

Exercices Javascript DOM + corrections

Exercice 1 :

Manipuler les styles des éléments html en javascript.

Modèle HTML :

```
<!doctype html>

<html>

<head>

    <title>styles</title>

    <meta charset="utf-8">

</head>

<body>

    <p id="parag1">Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
    tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
    quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
    consequat.

    </p>

    <button onclick="changer_style()">Changer Style</button>

</body>
```

Question 1 :

Ecrire la fonction changer_style qui permet de styler le paragraphe au clic du bouton par :

une couleur blanche.

un background noir.

une bordure noire pointillée de 1px.

un retrait de 5px.

Exemple :

```
function changerStyle(){
    var x = document.getElementById('parag1');x
    x.style.color="white";
    ...
}
```

Correction :

```
function changerStyle(){
    var p = document.getElementById('parag1');
    p.style.color="white";
    p.style.backgroundColor="#bbb";
    p.style.border="1px solid gray";
    p.style.padding="5px";
}
```

Question 2 :

Définir les propriétés précédentes dans une classe "active" et modifier la fonction changer_style de telle façon qu'elle ajoute la classe "active" au paragraphe.

Correction :

- On commence par la définition de la classe "**active**"

```
.active{
    color: #fff;
    background: #000;
    border: 1px solid black;
    padding: 5px;
}
```

- Après on ajoute la classe au paragraphe via la propriété classList, donc la fonction **changer_style** devient:

```
function changerStyle(){
    var p = document.getElementById('parag1');
    p.classList.add("active");
}
```

Exercice 2 :

Écrivez une fonction JavaScript pour obtenir la valeur des attributs href, hreflang, rel, target et type du lien spécifié.

Modèle HTML :

```
<!DOCTYPE html>

<html>

<head>

<meta charset=utf-8 />

</head>

<body>

<p><a id="w3r" type="text/html" hreflang="en-us" rel="nofollow" target="_self"
href="https://www.w3resource.com/">w3resource</a></p>

<button onclick="getAttributes()">Click here to get attributes value</button>

</body>

</html>
```

Correction :

```
function getAttributes()
{
    var u = document.getElementById("w3r").href;
    alert('The value of the href attribute of the link is : '+ u);
    var v = document.getElementById("w3r").hreflang;
```

```

alert('The value of the hreflang attribute of the link is : '+ v);

var w = document.getElementById("w3r").rel;

alert('The value of the rel attribute of the link is : '+ w);

var x = document.getElementById("w3r").target;

alert('The value of the taget attribute of the link is : '+ x);

var y = document.getElementById("w3r").type;

alert('The value of the type attribute of the link is : '+ y);

}

```

Exercise 3 :

Écrivez une fonction JavaScript pour obtenir les valeurs du prénom et du nom de famille du formulaire suivant.

Modèle HTML :

```

<!DOCTYPE html>

<html>

<head>

<meta charset=utf-8 />

<title>Return first and last name from a form - w3resource</title>

</head><body>

<form id="form1" onsubmit="getFormvalue()">

First name: <input type="text" name="fname" value="David"><br>

Last name: <input type="text" name="lname" value="Beckham"><br>

<input type="submit" value="Submit">

</form>

</body>

</html>

```

Correction 1 - Affichage dans un message d'alerte:

```
function getFormvalue()
{
    var x=document.getElementById("form1");
    for (var i=0;i<x.length;i++)
    {
        if (x.elements[i].value!='Submit')
        {
            alert(x.elements[i].value);
        }
    }
}
```

Correction 2 - Affichage dans la page html :

- Rajouter la balise en dehors du formulaire :

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

```
function getFormvalue() {
    var x = document.getElementById("form1");
    for (var i = 0; i < x.length; i++) {
        if (x.elements[i].value != 'Submit') {
            document.getElementById("result").innerHTML = x.elements[0].value + " " +
x.elements[1].value;
        }
    }
}
```

Exercice 4 :

À un document HTML, ajouter des alertes :

- Au chargement de la page (attribut **onload**)
- Quand on passe sur une image (attribut **onmouseover**)
- Quand on clique sur un bouton (attribut **onclick**)

Avec des boutons :

- Changer la couleur de fond quand on clique sur un bouton
- Changer aussi l'apparence du bouton
- Ajouter un bouton qui permette de revenir à la normale.

Sur des images :

- Changer une image par une autre quand la souris passe dessus (attribut **onmouseover**)
- Remettre l'image d'origine quand la souris quitte l'image (attribut **onmouseout**)

Correction :

```
<!DOCTYPE html>

<html lang="fr">

<head>

<title>Actions utilisateur et JavaScript</title>

<meta charset="utf-8" />

<link rel="stylesheet" href="idemm.css" />

<script>

function retournormal () {

document.documentElement.style.backgroundColor='white';

document.documentElement.style.color='black';

document.getElementById('bnuit').style.color='black';

}

</script>
```

```

</head>

<body onload="alert('bienvenue...');">

<h1>Actions utilisateur et JavaScript</h1>

<p>

Bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

bla bla bla bla bla bla bla bla bla bla bla bla bla bla bla

</p>


<div class="illustration">



</div>

<button onclick="alert('clac !');">cliquez ici !</button>

<br />

<button id="bnuit"

onclick="document.documentElement.style.backgroundColor='black';

document.documentElement.style.color='white';

document.getElementById('bnuit').style.color='red';">

nuit</button>

<button onclick="retournormal();">jour</button>

<br />

<div class="illustration">



```

```
</div>
</body>
</html>
```

Exercice 5 :

Un premier controle de formulaire

1. Ecrire une page HTML comportant un mini-formulaire compos e d'un champ de saisie et d'un simple bouton.

2. Creer un fichier exo1.js

3. Rajouter l'instruction precedente par :

```
function controler() {
var zoneSaisie = document.getElementById("zoneSaisie");
alert("La Zone de saisie contient : " + zoneSaisie.value);
}
```

Decrivez le code de cette fonction. Il faudrait maintenant associer ce code de controle a l'evenement click sur le bouton du formulaire. Ecrivez le code correspondant.

4. Modifier la fonction de controle pour afficher un message d'erreur lorsque le champ est vide, et afficher le contenu du champ lorsqu'il ne l'est pas.

5. La fonction eval, prenant en parametre une chaine de caracteres representant une expression, permet d'evaluer cette derniere.

Par exemple :

```
eval("3+4"); // retourne la valeur 7
eval("bon"+"jour"); //retourne bonjour
```

Ajouter (concatenez) le resultat de l'evaluation de l'expression donn ee par l'utilisateur dans le champ de saisie a votre affichage. Cela donne une page HTML contenant un formulaire permettant d'executer "online" de petites operations.

Correction :

index.html

```
<!DOCTYPE html>
```



```

<html lang="fr">

<head>

  <title>Actions utilisateur et JavaScript</title>

  <meta charset="utf-8" />

  <link rel="stylesheet" href="idemm.css" />

</head>

<body>

  <form method="get" action="" id="myForm">

    <p>

      <label for="zoneSaisie "> Entrer une valeur : </label>

      <input type="text" id="zoneSaisie" name="zoneSaisie" />

    </p>

    <p>

      <input type="button" value="ok" id="ok" />

    </p>

  </form>

</body>

</html>

```

exo1.js :

```

function controler() {

  var zoneSaisie = document.getElementById("zoneSaisie");

  if (zoneSaisie.value == "")

    alert("Saisissez une valeur");

  else {

    alert("La Zone de saisie contient : " + zoneSaisie.value + "et a pour evaluation " +
    eval(zoneSaisie.value));
  }
}

```

```
    }  
}  
  
var ok = document.getElementById("ok");  
  
ok.onclick = controler;
```