer une liste d'identifiant "listeProvisions"
  - Créer un nouvel item de liste et lui affecter un texte: «suite de la liste »
  - Ajouter ce nouvel élément à la liste

- 1kg de farine
- 10 oeufs
- 1l de lait
- 100g de beurre

TPs: enoncés

- TP9:
  - Reprendre le TP8
  - Ajouter un bouton permettant, par un appel de fonction, de supprimer le dernier élément de la liste



Références

- JavaScript: https://developer.mozilla.org/en-US/docs/Web/JavaScript
- Le Modèle Objet de Document (DOM: Document object Model): https://developer.mozilla.org/fr/docs/Web/API/Document_Object_Model/Introduction
- Document Object Model (DOM) Specification: http://www.w3.org/TR/dom/
- Le site officiel de jQuery: https://jquery.com/
- Introduction à jQuery: https://openclassrooms.com/fr/courses/3504441-introduction-a-jquery
- Le site officiel de Angular: https://angularjs.org/
- Introduction to Web Development: https://www.coursera.org/learn/web-development
- Introduction to Web Development with HTML, CSS, JavaScript: https://www.coursera.org/learn/introduction-to-web-development-with-html-css-javacript
- HTML Tutorial: https://www.w3schools.com/html/default.asp
- JavaScript Tutorial: https://www.w3schools.com/js/default.asp
- JSON Tutorial : https://www.w3schools.com/js/js_json_intro.asp