

function
var

FORMATIONS

JAVA

SCRIPT

DÉBUTANT



FORMATION JAVASCRIPT ENRICHIE

L'interactivité du web

CHAPITRE 1 — Introduction à JavaScript

JavaScript (JS) est le langage de programmation qui donne VIE à vos pages web. Si HTML est la structure et CSS le style, JavaScript est le COMPORTEMENT et l'INTERACTIVITÉ.

Comprendre JavaScript

Sans JavaScript, votre site est STATIQUE (ne bouge pas). Avec JavaScript, vous pouvez :

- Réagir aux clics de l'utilisateur
- Modifier le contenu de la page
- Créer des animations
- Valider des formulaires
- Charger des données sans recharger la page

Où placer le JavaScript ?

1. Dans la balise <script> (dans le HTML)

 index.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Ma page</title>
</head>
<body>

  <h1 id="titre">Bonjour !</h1>
  <button onclick="direBonjour()">Cliquez-moi</button>

  <!-- Script à la FIN du body (recommandé) -->
  <script>
    function direBonjour() {
      alert('Bonjour le monde !');
    }
  </script>

</body>
</html>
```

2. Dans un fichier externe ★ RECOMMANDÉ

Le JavaScript est dans un fichier .js séparé. Plus propre et réutilisable !

 index.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Ma page</title>
</head>
<body>

    <h1 id="titre">Bonjour !</h1>
    <button id="monBouton">Cliquez-moi</button>

    <!-- Lien vers le fichier JS externe -->
    <script src="script.js"></script>

</body>
</html>
```

 script.js

```
// Fichier JavaScript externe

// Attendre que la page soit chargée
document.addEventListener('DOMContentLoaded', function() {

    // Récupérer le bouton
    const bouton = document.getElementById('monBouton');

    // Ajouter un événement au clic
    bouton.addEventListener('click', function() {
        alert('Bonjour le monde !');
    });

});
```

Position du <script>

Placez TOUJOURS votre <script> à la FIN du <body>, juste avant </body>.

Pourquoi ? Pour que le HTML soit chargé AVANT que le JavaScript s'exécute.

Sinon, JavaScript ne trouvera pas les éléments !

CHAPITRE 2 — Variables et Types de données

Les variables sont des BOÎTES qui stockent des informations. On leur donne un nom pour les retrouver facilement.

Déclarer des variables

script.js

```
// 3 façons de déclarer une variable

// 1. let (recommandé pour les variables qui CHANGENT)
let age = 25;
let nom = 'Marie';
console.log(age); // Affiche : 25

age = 26; // On peut changer la valeur
console.log(age); // Affiche : 26

// 2. const (recommandé pour les variables CONSTANTES)
const PI = 3.14159;
const ville = 'Paris';
console.log(PI); // Affiche : 3.14159

// PI = 3.14; // ERREUR ! On ne peut pas changer une const

// 3. var (ancienne méthode, évitez-la)
var ancien = 'Ne pas utiliser';

// ✅ UTILISEZ let et const, pas var !
```

Types de données

Types en JavaScript

```
// 1. STRING (texte) - entre guillemets
let nom = 'Jean';
```

```
let prenom = "Marie";
let phrase = `Bonjour ${nom}`; // Template literal (backticks)

// 2. NUMBER (nombre)
let age = 25;
let prix = 19.99;
let negatif = -10;

// 3. BOOLEAN (vrai/faux)
let estMajeur = true;
let estMineur = false;

// 4. ARRAY (tableau/liste)
let fruits = ['pomme', 'banane', 'orange'];
let nombres = [1, 2, 3, 4, 5];
console.log(fruits[0]); // Affiche : pomme (index 0)

// 5. OBJECT (objet)
let personne = {
    nom: 'Dupont',
    prenom: 'Jean',
    age: 30,
    ville: 'Paris'
};
console.log(personne.nom); // Affiche : Dupont

// 6. NULL et UNDEFINED
let vide = null; // Valeur volontairement vide
let nonDefini; // undefined (pas de valeur assignée)
```

CHAPITRE 3 — Manipulation du DOM

Le DOM (Document Object Model) est la représentation de votre page HTML que JavaScript peut manipuler. C'est comme un arbre où chaque élément HTML est une branche.

Sélectionner des éléments

HTML de départ

```
<!DOCTYPE html>
<html>
<body>

<h1 id="titre">Mon titre</h1>
<p class="texte">Paragraphe 1</p>
<p class="texte">Paragraphe 2</p>
<button id="monBouton">Cliquer</button>

<script src="script.js"></script>
</body>
</html>
```

script.js - Sélection

```
// 1. Par ID (renvoie 1 élément)
const titre = document.getElementById('titre');
console.log(titre); // <h1 id="titre">Mon titre</h1>

// 2. Par CLASSE (renvoie une liste)
const paragraphes = document.getElementsByClassName('texte');
console.log(paragraphes); // HTMLCollection de 2 <p>

// 3. Par sélecteur CSS ★ MODERNE et RECOMMANDÉ
const titre2 = document.querySelector('#titre');
const premierP = document.querySelector('.texte');
const tousLesP = document.querySelectorAll('.texte');
```

```
// querySelector = 1er élément trouvé  
// querySelectorAll = TOUS les éléments
```

Modifier le contenu

Exemple complet HTML + JS

```
<!DOCTYPE html>  
<html>  
<body>  
  
<h1 id="titre">Ancien titre</h1>  
<p id="description">Ancienne description</p>  
<button onclick="changerTexte()">Changer le texte</button>  
  
<script>  
    function changerTexte() {  
        // Modifier le texte du titre  
        const titre = document.getElementById('titre');  
        titre.textContent = 'NOUVEAU TITRE !';  
  
        // Modifier le HTML complet  
        const desc = document.getElementById('description');  
        desc.innerHTML = '<strong>Nouvelle</strong> description !';  
  
        // Modifier le style  
        titre.style.color = 'blue';  
        titre.style.fontSize = '48px';  
    }  
</script>  
  
</body>  
</html>
```

 **textContent vs innerHTML**

- `textContent` : Texte BRUT uniquement (plus sûr) • `innerHTML` : Accepte les balises HTML Attention avec `innerHTML` : si le contenu vient de l'utilisateur, il peut injecter du code malveillant ! Préférez `textContent` quand possible.

CHAPITRE 4 — Les Événements

Les événements permettent à JavaScript de RÉAGIR aux actions de l'utilisateur : clics, frappes clavier, survol de souris, etc.

addEventListener() - La méthode moderne

Exemple complet - Bouton interactif

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    font-family: Arial, sans-serif;
    padding: 50px;
}

#monBouton {
    background: #3498db;
    color: white;
    padding: 15px 30px;
    border: none;
    border-radius: 5px;
    font-size: 18px;
    cursor: pointer;
    transition: background 0.3s;
}

#monBouton:hover {
    background: #2980b9;
}

#compteur {
    font-size: 48px;
    font-weight: bold;
    color: #2c3e50;
    margin: 20px 0;
```

```
        }
    </style>
</head>
<body>

<h1>Compteur de clics</h1>
<div id="compteur">0</div>
<button id="monBouton">Cliquez-moi !</button>

<script>
    // Variable pour compter les clics
    let nombreClics = 0;

    // Récupérer les éléments
    const bouton = document.getElementById('monBouton');
    const compteur = document.getElementById('compteur');

    // Ajouter l'événement de clic
    bouton.addEventListener('click', function() {
        // Incrémenter le compteur
        nombreClics++;

        // Mettre à jour l'affichage
        compteur.textContent = nombreClics;

        // Changer la couleur selon le nombre
        if (nombreClics >= 10) {
            compteur.style.color = 'red';
        } else if (nombreClics >= 5) {
            compteur.style.color = 'orange';
        }
    });
</script>

</body>
</html>
```

Autres événements courants

Exemples d'événements

```
// 1. CLICK (clic de souris)
element.addEventListener('click', function() {
  console.log('Élément cliqué !');
});

// 2. MOUSEOVER / MOUSEOUT (survol)
element.addEventListener('mouseover', function() {
  this.style.background = 'yellow';
});

element.addEventListener('mouseout', function() {
  this.style.background = 'white';
});

// 3. KEYPRESS (frappe clavier)
input.addEventListener('keypress', function(event) {
  console.log('Touche pressée :', event.key);
});

// 4. SUBMIT (soumission formulaire)
formulaire.addEventListener('submit', function(event) {
  event.preventDefault(); // Empêche le rechargement
  console.log('Formulaire soumis !');
});

// 5. CHANGE (changement de valeur)
select.addEventListener('change', function() {
  console.log('Nouvelle valeur :', this.value);
});
```

Exemple pratique : Changer la couleur de fond

 HTML + CSS + JavaScript complet

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    font-family: Arial, sans-serif;
    text-align: center;
    padding: 50px;
    transition: background-color 0.5s;
}

h1 {
    color: white;
    text-shadow: 2px 2px 4px rgba(0,0,0,0.5);
}

.btn {
    padding: 15px 30px;
    margin: 10px;
    border: none;
    border-radius: 8px;
    font-size: 18px;
    cursor: pointer;
    color: white;
    font-weight: bold;
}

#btnRouge { background: #e74c3c; }
#btnBleu { background: #3498db; }
#btnVert { background: #2ecc71; }

</style>
</head>
```

```
<body>

    <h1>Changeur de couleur</h1>
    <p style="color: white; font-size: 20px;">
        Cliquez sur un bouton pour changer la couleur de fond
    </p>

    <button class="btn" id="btnRouge">Rouge</button>
    <button class="btn" id="btnBleu">Bleu</button>
    <button class="btn" id="btnVert">Vert</button>

    <script>
        // Récupérer les boutons
        const btnRouge = document.getElementById('btnRouge');
        const btnBleu = document.getElementById('btnBleu');
        const btnVert = document.getElementById('btnVert');

        // Ajouter les événements
        btnRouge.addEventListener('click', function() {
            document.body.style.backgroundColor = '#c0392b';
        });

        btnBleu.addEventListener('click', function() {
            document.body.style.backgroundColor = '#2980b9';
        });

        btnVert.addEventListener('click', function() {
            document.body.style.backgroundColor = '#27ae60';
        });
    </script>

</body>
</html>
```

 **Testez vos créations !**

Copiez ce code dans un fichier .html, ouvrez-le dans votre navigateur, et testez ! Le meilleur moyen d'apprendre JavaScript est de PRATIQUER et d'expérimenter.

✨ Formation JavaScript ENRICHIE terminée ! ✨

Continuez à pratiquer, testez tous les exemples, et créez vos propres projets !