



TP1 PWCR

29.01.2021

JavaScript, images et événements souris

AHAROUITE Ahmed

Groupe 1A

DUT Informatique



Table des matières :

Introduction	1
Objectifs	1
Exercice 1 : rollover sur les images	2
Observation de la balise input	2
Menu dynamiques	3
Exercice 2 : Carrousel	4
Rotation d'image en continue	4
Conclusion	5

Introduction

Ce TP portera sur le langage JavaScript. Nous allons apprendre à créer des événements liés à des images.

Objectifs

1. Savoir créer / utiliser des événements js.
2. Lier / appeler des événements js sur des images.

Exercice 1 : rollover sur les images

I. Observation de la balise input

Cette balise appelle la fonction clicclac avec comme paramètre cette dernière, sur l'événement onclick du bouton (clic du bouton).

```
<input type="button" value="cliquez ici pour voir" onclick="clicclac(this)"/>
```

```
function clicclac(unObjet) {  
    // Test de la valeur de l'objet  
    if (unObjet.value !== undefined)  
    {  
        if (unObjet.value === 'clac')  
        {  
            unObjet.value = 'clic'; // Changement de la valeur de l'objet  
        } else {  
            unObjet.value = 'clac'; // Autre changement ...  
        }  
    }  
}
```

Résultat:

Par défaut	Après un clic	Après un autre clic
<input type="button" value="cliquez ici pour voir"/>	<input type="button" value="clac"/>	<input type="button" value="clic"/>

La valeur du bouton après le premier clic suit le cycle: clic, clac, clic, clac ...

II. Menu dynamiques

Pour définir des menus "dynamiques", on souhaite utiliser 3 images :

- nomImage.gif : pour le bouton au repos
- nomImage_over.gif : quand la souris survole l'image
- nomImage_click.gif : quand l'image est cliquée

Grâce à cette règle de nommage, on souhaite réaliser la mise à jour des images à l'aide d'une méthode :

```
function imageChange(objetImage, nomImage){
    //L'attribut src prend une nouvelle source appartenant à une image
    objetImage.src = nomImage;
}
```

Mise à jour de l'image email.gif :

```
onmouseover="imageChange(this, 'email_over.gif')"
```

```
onmouseout="imageChange(this, 'email.gif')"
```

```
onclick="imageChange(this, 'email_click.gif')"
```

La mise à jour de l'image home.gif fonctionne de la même façon que l'image email.gif. Les deux images appellent donc la méthode imageChange avec comme premier paramètre ces dernières, et pour le deuxième la source d'une image.

Cette fonction est appelée sur trois événements, voici les résultats :

→ onmouseover : quand le bouton est au repos



→ onmouseout : quand la souris survole l'image



→ onclick (Bonus) : quand l'image est cliquée



Exercice 2 : Carrousel

III. Rotation d'image en continue

Dans cette même page HTML, on souhaite maintenant faire défiler toutes les deux secondes différentes images de papillons (fournies).

Pour cela, j'ai pu:

- définir un tableau contenant les différentes images.

```
let tabSrc = ["papillon1.jpg", "papillon2.jpg", "papillon3.jpg", "papillon4.jpg"];
```

- ajouter un « id » à l'image à traiter.

```

```

- définir une fonction qui changera le contenu de l'image associée à un id spécifique toutes les 2 secondes (grâce à la fonction sleep).

```
function sleep(ms) {
  return new Promise(
    resolve => setTimeout(resolve, ms)
  );
}

async function imageChange2(){
  //Récupération de l'image par son id
  let objetImage = document.getElementById('idImage');

  //Boucle infinie
  while(true){
    for(let i=0; i<tabSrc.length; i++){
      imageChange(objetImage, tabSrc[i]); //Changement de source
      await sleep(2000); // 2 secondes de pause
    }
  }
}
```

- lancer une première fois cette fonction (immédiatement).

```
<body onload="imageChange2()">
```

L'image change bien de source/contenu toutes les deux secondes.

Bonus css : Oui, il est possible de donner un effet de fondu à votre carrousel grâce à des directives css. En css, les événements connus en js peuvent être remplacés par des animations (propriété css). Leurs durées peuvent être initialisées et donc l'opacité de l'image peut être modifiée dans le css et dans ces animations.

Conclusion

Ce TP01 a pour but de mettre en application ce que nous avons déjà vu au cours des TD et cours précédents, c'est-à-dire la manipulation du langage JavaScript sur une page HTML.

Durant ce TP, on a pu observer des événements, définir des menus dynamiques et créer des méthodes qui sont liées à des événements et des éléments de la page HTML.

Tout cela, nous a donc permis d'approfondir nos connaissances sur les événements et leur exploitation grâce au langage JavaScript.

