

JavaScript Côté Client

Introduction

JavaScript est un langage de programmation essentiel pour le développement web. Utilisé principalement côté client, il permet de rendre les pages web interactives et dynamiques.

Table des Matières

1. [Les Bases de JavaScript](#)
2. [Manipulation du DOM](#)
3. [Gestion des Événements](#)
4. [Ajax et Fetch API](#)
5. [LocalStorage et SessionStorage](#)
6. [Bibliothèques et Frameworks](#)
7. [Bonnes Pratiques](#)
8. [Ressources Utiles](#)

Les Bases de JavaScript

Syntaxe de Base

```
// Déclaration de variables
var nom = "Damien";
let age = 33;
const PI = 3.14;

// Fonctions
```

```
function saluer(prenom) {
    return `Bonjour, ${prenom}!`;
}

// Conditions
if (age > 18) {
    console.log("Adulte");
} else {
    console.log("Mineur");
}

// Boucles
for (let i = 0; i < 5; i++) {
    console.log(i);
}
```

Types de Données

- **Number:** `let x = 5;`
- **String:** `let str = "Hello";`
- **Boolean:** `let isTrue = true;`
- **Array:** `let arr = [1, 2, 3];`
- **Object:** `let obj = { name: "John", age: 30 };`

Opérateurs

- **Arithmétiques:** `+`, `-`, `*`, `/`, `%`
- **Comparaison:** `==`, `===`, `!=`, `!==`, `>`, `<`, `>=`, `<=`
- **Logiques:** `&&`, `||`, `!`

Manipulation du DOM

Sélection d'Éléments

```
let element = document.getElementById("id");
let elements = document.getElementsByClassName("classe");
let elements = document.getElementsByTagName("div");
let element = document.querySelector(".classe");
let elements = document.querySelectorAll(".classe");
```

Modification du Contenu

```
let element = document.getElementById("id");
element.innerHTML = "Nouveau contenu";
element.textContent = "Nouveau contenu";
```

Modification des Styles

```
let element = document.getElementById("id");
element.style.color = "red";
element.style.backgroundColor = "blue";
```

Création et Suppression d'Éléments

```
// Création
let nouveauDiv = document.createElement("div");
nouveauDiv.textContent = "Bonjour!";
document.body.appendChild(nouveauDiv);

// Suppression
let element = document.getElementById("id");
element.parentNode.removeChild(element);
```

Gestion des Événements

Ajout d'Événements

```
let bouton = document.getElementById("bouton");
bouton.addEventListener("click", function() {
    alert("Bouton cliqué!");
});
```

Types d'Événements Courants

- `click`
- `mouseover`
- `mouseout`
- `keydown`
- `keyup`
- `submit`

Ajax et Fetch API

Utilisation de XMLHttpRequest

```
let xhr = new XMLHttpRequest();
xhr.open("GET", "https://api.example.com/data", true);
xhr.onload = function() {
    if (xhr.status === 200) {
        console.log(xhr.responseText);
    }
};
xhr.send();
```

Utilisation de Fetch API

```
fetch("https://api.example.com/data")
  .then(response => response.json())
  .then(data => console.log(data))
  .catch(error => console.error("Erreur:", error));
```

LocalStorage et SessionStorage

LocalStorage

```
// Enregistrer des données
localStorage.setItem("prenom", "Damien");

// Récupérer des données
let nom = localStorage.getItem("prenom");

// Supprimer des données
localStorage.removeItem("prenom");

// Vider le LocalStorage
localStorage.clear();
```

SessionStorage

```
// Enregistrer des données
sessionStorage.setItem("prenom", "Damien");

// Récupérer des données
let nom = sessionStorage.getItem("prenom");

// Supprimer des données
sessionStorage.removeItem("prenom");
```

```
// Vider le sessionStorage
sessionStorage.clear();
```

Bibliothèques et Frameworks

jQuery

```
$(document).ready(function() {
    $("#bouton").click(function() {
        alert("Bouton cliqué!");
    });
});
```

React

```
import React from "react";
import ReactDOM from "react-dom";

function App() {
    return <h1>Bonjour, monde!</h1>;
}

ReactDOM.render(<App />, document.getElementById("root"));
```

Ressources Utiles

- [MDN Web Docs](#)
- [W3Schools](#)

- [JavaScript.info](#)