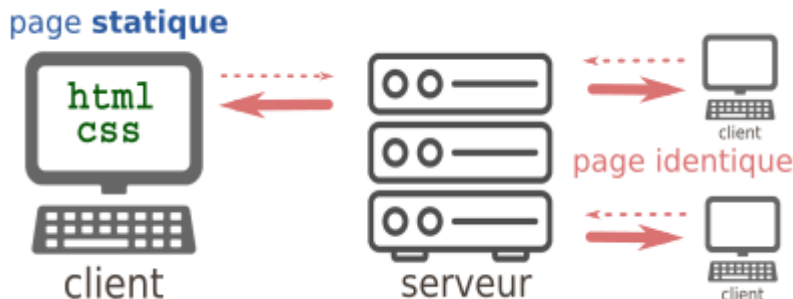


IHM. Interactions avec le Web (Client)

I/ Sites statiques

Il s'agit du cas le plus simple comme présenté ci-dessous :



Les pages de Wikipédia sont un exemple de ce type de sites où le client ne peut interagir sur les pages.

Lorsque le client demande au serveur le contenu d'une page web, celui-ci lui renvoie dans ce cas une simple page html.

HTML est un langage dit « à balises ». Ce n'est pas à proprement parler un langage de programmation, mais plutôt un langage de description de contenu.

Voici un exemple de page web minimal et le résultat à côté :

```
<!DOCTYPE html>
<html lang='fr'>
  <head>
    <meta charset="utf-8">
    <title>Un titre très original</title>
  </head>
  <body>
    <p>Ceci est le texte introductif de ma page.</p>
    <p>
      <h1> Ceci est un titre de niveau 1 </h1>
      Mais sans rien d'intéressant.
      <h2> Ceci est un titre de niveau 2 </h2>
      <ul>
        <li> le début d'une liste indentée </li>
        <li> la suite ... </li>
      </ul>
      Pour apprendre le fonctionnement des balises, voir <a href="https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Getting_started">ici</a> par exemple !
    </p>
  </body>
</html>
```

Ceci est le texte introductif de ma page.

Ceci est un titre de niveau 1

Mais sans rien d'intéressant.

Ceci est un titre de niveau 2

- le début d'une liste indentée
- la suite ...

Pour apprendre le fonctionnement des balises, voir [ici](https://developer.mozilla.org/fr/docs/Apprendre/HTML/Introduction_%C3%A0_HTML/Getting_started) par exemple !

On remarque que le résultat n'est pas très esthétique, c'est justement le rôle du CSS !

L'acronyme **CSS** signifie *Cascading Style Sheets*. L'idée est de regrouper dans un seul fichier toutes les informations relatives à la mise en forme des éléments de la page html. De manière très simplifiée, on peut dire que le fichier **html s'occupe du fond** tandis que le fichier **css s'occupe de la forme**.

Le fichier CSS (souvent nommé style.css) doit être référencé au début du fichier html , au sein de la balise <head>.

Voici un exemple de page web minimal combinant html et css et le résultat en bas à droite :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>page test</title>
    <link href="style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
    <p>
      <h1> Ceci est un titre de niveau 1 </h1>
      Mais sans rien d'intéressant.
      <h2> Ceci est un titre de niveau 2 </h2>
    </p>
  </body>
</html>
```

```
1 html {
2   font-size: 15px;
3   font-family: sans-serif;
4   background-color: lightgray;
5 }
6
7 h1 {
8   color: red;
9 }
```

Ceci est un titre de niveau 1

Mais sans rien d'intéressant.

Ceci est un titre de niveau 2

En savoir plus :

Le site suivant (anciennement site du zéro) :

<https://openclassrooms.com/fr/courses/1603881-apprenez-a-creeer-votre-site-web-avec-html5-et-css3>

Attention : ce site nécessite une inscription et utilise l'éditeur de texte *Sublime Text* téléchargeable [ici](#) et gratuit. On peut toutefois utiliser Notepad++.

Le tutoriel vidéo très complet de Pierre Giraud : <https://www.youtube.com/watch?v=8FqZZrbnwkM> en 3 parties
Mots clefs : « Pierre Giraud html css ». Durées : 1h23, 1h15 et 1h42

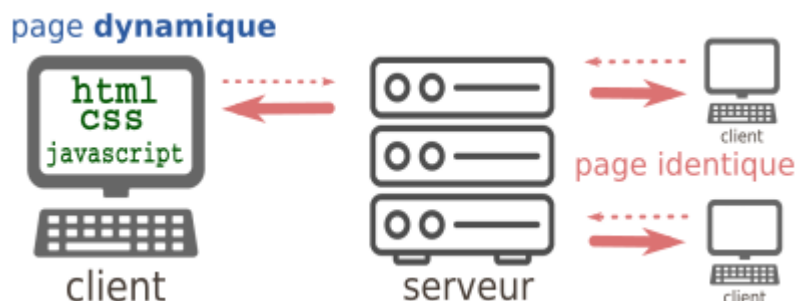
II/ Sites dynamiques

Ce type de sites, plus avancés, s'opposent aux statiques précédents. Ils permettent donc de proposer des pages personnalisées en fonction des interactions du client avec le site ou de ses demandes.

Il y a deux types d'interactions :

- Dans le premier cas, il s'agit par exemple d'un clic sur un bouton, d'un prénom entré par un client. Ce type d'actions ne nécessitent pas l'intervention d'un serveur : ce sont des **interactions client – utilisateur**.
- Dans le second cas, il s'agit par exemple de consulter son compte bancaire, de commander un type de vêtements : ce type d'actions nécessitent la consultation d'un serveur : ce sont des interactions **client – serveur**.

On s'intéressera dans un premier temps aux interactions client – utilisateur.



Inventé en 1995 par Brendan Eich pour le navigateur Netscape, le langage **JavaScript** s'est imposé comme la norme auprès de tous les navigateurs pour apporter de l'interactivité aux pages web après des débuts assez chaotiques : en effet, ce langage peut être utilisé abusivement notamment dans l'apparition de pop-ups (apparition de petites fenêtres non demandées et dérangeantes) à tel point que les navigateurs permettent de bloquer Javascript !)

Voici un exemple de page web minimal combinant html et javascript et le résultat en bas à droite :

HTML ▼

```
1 <!doctype html>
2 <html lang="fr">
3 <head>
4   <meta charset="utf-8">
5   <title>Le trio</title>
6   <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9   <h1>Le trio : HTML, CSS et JavaScript</h1>
10  <p>Voici une page web qui ne fait pas grand chose</p>
11  <button onclick="maFonction()">Cliquez ici</button>
12 </body>
13 <script src="script.js"></script>
14 </html>
```

JavaScript + No-Library (pure JS) ▼

```
1 function maFonction() {
2   alert("Le JavaScript fonctionne !")
3 }
```

CSS ▼

```
1 html {
2   font-size: 15px;
3   font-family: sans-serif;
4   background-color: lightgray;
5 }
6
7 h1 {
8   color: red;
9 }
```

Le trio : HTML, CSS et JavaScript

Voici une page web qui ne fait pas grand chose

Cliquez ici

Lorsque l'on appuie sur le bouton « Cliquez ici », il apparaît le pop-up à droite :

Une page intégrée à l'adresse fiddle.jshell.net indique
Le JavaScript fonctionne !

OK

En savoir plus :

Le site suivant (anciennement site du zéro) :

<https://openclassrooms.com/fr/courses/6175841-apprenez-a-programmer-avec-javas>

Le tutoriel vidéo très complet de Pierre Giraud : <https://www.youtube.com/playlist?list=PLwLsbqvBllmFB8AuT6ENlg-s87ys4yGWI> , 65 petites vidéos.

Mots clefs : « Pierre Giraud javascript ». Durées : variable en fonction des vidéos.