



Support de cours 2

JavaScript

FRÉDÉRIC NADARADJANE

LA GRANDE CLASSE

31/03/2023

7.L'objet Window

L'objet Window

Présentation

L'objet de la fenêtre est pris en charge par tous les navigateurs. Il représente la fenêtre du navigateur.

- Tous les objets JavaScript globaux deviennent automatiquement membres de l'objet fenêtre.
 - ❖ Les variables globales sont des propriétés de l'objet fenêtre.
 - ❖ Les fonctions globales sont des méthodes de l'objet fenêtre.
- Même l'objet document (du DOM HTML) est une propriété de l'objet fenêtre.

L'objet Window

Dimensions

Deux propriétés peuvent être utilisés pour déterminer la taille de la fenêtre du navigateur.

Les deux propriétés renvoient les tailles en pixels

Les deux propriétés renvoient les tailles en pixels :

- `Window.innerHeight` : la hauteur interne de la fenêtre du navigateur (en pixels)
- `Window.innerWidth` : la largeur interne de la fenêtre du navigateur (en pixels)

L'objet Window

Méthodes

Quelques autres méthodes :

- `window.open()` – ouvre une nouvelle fenêtre

`myWindow = window.open("", "", "width=100, height=100")`

- `window.close()` – ferme la fenêtre actuelle

- `window.moveTo()` – mouvement de la fenêtre actuelle

- `window.resizeTo()` – resize de la fenetre en cours

L'objet Window History

L'objet `window.history` contient l'historique des navigateurs.

Pour protéger la confidentialité des utilisateurs, il existe des limites à la façon dont JavaScript peut accéder à cet objet.

Quelques méthodes :

- `history.back ()` – identique la page consulter précédant la page actuelle
- `history.forward ()` – identique la page consulter suivante la page actuelle

L'objet Window

Boîte contextuelles

Les boîtes contextuelles que nous avons vues précédemment sont des méthodes de l'objet window :

- `window.alert()`
- `window.prompt()`
- `window.confirm()`

L'objet Window

Exercice

Ouvrez une window de taille 200x200. on demande à l'utilisateur la taille du resize souhaité et on l'applique. Bougez là ensuite à 600,300.

L'objet Window

Exercice

Ouvrez une window de taille 100x600. on resize à 400x200. Bougez là ensuite à 200,600.

8. Les évènements

Les évènements

Présentation

Le JavaScript peut être exécuté lorsqu'un événement se produit.

Exemples d'évènements :

- Lorsqu'un utilisateur clique sur la souris
- Lorsqu'une page web a été chargée
- Lorsqu'une image a été chargée
- Lorsque la souris se déplace sur un élément
- Lorsqu'un champ d'entrée est modifié
- Lorsqu'un formulaire HTML est soumis
- Lorsqu'un utilisateur appuie sur une touche

Les évènements

Intégration – inline

Pour exécuter du code lorsqu'un utilisateur clique sur un élément, ajoutez du code JavaScript à un attribut d'événement HTML:

onclick = JavaScript

Exemple :

```
<h1 onclick = "this.innerHTML = 'Oooops'">Click on this text!</h1>
```

Les évènements

Intégration – inline

Une deuxième possibilité est d'appeler directement une fonction JavaScript pour effectuer une action, ou une série d'actions :

onclick= fonctionJavaScript

Exemple :

HTML :

```
<h1 onclick="changeText(event)"> Click on this text !</h1>
```

JS :

```
function changeText(event){  
    event.target.innerHTML = "Oooooops!"  
}
```

Les évènements

Intégration – externe

Enfin, une troisième possibilité existe permettant d'attacher un évènement à une balise HTML, offrant la possibilité de séparer complètement le HTML du JavaScript :

#objet récupéré#.onclick = changeText()

Exemple :

HTML :

```
<h1 id= 'titlePrimary'>Click on this text </h1>
```

JS :

```
document.getElementById("titlePrimary").onclick = changeText;
```

```
function changeText(event){  
    event.target.innerHTML="Oooops!";}
```

Les évènements

List des évènements

Liste des évènements communs :

Propriété	Description
onchange	Un élément HTML a été modifié
onclick	L'utilisateur clique sur un élément HTML
onmouseover	L'utilisateur déplace la souris sur un élément HTML
onmouseout	L'utilisateur déplace la souris hors d'un élément HTML
onkeydown	L'utilisateur appuie sur une touche du clavier
onload	Le navigateur a fini de charger la page

Liste des événements : <https://developer.mozilla.org/fr/docs/Web/Events>

Les évènements

Exemple

Événement lié à l'appui et le relâchement de la souris

Exemple :

HMTL : `<div onmousedown="mDown(event)" onmouseup="mUp(event)"
style="background-color:green;width:90px;height:20px;padding:40px;">
Click Me</div>`

JS:

```
function mDown(event) {  
    var elem = event.target;  
    elem.style.backgroundColor = "#1ec5e5"; elem.innerHTML = "Click down";  
}  
function mUp(event) {  
    var elem = event.target;  
    elem.style.backgroundColor="#D94A38"; elem.innerHTML="Click up";  
}
```


Les évènements

Exemple

Événement lié au survol de la souris

Exemple :

HMTL : `<div onmouseover="mOver(this)" onmouseout="mOut(this)"
style="background-color:#D94A38;width:120px;height:20px;padding:40px;">
Mouse Over Me</div>`

JS:

```
function mOver(obj) {  
    obj.innerHTML = "Thank You";  
}  
  
function mOut(obj) {  
    obj.innerHTML = "Mouse Over Me";  
}
```

Les évènements

Exemple

Événement lié au chargement

Exemple :

HTML :

```
<body onload = "mymessage()"> </body>
```

JS:

```
function mymessage(){  
    alert("This message was triggered from the onload event");  
}
```

Les évènements

Exercice

Dans une page html, mettez un titre h1 : Bonjour

Ensuite dans le javascript, demander le nom de l'utilisateur, via un prompt.

Lors du clique sur le titre « bonjour », le texte doit afficher à la place de celui-ci un titre en vert :

"Merci d'avoir cliqué [Nom de l'utilisateur]".

Les évènements

Exercice

Afficher une div avec ces propriétés :

- Un background orange
- Dimension : 400 x 400
- Des bords arrondi de 5px

On demandera le nom d'utilisateur via un prompt et lors du survole, mettez une alert "Merci [#PRENOM#], d'avoir survolé le carré".

Les évènements

Exercice

Créer un p avec un lorem,

Au clique sur ce texte on demandera à l'utilisateur son âge, s'il à plus de 60 ans la taille du texte sera de 70px, sinon si l'utilisateur à plus de 12 ans le texte aura pour taille 24px, et le texte aura 12px dans les autre cas.

Les évènements

Exercice

Compteur de clique :

Faire une div qui sera de couleur rouge et aura pour dimension :

600x600

A chaque clique sur la div un compteur sera incrémenté et on affichera ce compteur en dessous de la div

Les évènements

Ecouteur d'évènements - Présentation

L'écouteur d'évènements permet de gérer les différents événements attachés à un objet HTML

- La méthode `addEventListener()` attache un gestionnaire d'événements à l'élément spécifié.
- La méthode `addEventListener()` n'écrase pas les gestionnaires d'événements existants
- Vous pouvez ajouter de nombreux gestionnaires d'événements à un seul élément.
- Vous pouvez ajouter plusieurs gestionnaires d'événements du même type à un élément, c'est-à-dire deux événements "clic".

Les évènements

Ecouteur d'évènements - Avantages

Les avantages d'un écouteur d'évènements :

- La méthode `addEventListener()` facilite le contrôle de la réaction de l'évènement
- Vous pouvez facilement supprimer un écouteur d'évènements en utilisant la méthode `removeEventListener()`

Les évènements

Ecouteur d'événements – Ajout d'événements

Syntaxe pour réagir au clic :

```
#element récupéré#.addEventListener("click", function(){ alert("Hello world");});
```

```
#element récupéré#.addEventListener("click", myFunction);
```

Ajouter des événements de différents types au même élément :

```
#element récupéré#.addEventListener("mouseover", myFunction);
```

```
#element récupéré#.addEventListener("click", mySecondFunction);
```

```
#element récupéré#.addEventListener("mouseout", myThirdFunction);
```

Les évènements

Ecouteur d'événements – Arguments

Passer des arguments à un écouteurs d'événements :

Exemple :

```
document.getElementById("myBtn").addEventListener("click",function(){  
    myFunction(p1, p2);  
});
```

Les évènements

Ecouteur d'évènements – Supprimer

Supprimer un écouteur d'évènement :

```
#élément récupéré#.removeEventListener("mousemove", myFunction);
```

Exemple :

```
Document.getElementById("myDiv").removeEventListener("mousemove", myFunction);
```

Les évènements

Les transitions

On peut également ajouter l'attribut transition vue en css

JS :

```
function changeText(event){  
    event.target.innerHTML = "oops";  
    event.target.style.color = 'blue';  
    event.target.style.transition = "all 1s ease-out";  
    document.getElementById('lorem').style.color = 'red';  
}
```

```
document.getElementById("text").onclick = changeText;
```

Les évènements

Les transitions

On peut également déplacer des éléments html en fonction du clique

HTML :

```
<p>Click anywhere to move the ball</p>
```

```
<div id="ball"></div>
```

Les évènements

Les transitions

JS

```
var b = document.getElementById("ball");

document.addEventListener('click',
function(ev){

    b.style.transform = 'translateY(' + (ev.clientY-
25) + 'px)' + 'translateX(' + (ev.clientX-25) + 'px)';

    b.style.transition = "transform 1s";

});
```

CSS

```
p{padding-left:60px;}

#foo{

border-radius:50px;

width:50px;

height:50px;

background:#c00;

position:absolute;

top:0;

left:0;

}
```

Les évènements

Exemple

Créons un h1 avec le texte : « survolez-moi », qui lors du survole de celui-ci change pour devenir Merci d'avoir survoler.

Un bouton permettra d'arrêter l'évènement.

Les évènements

Exercice

Crée un lorem dans un p au click il changera en h1, le changement se fera avec une transition d'1s en ease-out.

Le titre final sera en bleu.