

# Cours d'Intro à JavaScript



```
alert("pop")
```

1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
0  
0

1  
0  
1  
0  
0  
1  
0  
1  
0  
1  
0  
0  
1  
0  
1  
0



# Table des matières

01

Introduction

02

Variables et  
Types de Données

03

Opération et  
Conditions

04

Les Boucles

05

Les Fonctions

06

Manipulation du  
DOM

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0



01

# Introduction à Javascript

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0



# Qu'est-ce que JavaScript

**Définition:** JavaScript est un langage de programmation orienté objet, utilisé principalement pour rendre les pages web interactives et dynamiques. Il fonctionne côté client (navigateur) mais peut également s'exécuter côté serveur grâce à des environnements comme Node.js.

## Historique:

1995 : Création par Brendan Eich chez Netscape sous le nom de Mocha, renommé rapidement en JavaScript.

1996 : Microsoft lance une version concurrente appelée JScript, entraînant des différences d'implémentation.

1997 : Standardisation par l'organisme ECMA International, donnant naissance à ECMAScript (base de JavaScript).

2009 : Introduction de Node.js, permettant d'utiliser JavaScript en dehors du navigateur.

2015 : Publication d'ECMAScript 6 (ES6), une mise à jour majeure avec de nouvelles fonctionnalités modernes.

# Différence entre HTML, CSS et JavaScript



## HTML

HTML est le langage utilisé pour structurer le contenu d'une page web. Il permet de créer des éléments comme les titres, paragraphes, images et liens pour construire l'ossature d'un site.



## CSS

CSS est un langage de style qui contrôle l'apparence visuelle d'une page web. Il permet de définir les couleurs, polices, marges, alignements et bien plus pour rendre le site attrayant.



## JavaScript

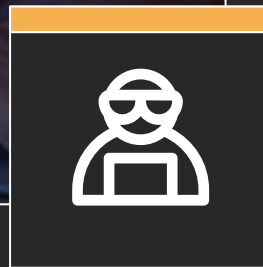
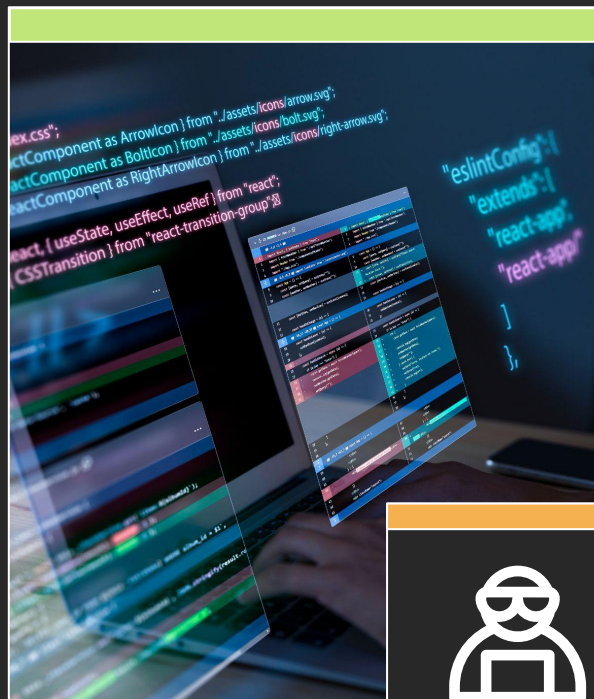
JavaScript est un langage de programmation qui ajoute de l'interactivité et de la dynamisme aux pages web. Il permet de créer des animations, des formulaires interactifs et des fonctionnalités avancées.

1  
0  
0  
1  
0  
1  
0  
1  
0  
0  
0  
1  
0  
1  
0  
1  
01  
0  
1  
0  
0  
1  
0  
0  
1  
0  
1  
0  
1  
1  
0  
1  
1

# Où écrire du JavaScript ?

2 Façons principale de le faire

- **Console du navigateur:** avec `console.log()`
- **Une page Web:** Inclusion d'un script avec la balise `<script>` dans la page HTML





# Activité 1



## Hello World

Ouvrir la console du navigateur et  
afficher "Hello World" avec  
`console.log()`

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



02

# Variables, Types de Données

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0





# Déclaration d'une variable

Caractéristique	Let	Const	Var
Réaffectation	Oui	Non	Oui
Portée	Bloc	Bloc	Fonction/Globale
Initialisé	Non	Non	Oui mais à undefined

⚠ **Note:** Var est l'ancienne manière de déclarer des variables. Elle est désormais dépréciée.

# Types de données

## Nombre

Représente des valeurs numériques, qu'elles soient entières ou décimales.

## Null

Représente une absence volontaire de valeur.

## Chaîne de caractères

Représente du texte, entouré de guillemets simples (') ou doubles (").

## Booléen

Représente une valeur vraie ou fausse.

## Undefined

Indique qu'une variable a été déclarée mais n'a pas encore de valeur attribuée.

⚠ **Note:** Vous pouvez utiliser l'opérateur `typeof` pour connaître le type d'une donnée

1  
0  
0  
1  
0  
1  
0  
1  
0  
0  
1  
0  
1  
0  
1  
0

1  
0  
1  
0  
0  
1  
0  
0  
1  
0  
1  
1  
0  
1  
1  
1



# Activité 2



## Catch them all

Créer et afficher des variables avec différents types de données dans la console.

1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0  
0  
0  
0  
1  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



03

# Opérations et Conditions

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0



# Opérations de base

## Addition

let somme = 5 + 3; // 8

## Multiplication

let produit = 6 \* 7; // 42

## Soustraction

let difference = 10 - 4; // 6

## Division

let quotient = 20 / 5; // 4

## Modulo

let reste = 10 % 3; // 1

1  
0  
0  
1  
0  
1  
0  
1  
0  
0  
1  
0  
1  
0  
1  
0

1  
0  
1  
0  
0  
1  
0  
0  
1  
0  
1  
1  
0  
1  
1  
1

# Conditions ( **if**, **else if**, **else** )

**Astuce** : Opérateur de  
comparaison

**===** : Égalité stricte

**!==** : Différence stricte

**<, >, <=, >=** : Comparaisons  
numériques

```
if (condition) {  
    // Code exécuté si la condition est vraie  
} else if (autreCondition) {  
    // Code exécuté si l'autre condition est vraie  
} else {  
    // Code exécuté si aucune condition n'est vraie  
}
```



# Activité 3



## Majeur ou non ?

Écrire un programme qui vérifie si  
une personne est majeure

1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



04

# Les Boucles

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0





# Répéter des actions

## For

```
for (initialisation; condition; incrément) {  
    // Code à exécuter  
}
```

## While

```
while (condition) {  
    // Code à exécuter  
}
```

1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



# Activité 4



## 1 à 10

Écrire un programme qui affiche  
les nombres de 1 à 10 dans la  
console.

1  
0  
1  
1  
0  
1  
0  
1  
1  
1  
1  
0  
0  
0  
0  
1  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0



05

# Les Fonctions

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0



# Qu'est-ce qu'une fonction

**Définition:** Une fonction est un bloc de code réutilisable qui exécute une tâche ou calcule une valeur. Elle permet de structurer et d'optimiser le code.

## Caractéristiques des fonctions

- **Paramètres** : Valeurs passées à la fonction (facultatif).

```
function bonjour(nom) {  
  console.log("Bonjour " + nom);  
}
```

- **Retour (return)** : Résultat renvoyé par la fonction (facultatif).

- **Réutilisabilité** : Une fonction peut être appelée plusieurs fois.

```
function nomDeLaFonction(param1,  
param2) {  
  // Instructions  
  return resultat; // Optionnel  
}
```



# Activité 5



## Code ton Calcul : Aire d'un Rectangle

Créer une fonction qui calcule l'aire  
d'un rectangle (longueur  $\times$  largeur).

1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



06

# Manipulation du DOM

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0

1  
0  
1  
0  
1  
0  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0



# Accéder à des éléments HTML

## document.getElementById()

```
let titre = document.getElementById("monTitre");
console.log(titre.textContent);
```

Sélectionne un élément par son **id**.

Exemple de l'HTML:

```
<h1 id="monTitre">Bienvenue</h1>
```

## document.querySelector()

```
let paragraphe =
document.querySelector(".monParagraphe");
console.log(paragraphe.textContent);
```

Sélectionne le **premier** élément correspondant à un sélecteur CSS.

Exemple de l'HTML:

```
<p class="monParagraphe">lorem ipsum</p>
```

# Modifier le contenu et le style

## Modifier le contenu

```
let titre = document.getElementById("monTitre");
titre.textContent = "Nouveau Titre !"; // Change le texte
titre.innerHTML = "<em>Titre en italique</em>"; //
// Accepte du HTML
```

## Modifier le style

```
let bouton = document.querySelector(".btn");
bouton.style.backgroundColor = "blue"; // Change la
// couleur de fond
bouton.style.color = "white"; // Change la couleur du
// texte
```

⚠ **Note:** `innerHTML` est déprécié par rapport à `textContent` car moins sécurisé. Pour ajouter de l'HTML se référer à `createElement()` and `appendChild()`.





# Activité 6



## Attention Click

Créer une page HTML avec un bouton. Utiliser JavaScript pour afficher une alerte quand le bouton est cliqué.

1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0  
0  
0  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
0



# Activité 7



## Projet Final

Créer une calculatrice simple avec  
**HTML**, **CSS** et **JavaScript**.

Fonctionnalités : **addition**,  
**soustraction**, **multiplication**,  
**division**.

1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0  
0  
0  
0  
1  
1  
1  
1  
0

1  
1  
0  
0  
1  
0  
1  
0  
1  
0  
1  
0  
1  
1  
1  
1  
0



# Webographie

1 **Documentation officielle** : <https://developer.mozilla.org/fr>

0

0 **Lien du github** : <https://github.com/HotTeaDrink/Tutoring/tree/main>

1

0

1

0

1

0

0

0

1

0

1

0

1

0

1

0

1

0

0

1

0

0

1

0

1

1

0

1

1

1

1