

## SEANCE 3 : gestion événementielle

**Exercice 1** : Indiquer pour chaque séquence, la cible et l'événement pris en compte.

```
<!DOCTYPE html>
<html>
<head>
  <title>Exemple Eévénements</title>
</head>
<body>
  <h1 class="titre">Titre</h1>
  <p id="p1">Debutfin</p>
  <script>
  </script>
</body>
</html>
```

```
document.addEventListener('keypress', action1);
function action1(evt) {window.alert(evt.charCode);}
```

le type	
la cible/target	
l'action associée (sous forme de fonction)	

```
var target;
target=document.querySelector('h1');
target.addEventListener('mouseover', action2);
function action2(evt) {window.alert("survol d'un titre");}
```

le type	
la cible/target	
l'action associée (sous forme de fonction)	

```
var position=document.getElementById('p1');
position.addEventListener('click', action3);
function action3(evt) {window.alert(evt.clientX + " : " + evt.clientY);}
```

le type	
la cible/target	
l'action associée (sous forme de fonction)	

✓ Tester ces différents exemples.

**Exercice 2 :** Positionner un écouteur d'événement sur l'objet window pour que s'affiche le message « document chargé » dans la console lorsque le document est chargé.

D'après vous, quand doit-on s'assurer que le document est chargé ?

**Exercice 3 :**

Écouter les saisies clavier sur le body et afficher un message quand on tape la lettre 'a'.

**Exercice 4** - des cartes

Récupérer le fichier exo4.html

Événements liés aux interactions avec le clavier

Ajoutez le code Javascript pour que, lorsque l'utilisateur saisit un caractère, ce caractère soit ajouté après le texte Date

Événements liés aux interactions avec la souris

Ajoutez un écouteur d'événements pour que lorsque l'utilisateur clique sur une des deux cartes (balises d'identifiant "c1" ou "c2"), elle change de couleur (pour le changement aléatoire de couleur reprendre la fonction du TP n°1).

**Exercice 5** – déplacement au clavier

Récupérer le fichier exo5.html.

- a) Le but de l'exercice est de déplacer la carte en utilisant les touches de direction du clavier flèche bas, flèche droite, flèche haut, flèche gauche.

Attention le type d'événement keypress ne fonctionne pas pour toutes les touches du clavier ; chercher dans la documentation en ligne les autres événements liés au clavier.

Tester en affichant dans la console le code de la touche.

Les propriétés offsetLeft et offsetTop permettent de récupérer les positions actuelles absolues de l'objet (en nombre de pixels).

- b) Ajouter la gestion des bords pour éviter que la carte disparaisse.

Les propriétés innerHeight et innerWidth de l'objet window permettent de récupérer la largeur et la hauteur de la fenêtre.

**Exercice 6** – déplacement à la souris

Le fichier de départ est le même que pour l'exercice 5.

- a) Le but de l'exercice est de déplacer la carte à chaque clic de la souris, vers l'endroit cliqué.
- b) Ajouter un effet de zoom (+10px sur la largeur et la hauteur) sur la carte si le clic a lieu sur la carte
- c) Si on clique sur la carte elle ne doit pas se déplacer.