

Javascript

Modalités



► Philosophie du cours

10H de cours et 10H de TE/TI

Vous êtes vos propres enseignants

- ❖ Pédagogie inversée
- ❖ Favoriser la pratique
- ❖ Savoir-être professionnel
- ❖ Travaux de groupe
- ❖ Actualités

<http://ldfreelance.fr/formation/EPsi/>



► Objectifs du cours

- ❖ Écrire et déboguer du code JavaScript
- ❖ Utiliser les dernières techniques
- ❖ Utiliser la technologie AJAX
- ❖ Connaître une partie des API HTML5
- ❖ Utiliser une librairie
- ❖ Utiliser Node.js
- ❖ Acquérir une vision générale de cette technologie



Plan de cours

18 octobre : Introduction et rappels Table dansantes (canvas)	13 novembre : TE/TI Websocket + exposé
23 octobre : TE/TI Sololearn Javascript	15 novembre : Pratique Classes et système de particules
25 octobre : Pratique AJAX + Leaflet	20 novembre TE/TI (4h) Mini-projet : Système notation workshop
08 novembre : TE/TI HTML5 API	22 novembre : Évaluation Exposés technologies
08 novembre : Pratique Introduction node.js	

Javascript

Introduction et rappels



Présentation

Langage de programmation

1995 Livescript - Netscape
1996 Javascript - Netscape 2 JScrip - IE 3
1997 ECMAScript JS donné à ECMA
1999 ECMAScript 3 Tous les navigateurs
2009 ECMAScript 5

2015 ECMAScript 2015 (ES6)
2016 ECMAScript 2016 (ES7)
2017 ECMAScript 2017 (ES8)
2018 ECMAScript 2018 (ES9)
ESNext...

- ❖ Orienté objet à prototype
- ❖ Normé, [ECMAScript](#)
- ❖ Faiblement typé
- ❖ Non compilé
 - Rapide à produire
 - Plus lent qu'un code machine
 - Sécurité renforcé
- ❖ Compatibilité système, serveur et éditeur

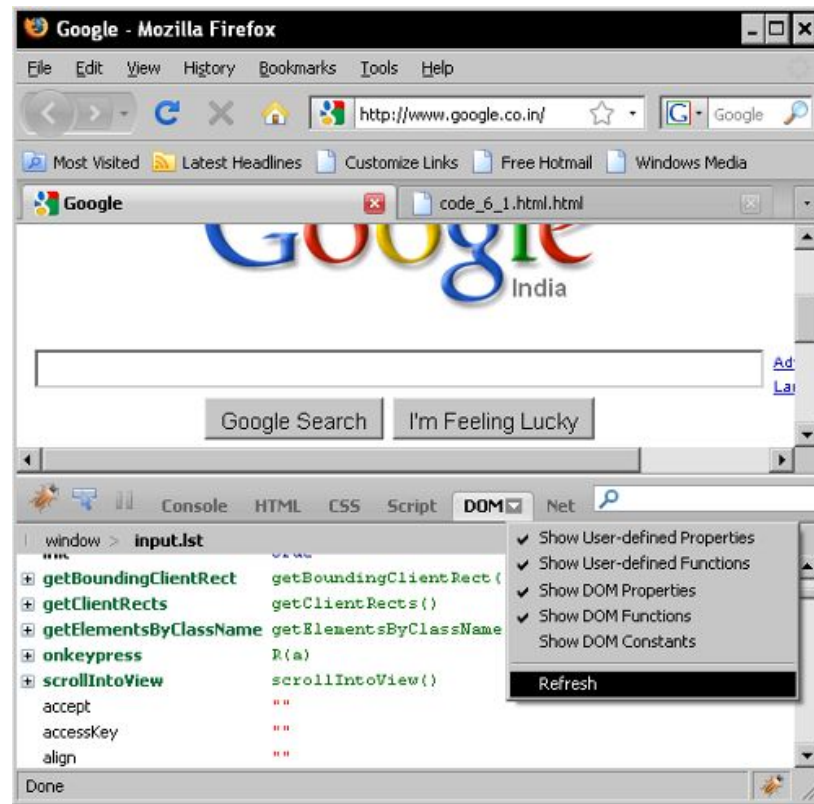
https://www.w3schools.com/js/js_versions.asp

<http://kangax.github.io/compat-table/es5/>

▶ Pour faire quoi ?

- ❖ Interface (interactivité, traitement)
- ❖ Manipulation du DOM
- ❖ Utilisation de bibliothèques
- ❖ Utilisation des API Natives HTML5

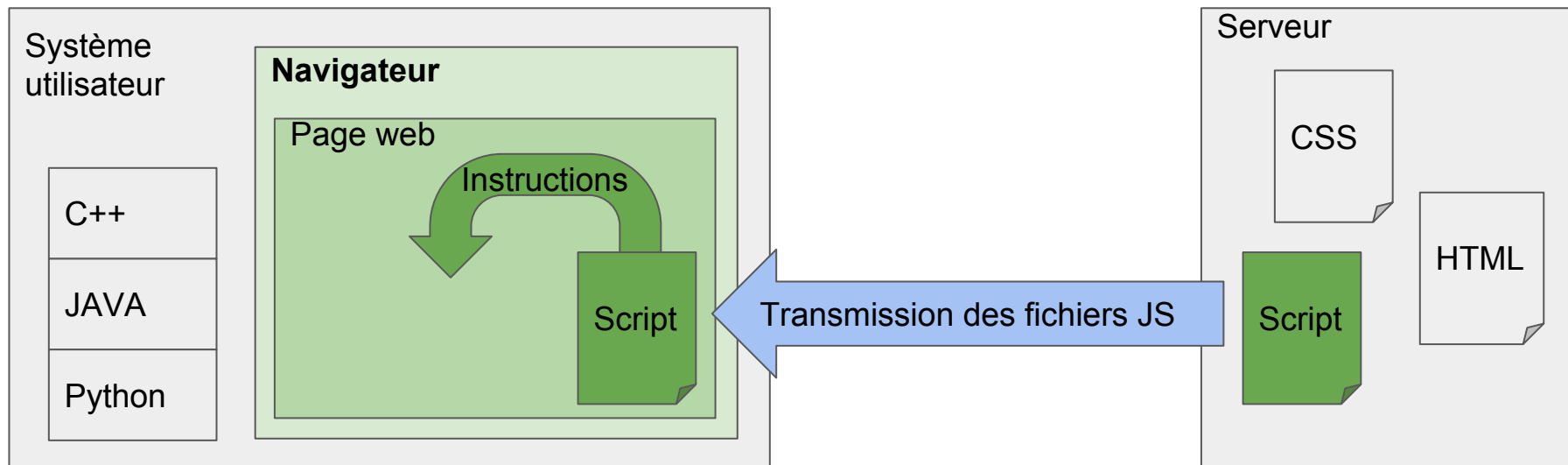
FRONT-END Développement



► Synthèse de fonctionnement

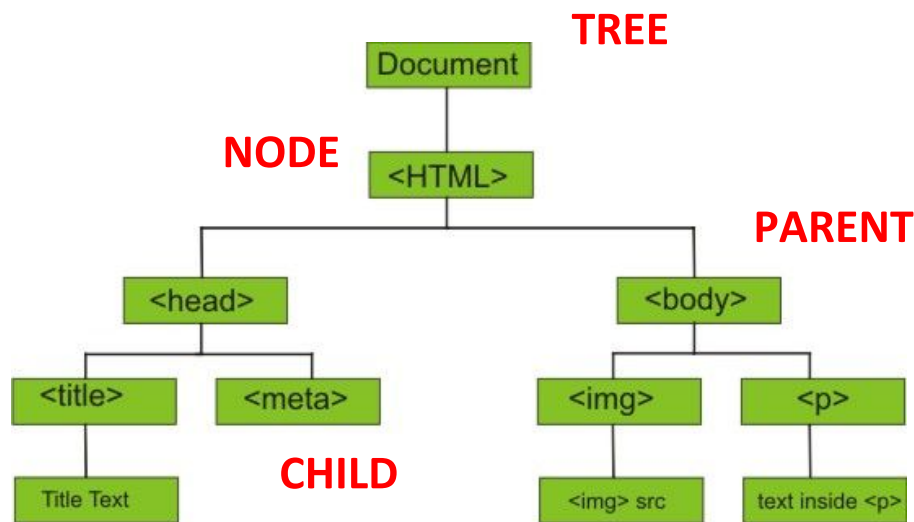
Langage de programmation côté client

Rien à voir avec JAVA



▶ Document Object Model

Le DOM est une convention qui permet de manipuler les éléments d'une page web
Javascript voit une page web comme une collection d'objets



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Simple Page</title>
  </head>
  <body>
    <h1>Une simple page HTML</h1>
    <p>Ceci est une simple page HTML.</p>
    <p>Elle est aussi basique que possible. Elle contient:</p>
    <ul>
      <li>Un titre H1</li>
      <li>Deux paragraphes</li>
      <li>Une liste non ordonnée</li>
    </ul>
  </body>
</html>
```

<https://www.javascripting.com/>

JavaScripting

[Submit a Library](#) | [Follow Us](#) 

The definitive source of the best
JavaScript libraries, frameworks, and plugins.

three.js d3.js leaflet



All Items
Animation
▶ Application Tools
Audio
Development Aids
▶ Dom
Forms
Games
▶ Helpers
▶ Images
Data
Mobile and Touch
Typography
User Interface
Video

799 results

three.js d3.js leaflet

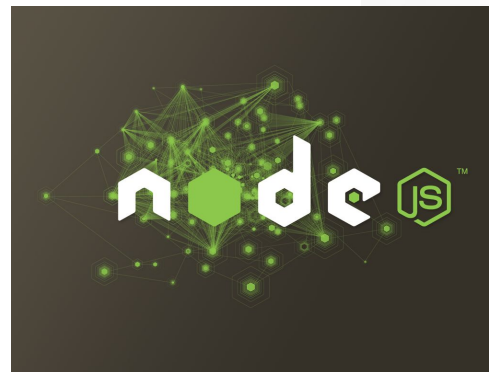
Leaflet	JavaScript library for mobile-friendly interactive maps	97
Three.js	JavaScript 3D library.	99
D3	A JavaScript visualization library for HTML and SVG.	94
D3 Cloud	Create word clouds in JavaScript.	51
Comic.js	Cartoon style drawing for HTML5 Canvas & Raphael.js & D3.js & SVG.js	14
Uv Charts	Simple yet powerful JavaScript Charting library built using d3.js	27
Dc.js	Multi-Dimensional charting built to work natively with crossfilter rendered with d3.js	84
Functional Js	A functional js library that facilitates currying and point-free / tacit programming	23
P5.js	A JS client-side library for creating graphic and interactive experiences, based on the core...	84

► Utilisation uniquement côté client résolue...

En tant que langage (API) dans des outils de renom



Côté serveur



► Tendances

<https://trends.google.fr/trends/?geo=FR> : jQuery ?

<https://www.slideshare.net/bymarkone/the-javascript-toolkit-20>

<https://x-team.com/blog/top-javascript-trends-2018/>



Javascript moderne

Qu'est ce que vanilla js ?

<http://vanilla-js.com/>

Découverte ES5

<https://gist.github.com/sym3tri/2425983>

Approche fonctionnelle : Map, Filter, Reduce

<https://danmartensen.svbtle.com/javascripts-map-reduce-and-filter>

Découverte ES6

<http://es6-features.org/#ExpressionBodies>

Cas fat arrow

<http://kangax.github.io/compat-table/>

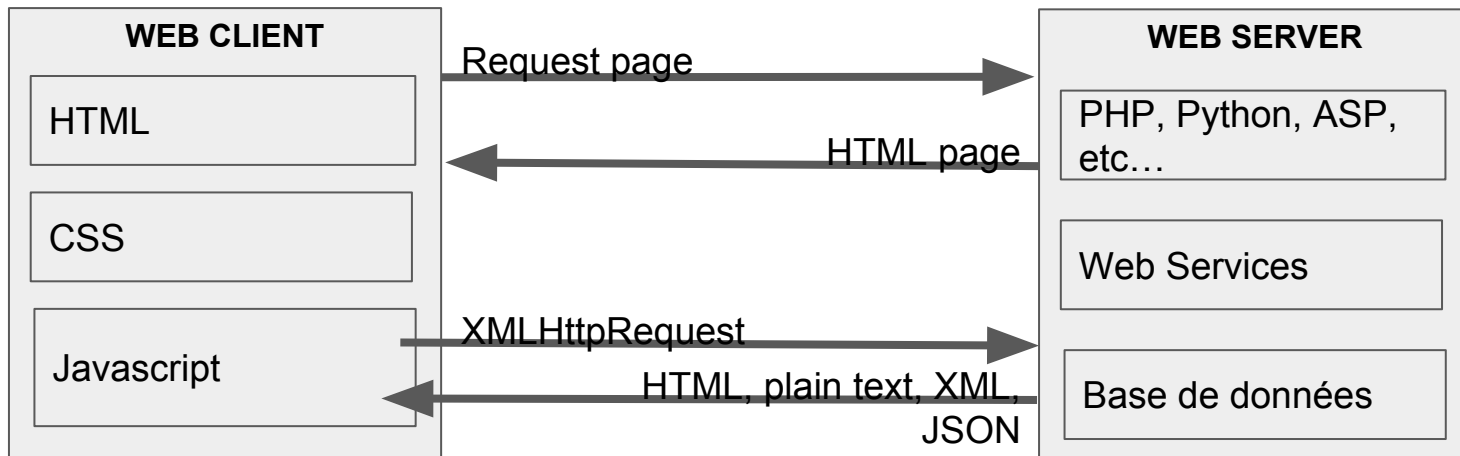
Cas de l'égalité ? == versus ===

Cas du let et const ?

<https://caniuse.com/>

▶ GET, POST, AJAX

- ❖ GET et POST sont les deux moyens classique pour envoyer une requête
- ❖ AJAX est une technologie qui permet de manipuler des objet XMLHttpRequest :
 - Pas de rechargement de page



Ressources conseillées

[MDN javascript](#)

[W3schools](#)

[Tutorialspoint](#)

[Sololearn JavaScript](#)

Rappels

Mise en pratique



Rappels

Les variables (5 principaux types)

Assignation

[Valeur initiale](#)

Affichage (4 méthodes)

Conditions

Tableaux

Boucles

Objets Date et Math

Les fonctions

Gestion du temps

Manipulation du DOM

Objets JavaScript

Évènements

JSON

Canvas et SVG

<https://htmlcheatsheet.com/js/>

<http://overapi.com/javascript>

SensitiveCase

▶ Canvas

Javascript pour dessiner un rectangle

```
var canvas = document.getElementById("myCanvas");
```

```
var ctx = canvas.getContext("2d");
```

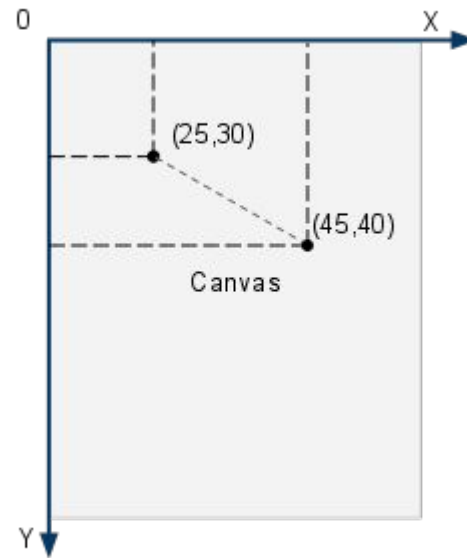
```
ctx.fillStyle = "#FF0000";
```

```
//ctx.fillRect(x, y, width, height);
```

```
ctx.fillRect(25, 30, 20, 10); // trace où ?
```

La balise HTML indispensable

```
<canvas id="myCanvas" width="500px" height="500px"></canvas>
```



<http://cheatsheetworld.com/programming/html5-canvas-cheat-sheet/>

► Dessiner

```
var pressed = false;

function mousedown(event) {pressed = true;}

function mouseup(event) {pressed = false;}

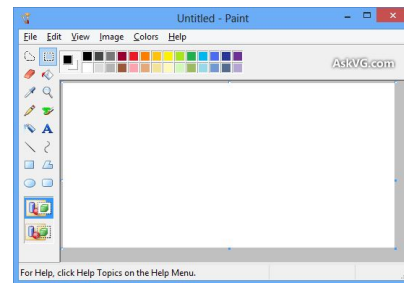
function mousemove(event) {
    event.preventDefault();
    var x = event.pageX - c.offsetLeft;
    var y = event.pageY - c.offsetTop;
    if(pressed) draw(x, y);
}

function draw(x, y) {
    ctx.fillStyle = "#000000";
    ctx.fillRect(x-1, y-1, 3, 3);
}
```

```
var c, ctx;

function init() {
    c = document.getElementById("myCanvas");
    ctx = canvas.getContext("2d");
    c.addEventListener("mousedown", mousedown);
    c.addEventListener("mouseup", mouseup);
    c.addEventListener("mousemove", mousemove);
}

// que manque t il ???
```



▶ Représentation graphique de la multiplication

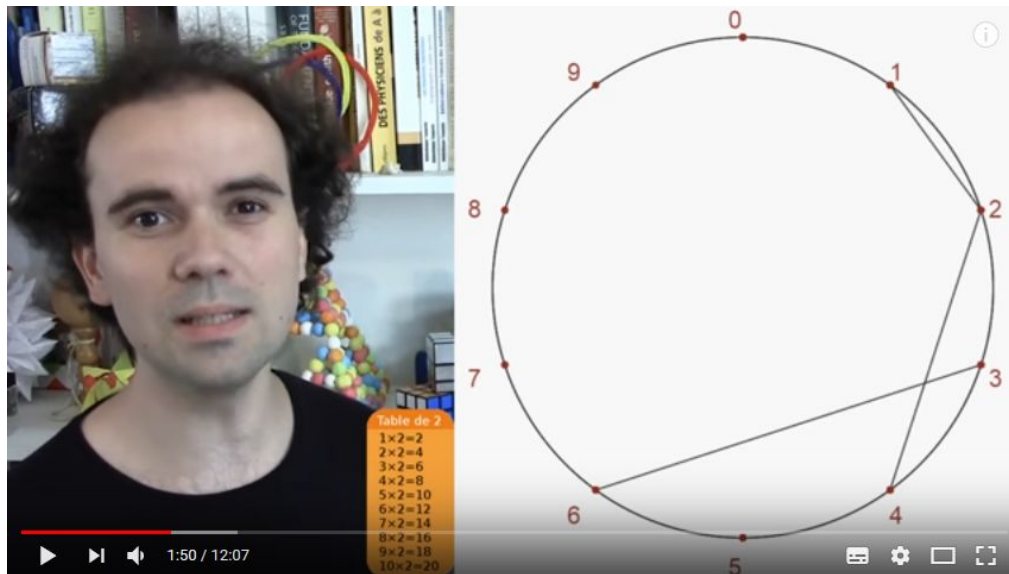
$$1 * 2 = 2$$

$$2 * 2 = 4$$

$$3 * 2 = 6$$

...

$$i * \text{table} = ?$$



$$1 * 2 \% 10 = 2$$

$$2 * 2 \% 10 = 4$$

$$3 * 2 \% 10 = 6$$

...

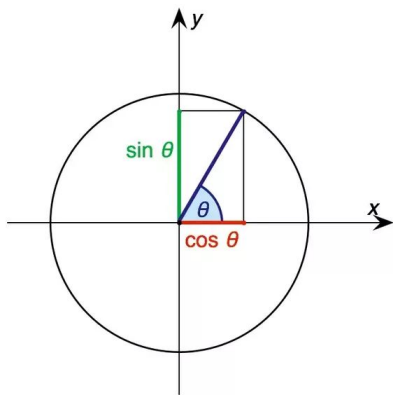
$$i * \text{table} \% \text{nbpoints} = ?$$

► Calcul de position

Comment trouver des **coordonnées** (x,y) pour un point donné sur un **cercle** ?

```
// ceci est un ?  
let step = 2 * PI / nbPoints;
```

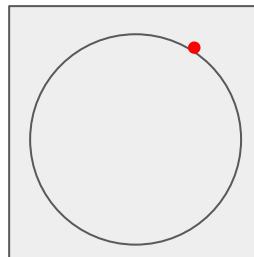
```
// coordonnées  
let x = cos(i * step);  
let y = sin(i * step);
```



Comment **positionner** un point par rapport à un **point central** ?

```
// ceci est un ?  
let step = 2 * PI / points;
```

```
// coordonnées  
x = centreX + rayon * cos(i * step);  
y = centreY + rayon * sin(i * step);
```



Démonstration du résultat attendu

Question de groupe

Qu'est ce qu'une closure ?

A quoi cela est il utile ?