

Les Fondamentaux de JavaScript

1.1 Introduction

Ce cours présente les concepts de base de JavaScript en les comparant avec Python, que vous connaissez déjà. JavaScript est le langage de programmation du web : il permet de rendre les pages interactives.

Principales différences avec Python :

- JavaScript s'exécute dans le navigateur (côté client)
 - La syntaxe utilise des accolades {} au lieu de l'indentation
 - Il existe plusieurs façons de déclarer des variables
 - JavaScript manipule directement les éléments HTML (le DOM)
-

1.2 Les Variables : let, const et var

1.2.1 En Python (rappel)

En Python, on déclare une variable simplement :

```

1 # Une variable simple
2 age = 15
3 nom = "Alice"
4
5 # On peut la modifier
6 age = 16

```

1.2.2 En JavaScript : trois mots-clés

JavaScript propose **trois façons** de déclarer une variable :

```

1 let age = 15;           // Variable qui PEUT changer
2 const nom = "Alice";   // Constante qui NE PEUT PAS changer
3 var ancien = 10;        // Ancienne syntaxe (à éviter)

```

Le *typage* est dynamique, comme en *python* : il n'est pas besoin de préciser le type.

Règle d'utilisation

Mot-clé	Usage	Peut changer ?	Recommandation
const	Valeur fixe	[X] Non	À utiliser par défaut
let	Valeur variable	[V] Oui	Quand la valeur change
var	Ancienne syntaxe	[V] Oui	À éviter (obsolète)

Exemples concrets

```

1 // Utiliser const par défaut
2 const pi = 3.14159;           // Ne changera jamais
3 const nom = "Alice";         // Ne changera jamais
4 const bouton = document.getElementById('monBouton'); // Ne
   changera jamais
5
6 // Utiliser let quand ça change
7 let compteur = 0;             // Va augmenter
8 let temperature = 20;         // Peut varier
9 let score = 0;                // Va évoluer
10
11 // Erreur : on ne peut pas modifier une const
12 const age = 15;
13 age = 16; // ERREUR : Assignment to constant variable
14
15 // Correct : on peut modifier un let
16 let age = 15;
17 age = 16; // OK

```

Remarque : A moins que vous soyez sûr que la valeur puisse être modifiée, déclarez vos variables avec `const`, puis changez en `let` seulement si le compilateur vous indique une erreur.

1.2.3 Quelques différences Python ↔ JavaScript

Python	JavaScript
<code>age = 15</code>	<code>let age = 15;</code>
PI = 3.14 (convention)	<code>const PI = 3.14;</code> (imposé)
Pas de point-virgule	Point-virgule ; recommandé
<code>print(age)</code>	<code>console.log(age)</code>

1.3 Le DOM : Document Object Model

1.3.1 Qu'est-ce que le DOM ?

Le DOM est la représentation en mémoire de votre page HTML. JavaScript peut :

- Lire les éléments HTML
- Les modifier
- Créer de nouveaux éléments
- Réagir aux actions de l'utilisateur

Analogie avec Python :

```

1 # En Python, vous manipulez des fichiers
2 fichier = open('data.txt', 'r')
3 contenu = fichier.read()
4 fichier.close()

5
6 # En JavaScript, vous manipulez des éléments HTML
7 element = document.getElementById('monId')
8 contenu = element.textContent

```

1.3.2 Récupérer un élément : getElementById()

HTML de départ

Placer un attribut id aux balises que l'on veut *indexer*.

```

1 <h1 id="titre">Bienvenue</h1>
2 <p id="description">Ceci est un paragraphe</p>
3 <button id="monBouton">Cliquez-moi</button>

```

JavaScript

```

1 <script>
2 // Récupérer les éléments
3 const titre = document.getElementById('titre');
4 const description = document.getElementById('description');
5 const bouton = document.getElementById('monBouton');

6
7 console.log(titre); // Affiche: <h1 id="titre">Bienvenue</h1>
8 </script>

```

A faire vous-même (1) : Placer les 2 parties du script ci-dessus dans une même page (page.html). Ouvrir avec un navigateur.

Remarque importante : L'ID doit être **exactement** le même (sensible à la casse)

```

1 // Correct
2 const element = document.getElementById('monBouton');
3 console.log(element); // Affiche <button id="monBouton">Cliquez-
4                         moi</button>

5 // Erreur : mauvaise casse, ou ID inexistant
6 const element = document.getElementById('MonBouton');
7 console.log(element); // Affiche: null

```

1.4 Attributs vs Propriétés

1.4.1 Comprendre la différence

Un élément HTML possède :

- Des **attributs** (définis dans le HTML)
- Des **propriétés** (accessibles en JavaScript)

Exemple HTML

```
1 <input type="text" id="champNom" value="Alice">
```

Les attributs (dans le HTML)

- `type="text"`
- `id="champNom"`
- `value="Alice"`

Les propriétés (en JavaScript)

```
1 const champ = document.getElementById('champNom');

2

3 // Propriétés disponibles :
4 champ.type          // "text"
5 champ.id            // "champNom"
6 champ.value         // "Alice" (peut être modifié par l'utilisateur
7 )
8 champ.textContent // ""
9 champ.style         // Objet avec toutes les propriétés CSS
```

1.4.2 La notation pointée

En JavaScript, on accède aux propriétés avec un **point** (comme en Python) :

```
1 # Python: notation pointée pour objet.attribut ou objet.methode()
2 for event in pygame.event.get():
3     if event.type == pygame.QUIT:
4         running = False
```

```
1 // JavaScript : notation pointée pour les éléments HTML
2 const element = document.getElementById('titre');

3
4 // Lire une propriété
5 console.log(element.textContent);    // "Bienvenue"
6 console.log(element.id);           // "titre"
7
```

```

8 // Modifier une propriété
9 element.textContent = "Nouveau titre";
10 element.style.color = "red";

```

1.4.3 Propriétés courantes

textContent : Le texte à l'intérieur

```

1 <p id="message">Bonjour tout le monde</p>

```

A faire vous-même (2) : Placer dans votre page.html le paragraphe Bonjour tout le monde. Ouvrir la console Web et placer l'une après l'autre chacune des instructions javascript ci-dessous.

```

1 const message = document.getElementById('message');
2
3 // Lire le texte
4 console.log(message.textContent); // "Bonjour tout le monde"
5
6 // Modifier le texte
7 message.textContent = "Nouveau message";
8 // Le HTML devient : <p id="message">Nouveau message</p>

```

value : La valeur d'un champ de saisie

```

1 <input type="text" id="nom" value="Alice">
2 <input type="range" id="age" min="0" max="100" value="25">
3 <script>
4 const champNom = document.getElementById('nom');
5 const curseurAge = document.getElementById('age');
6 </script>

```

A faire vous-même (3) : Tester cet exemple : Placer le script dans votre page.html. Puis écrire chacune de ces instructions dans la console :

```

1 // Lire les valeurs
2 console.log(champNom.value); // "Alice"
3 console.log(curseurAge.value); // "25"
4
5 // Modifier les valeurs
6 champNom.value = "Bob";
7 curseurAge.value = 30;

```

style : Modifier le CSS

```

1 <h1 id="titre">Le Javascript c'est simple</h1>

```

A faire vous-même (4) : Tester cet exemple : Placer le script dans votre page.html. Puis écrire chacune de ces instructions dans la console :

```

1 const element = document.getElementById('titre');

2

3 // Modifier le style
4 element.style.color = "blue";
5 element.style.fontSize = "24px";
6 element.style.backgroundColor = "yellow";

7

8 // Note: les propriétés CSS avec tirets deviennent camelCase
9 // CSS: background-color → JavaScript: backgroundColor
10 // CSS: font-size → JavaScript: fontSize

```

1.4.4 Tableau récapitulatif

Propriété	Type d'élément	Usage
textContent	Tous	Lire/modifier le texte
value	Input, textarea, select	Lire/modifier la valeur saisie
style	Tous	Modifier le CSS
id	Tous	Lire/modifier l'ID
className	Tous	Lire/modifier les classes CSS

1.5 Comparaison complète Python ↔ JavaScript

1.5.1 Variables

Concept	Python	JavaScript
Déclarer une variable	x = 10	let x = 10;
Constante	PI = 3.14 (convention)	const PI = 3.14;
Chaîne de caractères	"texte" ou 'texte'	"texte" ou 'texte'
Concaténation	"Hello " + nom	"Hello " + nom
Affichage console	print(x)	console.log(x);

1.5.2 Types de données

Type	Python	JavaScript
Entier	age = 15	let age = 15;
Décimal	prix = 19.99	let prix = 19.99;
Texte	nom = "Alice"	let nom = "Alice";
Booléen	actif = True	let actif = true;

Type	Python	JavaScript
Rien/Null	None	null

Attention : True/False en Python → true/false en JavaScript (minuscules !)

1.5.3 Opérateurs

Opération	Python	JavaScript
Addition	x + 1	x + 1
Incrémation	x += 1	x++ ou x += 1
Égalité	x == 5	x === 5
Different	x != 5	x !== 5
Et logique	and	&&
Ou logique	or	
Négation	not	!

[!] Important : En JavaScript, préférez toujours === et !== (égalité stricte)

1.6 Fonctions

1.6.1 Déclaration

```

1 # Python
2 def dire_bonjour(nom):
3     print(f"Bonjour {nom}")
4
5 dire_bonjour("Alice")

```

```

1 // JavaScript - Syntaxe classique
2 function direBonjour(nom) {
3     console.log("Bonjour " + nom);
4 }
5
6 direBonjour("Alice");
7
8 // JavaScript - Fonction fléchée (moderne)
9 const direBonjour = (nom) => {
10     console.log("Bonjour " + nom);
11 };
12
13 direBonjour("Alice");

```

1.6.2 Fonctions sans paramètres

```

1 # Python
2 def afficher_message():
3     print("Message affiché")
4
5
6 // JavaScript
7 function afficherMessage() {
8     console.log("Message affiché");
9 }
10
11 // Ou avec fonction fléchée
12 const afficherMessage = () => {
13     console.log("Message affiché");
14 };

```

1.6.3 Retour de valeur

```

1 # Python
2 def additionner(a, b):
3     return a + b
4
5 resultat = additionner(5, 3) # 8

```

```

1 // JavaScript
2 function additionner(a, b) {
3     return a + b;
4 }
5
6 const resultat = additionner(5, 3); // 8

```

1.7 Exemple complet : Compteur

1.7.1 Version Python (pour comparaison)

```

1 compteur = 0
2
3 def incrementer():
4     global compteur
5     compteur += 1
6     print(f"Compteur: {compteur}")
7
8 incrementer() # Affiche: Compteur: 1
9 incrementer() # Affiche: Compteur: 2

```

1.7.2 Version JavaScript (interaction avec HTML)

```

1 <button id="bouton">Cliquer</button>
2 <p>Compteur: <span id="affichage">0</span></p>

1 // Récupération des éléments
2 const bouton = document.getElementById('bouton');
3 const affichage = document.getElementById('affichage');

4
5 // Variable compteur
6 let compteur = 0;

7
8 // Fonction d'incrémentation
9 function incrementer() {
10     compteur++;
11     affichage.textContent = compteur;
12 }
13
14 // Associer la fonction au clic
15 bouton.addEventListener('click', incrementer);

```

Explication ligne par ligne :

1. `const bouton = ...` : Récupère l'élément bouton (ne changera jamais)
 2. `const affichage = ...` : Récupère l'élément d'affichage (ne changera jamais)
 3. `let compteur = 0` : Variable qui va changer (on utilise `let`)
 4. `function incrementer()` : Déclare la fonction
 5. `compteur++` : Ajoute 1 au compteur (équivalent à `compteur += 1`)
 6. `affichage.textContent = compteur` : Met à jour le texte affiché
 7. `addEventListener('click', incrementer)` : "Quand on clique, exécute la fonction"
-

1.8 Exercices d'application

1.8.1 Exercice 1 : Variables

Complétez le code suivant :

```

1 // 1. Déclarez une constante PI valant 3.14159
2
3 // 2. Déclarez une variable score initialisée à 0
4
5 // 3. Augmentez le score de 10
6
7 // 4. Affichez le score dans la console

```

1.8.2 Exercice 2 : Manipulation du DOM

Étant donné le HTML suivant :

```
1 <h1 id="titre">Titre initial</h1>
2 <p id="texte">Paragraphe initial</p>
```

Écrivez le JavaScript pour : 1. Récupérer le titre 2. Changer son texte en “Nouveau titre” 3. Récupérer le paragraphe 4. Changer son texte en “Nouveau paragraphe”

1.8.3 Exercice 3 : Formulaire interactif

HTML fourni :

```
1 <input type="text" id="prenom" placeholder="Votre prénom">
2 <button id="valider">Valider</button>
3 <p>Bonjour <span id="affichage">...</span></p>
```

Écrivez le JavaScript pour afficher le prénom saisi quand on clique sur “Valider”.

1.9 Erreurs courantes à éviter

1.9.1 Erreur 1 : Oublier const ou let

```
1 // Mauvais (crée une variable globale implicite)
2 compteur = 0;
3
4 // Bon
5 let compteur = 0;
```

1.9.2 Erreur 2 : Utiliser const pour une valeur qui change

```
1 // Erreur
2 const score = 0;
3 score = 10; // TypeError: Assignment to constant variable
4
5 // Correct
6 let score = 0;
7 score = 10;
```

1.9.3 Erreur 3 : Confondre textContent et value

```
1 // Pour un <p>, <h1>, <div>, etc.
2 element.textContent = "texte"; //
3
4 // Pour un <input>, <textarea>
```

```
5 element.value = "texte"; //
```

1.10 Aide-mémoire

```

1 // =====
2 // VARIABLES
3 // =====
4 const PI = 3.14;           // Constante (ne change pas)
5 let compteur = 0;          // Variable (peut changer)
6
7 // =====
8 // RÉCUPÉRER UN ÉLÉMENT
9 // =====
10 const element = document.getElementById('monId');
11
12 // =====
13 // LIRE DES PROPRIÉTÉS
14 // =====
15 element.textContent      // Le texte affiché
16 element.value            // La valeur d'un input
17 element.id               // L'identifiant
18 element.style.color     // La couleur CSS
19
20 // =====
21 // MODIFIER DES PROPRIÉTÉS
22 // =====
23 element.textContent = "Nouveau texte";
24 element.value = "Nouvelle valeur";
25 element.style.color = "red";
26
27 // =====
28 // FONCTION
29 // =====
30 function maFonction() {
31     // Code ici
32 }
33
34 // =====
35 // ÉVÉNEMENT
36 // =====
37 bouton.addEventListener('click', maFonction);
38
39 // =====
40 // OPÉRATEURS COURANTS
41 // =====
42 compteur++                // Ajoute 1
43 compteur += 5              // Ajoute 5

```

```

44 x === y                      // Égalité stricte
45 x !== y                      // Différence stricte
46
47 // =====
48 // CONSOLE
49 // =====
50 console.log(variable);       // Affiche dans la console

```

1.11 Pour aller plus loin

1.11.1 const par défaut

```

1 // BONNE PRATIQUE : const pour les références
2 const bouton = document.getElementById('btn');
3 const affichage = document.getElementById('resultat');
4
5 // Ces éléments ne changeront jamais, même si leur contenu change
6 bouton.textContent = "Nouveau texte"; // OK !
7 affichage.textContent = "Score: 10"; // OK !
8
9 // On ne peut pas réassigner à un autre élément
10 bouton = document.getElementById('autre'); // ERREUR

```

1.11.2 Prochaines notions à découvrir

- Les conditions (`if, else`)
- Les boucles (`for, while`)
- Les tableaux (`array`)
- Les objets
- Les événements avancés
- La manipulation du CSS
- Les timers et animations

1.11.3 Ressources recommandées

- MDN Web Docs (français) : <https://developer.mozilla.org/fr/>
 - JavaScript.info : <https://fr.javascript.info/>
 - W3Schools : <https://www.w3schools.com/js/>
-

Solutions

```
1 // Exercice 1
2 const PI = 3.14159;
3 let score = 0;
4 score += 10;
5 console.log(score);
6
7 // Exercice 2
8 const titre = document.getElementById('titre');
9 titre.textContent = "Nouveau titre";
10
11 const texte = document.getElementById('texte');
12 texte.textContent = "Nouveau paragraphe";
13
14 // Exercice 3
15 const champPrenom = document.getElementById('prenom');
16 const boutonValider = document.getElementById('valider');
17 const affichage = document.getElementById('affichage');
18
19 function afficherPrenom() {
20     const prenom = champPrenom.value;
21     affichage.textContent = prenom;
22 }
23
24 boutonValider.addEventListener('click', afficherPrenom);
```