

Initiation

- Développement Web 420-427-RK
- Par Emeric Depierroz



Plan de la rencontre

Introduction
au JS

Exemples

Laboratoire :
projet



Introduction

Introduction

Définition : javascript

Contrairement à HTML et CSS, JavaScript est un vrai langage de programmation. Plus précisément, il s'agit d'un langage de script, c'est-à-dire qu'il est interprété plutôt que compilé.

Tout comme HTML et CSS, il s'agit d'une technologie employée pour le Web et, le plus souvent en programmation front-end (côté client).

Il s'agit d'un langage orienté objet à prototype (sans classe à l'origine) et à typage dynamique (le type d'une variable est déterminé lors de l'exécution).

Les navigateurs Web savent interpréter ce langage.
HTML.

Avantages

- ▶ Permet de réagir aux événements générés par l'internaute (clic de souris, touche appuyée, chargement d'une page...)
- ▶ Peut rendre une page Web statique plus dynamique, notamment en y ajoutant des interactions avec l'utilisateur, le traitement de l'information et aussi par le biais d'animations.
- ▶ Alors que les échanges de données entre client et serveur impliquent un temps de latence lié aux communications réseau (temps pendant lequel rien ne semble se produire), le JavaScript permet de fournir une rétroaction immédiate à l'internaute.
- ▶ Globalement, il rend l'expérience utilisateur plus vivante, fluide, rapide et ergonomique. Il permet également d'assurer la compatibilité avec différents supports.

Inconvénients

- ▶ Chaque navigateur Web interprète le JavaScript en respectant un standard, mais cela laisse quand même place à des interprétations légèrement différentes. Il faut donc parfois adapter l'écriture du code à des navigateurs spécifiques.
- ▶ Puisqu'il n'est pas conçu pour gérer la sécurité, il lui est interdit d'écrire directement sur le disque dur du client ou encore d'accéder directement à une base de données.
- ▶ L'utilisation de prototypes et la faiblesse du typage sont deux des éléments qui ternissent la réputation du langage. Il est souvent considéré comme faiblement orienté objet.
- ▶ Les erreurs de programmations sont généralement découvertes uniquement lorsque le code est interprété. Le débogage s'avère souvent fastidieux, bien que les outils pour le faire soient aujourd'hui performants.

Ce que permet le JS

- De changer le contenu HTML `.innerHTML()`
 - De modifier des attributs HTML ``
 - De changer des styles `.addClass()`
 - D'afficher ou de cacher des éléments `.hide()` / `.show()`
- Entre autres...

Où placer son JS

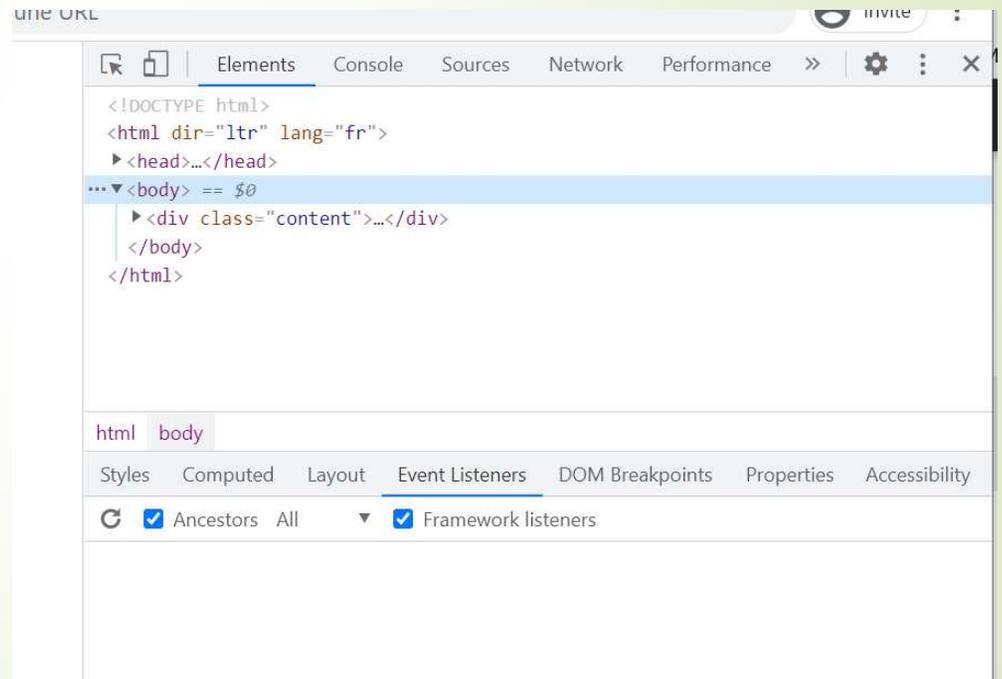
- ▶ Entre deux balises `<script>` dans le `<head>`
- ▶ Entre deux balises `<script>` dans le `<body>`, idéalement à la fin du body : le placer à la fin évite de ralentir le chargement des éléments HTML de la page
- ▶ Sur l'attribut d'évènement d'un élément HTML (button onclick="")

Afficher le résultat (texte simple)

- `Alert()` 
- `Console.log()` 
- `Document.write()` 
- `innerHTML()` 

Console.log()

- Se vérifie dans le débogueur du navigateur (F12)
- Permet de constater les défauts, les erreurs réseau,...



Syntaxe du javascript

- ▶ Le javascript est sensible à la casse : **maj != MAJ**
- ▶ Il n'est pas typé sinon dynamiquement
- ▶ Les mots réservés, que vous ne pouvez pas utiliser pour des variables ou des fonctions, sont

abstract	arguments	await	boolean
break	byte	case	catch
char	class	const	continue
debugger	default	delete	do
double	else	enum	eval
export	extends	false	final
finally	float	for	function
goto	if	implements	import*
in	instanceof	int	interface
let	long	native	new
null	package	private	protected
public	return	short	static
super	switch	synchronized	this
throw	throws	transient	true
try	typeof	var	void
volatile	while	with	yield

Syntaxe du javascript

- ▶ Une variable s'écrit :
 - ▶ Var nom = "Emeric" => *légataire mais encore bien utile*
 - ▶ Let nom = "Emeric" => *nouvelle méthode, non réassignable*
 - ▶ Const nom = "Emeric" => *constante, inamovible*
 - ▶ Nom = "Emeric" => *ça marche mais je déconseille*
- ▶ Elles ne signifient pas toutes la même chose, il est essentiel de bien les utiliser.
- ▶ Les variables ont une portée de bloc délimité par des { }



Syntaxe du javascript

► Une fonction s'écrit

```
Function DisMoiTonNom(){  
  Let nom = "Emeric";  
  console.log(nom)  
}
```

```
onClick = "DisMoiTonNom();"   
console.log(nom) 
```



Les variables instanciées **DANS** la fonction sont utilisables **DANS** la fonction

Syntaxe du javascript

▀ Les commentaires

// sur une ligne

/* sur

Plusieurs

Lignes */



Références

javascript:

<https://www.w3schools.com/js>

<https://developer.mozilla.org/fr/docs/Web/JavaScript>