

PHP et javascripts sont à placer avant HTTP et HTTPS. Pas au bon endroit à cause des activités sur ordinateurs et je n'avais pas accès au salles.

## Le Web.

### I Introduction et vocabulaire.

Un *hypertexte* est un texte augmenté de renvois automatiques à d'autres textes, des images ou des sons, un autre paragraphe dans le même texte.

Les renvois à d'autres documents se font grâce aux liens hypertextes. Un *lien hypertexte* (ou hyperlien, ou lien) est une référence dans un texte (ou sur une image) permettant de passer du document consulté à un autre document situé ailleurs sur un autre ordinateur.

Le *Web* (ou *toile* ou *réseau*) désigne le système donnant accès à un ensemble de données (page, image, son, vidéo) reliées par des *liens hypertextes* et accessibles sur le réseau Internet.

Tandis qu'Internet correspond à la connexion des machines entre-elles et les règles d'échange d'informations entre ordinateurs, le Web correspond à la connexion des contenus entre eux au moyen de liens hypertextes. Internet est formé des liens entre les machines tandis que le web est formé des liens entre les sites web.

Il ne faut donc pas parler des sites Internet mais bien des sites Web.

Schéma avec ordinateurs puis schéma avec sites web.

Pour que les hyperliens puissent fonctionner il nécessaire que chaque document ait une référence, un emplacement, une adresse unique qui permet de le retrouver. Cet emplacement unique est appelé une URL.

Une *URL* (Uniform Resource Locator) n'est pas simplement l'adresse d'un site mais l'adresse très précise d'un document sur ce site internet. Si l'adresse du sous-domaine français de wikipedia est <https://fr.wikipedia.org> l'URL de la page qui parle des URL est [https://fr.wikipedia.org/wiki/Uniform\\_Resource\\_Locator](https://fr.wikipedia.org/wiki/Uniform_Resource_Locator). Remarquons que les URL ne se limitent pas aux seuls sites Web et fonctionnent pour des adresses mail, les serveurs FTP, et d'autres encore.

Et que faut-t-il utiliser pour retrouver l'ordinateur sur lequel est installé le site internet correspondant à une URL ?

Si chacun choisi son alphabet il devient impossible de partager des textes. De même sur internet pour que tout les ordinateurs puisse le lire il faut des règles communes de communication, d'affichage.

Certaines normes furent décidées pour que les hypertextes fonctionnent correctement sur tous les ordinateurs.

- Trouver le document hypertexte : un adressage unique de chaque page internet appelé l'URL.
- Les règles d'écriture du document hypertexte : un langage informatique pour écrire le texte qui inclus des informations de mise en page et les indications de lien appelé le langage HTML.
- Des règles de mise en page du document hypertexte : un ensemble de réglages pour la présentation (couleur et polices) appelé le langage CSS.
- Le protocole, la façon, dont un hypertexte est communiqué d'un ordinateur à un autre : un ensemble de règles pour envoyer le contenu du page web à un autre ordinateur appelé protocole HTTP.
- Le logiciel pour afficher l'hypertexte : ils sont appelés navigateurs web (firefox, chrome, opéra,).

## II Les URL.

### Une adresse locale.

Il est possible avec Firefox d'afficher une adresse locale sur l'ordinateur par exemple sur mon ordinateur l'URL `file:///home/pi/unemainlavelautre/` affiche le contenu de mon dossier appelé « unemainlavelautre ».

Nous retrouvons bien toute l'arborescence de l'ordinateur : les différents répertoires et la façon dont ils sont rangés les uns dans les autres.

Comparer l'arborescence sur firefox et pcmanfm.  
Les représenter sous forme d'arbre probabiliste.

## Une adresse sur internet.

Voici un exemple d'URL :

`http` `://` `unemainlavelautre.net` `/` `site_exemple/page_exemple.html`

que nous pouvons déchiffrer de la façon suivante :

<code>http</code>	nom du protocole utilisé
<code>://</code>	séparateur
<code>unemainlavelautre.net</code>	nom de domaine
<code>unemainlavelautre</code>	nom du site
<code>net</code>	extension de domaine
<code>/</code>	séparateur
<code>site_exemple/page_exemple.html</code>	chemin local dans le répertoire du site web
<code>site_exemple</code>	nom du dossier où est rangé la page
<code>/</code>	séparateur
<code>page_exemple.html</code>	fichier au format html dans lequel est l'hypertexte

Montrer ce qu'il se passe sur mon ordinateur. Cliquer sur le lien mais aussi naviguer dans l'arborescence des répertoires pour mettre en évidence le rangement du site.

## Exercices.

### Exercice 1.

*Promenade dans les dossiers du serveur.*

1. Allez jusqu'à l'URL : `http://dpernoux.com/Cours/Cours_maths.htm`
2. Dans la barre d'adresse supprimez la fin de l'adresse et ne laissez que `http://dpernoux.com/Cours`, puis appuyez sur la touche entrée.
3. Cliquez sur le dossier « Cours » et expliquez ce que vous voyez.

Sur la plupart des sites web cette navigation sur le site est interdite par son créateur.

### Exercice 2.

*Le vrai du faux.*

1. Cliquez sur le lien suivant <https://lycee-levavasseur.re/voyage-a-madrid/>
2. Après avoir ouvert un nouvel onglet dans votre navigateur, rendez-vous sur le site web du lycée dont l'URL est : <https://lycee-levavasseur.re/>. Dans l'onglet « Vie des élèves » choisissez « Voyages scolaires » et enfin voyage à Madrid.
3. Comparez les deux pages obtenues aux questions 1 et 2 (jeux des sept différences).
4. Quels problèmes de sécurité cela concerne-t-il et comment y remédier ?

### Correction exercice 2

1. La principale façon d'identifier la fausse page est de déchiffrer l'URL. Il est également possible d'identifier des fautes d'orthographe ou des icônes qui s'affichent mal.

La méthode de fraude présentée ci-dessus s'appelle du *phishing* ou du *hameçonnage* en français. Elle consiste à proposer des liens qui mènent vers de faux sites qui propageront des virus ou plus classiquement essaieront de vous voler de l'argent en vous faisant acheter des produits que vous ne recevrez jamais.

Il existe diverses méthodes pour parer à ce genre de fraude.

- Plutôt que d'utiliser des liens allez directement à la page d'accueil du site et naviguez ensuite dans celui-ci.
- Vérifiez l'URL du site que vous consultez.
- Vérifiez les fautes d'orthographe et plus encore de grammaire.
- Préférez des connexion `https`.

### Exercice 3.

L'arborescence d'un site désigne l'arborescence des répertoires qui sont sur l'ordinateur qui sert de serveur web et que l'on peut atteindre avec un navigateur.

1. Ouvrez le navigateur Firefox allez sur le moteur de recherche Google et entrez « `site:https://lycee-levavasseur.re/` » (aucun espace).
2. Essayez de retrouver le début l'arborescence du site du lycée et présentez-la sous forme d'un arbre (comme ceux de probabilité) sur deux niveaux (n'allez pas plus profondément).

### Exercice 4.

Faites des recherches sur Tim Bernes-Lee et expliquez son rôle dans le développement de ce qui nous intéresse dans cette leçon.

### III Le langage HTML.

#### Le principe des balises.

Le langage HTML est formé de l'ensemble des règles qui permettent la mise en page d'un texte sur un site web.

Comme Python, le langage HTML est un langage textuel. Il est donc possible de l'écrire dans n'importe quel fichier texte d'un ordinateur en lui ajoutant l'extension « .html ».

Il fonctionne sur le principe des *balises* qui encadrent le texte. Ainsi pour que le texte « Hello world » soit écrit en gras il faut l'insérer entre les balises comme suit :

```
<b>Hello world</b>
```

Il y a une balise de début, `<b>`, et une balise de fin, `</b>`. La balise de fin est la même que celle de début mais précédé du slash (/).

#### Ouvrir une page HTML.

Nous utiliserons en général deux logiciels pour ouvrir nos pages HTML.

- Le logiciel du client (dans l'architecture client-serveur) qui est le navigateur web avec lequel on se promène sur internet : Firefox, Chrome, etc.
- Le logiciel du serveur, le logiciel pour créer la page HTML qui est un simple éditeur de texte (ou un peu plus sophistiqué) : bloc-note, notepad, etc.

#### Un exemple minimaliste de page HTML.

Une page HTML requière certains éléments incontournables.

Voici une page web dans un éditeur de texte.

```

<!DOCTYPE html>

<html>

<head>
<title>premiere_page</title>
<meta charset="UTF-8">
</head>

<body>

<!-- Un commentaire qui ne s'affichera pas -->

<h1>Ici un titre.</h1>

<p>
Paragraphe pour présenter un lien hypertexte:<br>
<a href="http://unemainlavelautre.net">Ici les mots qui seront un lien
Et une image:<br>

</p>

</body>

</html>

```

Cliquez [ici](#) pour voir la page dans un navigateur web.

Expliquons un peu ces différents éléments.

- Contrairement à Python les indentations n'ont aucune obligation en HTML. Mais il d'usage de les utiliser pour faciliter la lisibilité du document.
- `<!DOCTYPE html>` indique la version du langage HTML utilisé : ici la dernière version c'est-à-dire le HTML 5.
- Les balises `<html>` et `</html>` délimitent l'ensemble du contenu HTML.
- Les balises `<head>` `</head>` délimitent des informations sur la page qui ne seront pas affichées : les mots clefs utiles pour les moteurs de recherche, les styles d'écriture personnels choisis pour tout le document.

Les balises `<title>` `</title>` indiquent le titre qui sera affiché sur l'onglet du navigateur.

`<meta charset="UTF-8">` indique qu'il faut prendre en compte l'alphabet français avec des accents.

- Les balises `<body>` et `</body>` délimitent le corps du document c'est-à-dire ce qui sera affiché dans le navigateur.
- Les balises `<h1>` et `</h1>` indiquent un titre. Les sous-titres sont `<h2></h2>`, les sous-sous-titres `<h3></h3>...`  
Les balises `<p>` et `</p>` délimitent un paragraphe de texte.
- La balise `<br>` est seule elle indique un passage à la ligne.
- La balise `` est elle aussi seule. Elle permet d'afficher sur la page une image, ici "tux.webp", placée dans le même dossier que la page HTML.
- Pour insérer un lien hypertexte la balise doit évidemment contenir l'URL vers laquelle il va pointer : `<a href="URL">` et `</a>`.
- Pour écrire facilement les lettres françaises accentuées il est possible d'indiquer la balise unique `<meta charset="utf-8">` à l'intérieur du balisage `<head>` et `</head>`.

Il existe de nombreuses autres balises HTML qu'il est possible de trouver en cherchant sur le web je vous recommande tout particulièrement : [w3schools.com](http://w3schools.com).

## Exercices.

### Exercice 5.

Nous allons maintenant plagier un site web (faire un "Fake").

1. Dans votre répertoire personnel créez un dossier *web* puis dans ce répertoire créez un dossier *faux\_site*.
2. Rendez vous sur l'URL <https://mathoprof.jimdo.com/> et copiez-en le contenu (enregistrer sous ou ctrl+s) dans le répertoire « web/faux\_site ».
3. Trouvez, puis ouvrez avec notepad le fichier HTML que vous avez téléchargé. Recherchez (ctrl+f) le titre du site Mathoprof et modifiez-le comme bon vous semble.
4. Vérifiez que votre modification fonctionne en ouvrant maintenant la page HTML modifiée avec un navigateur web (double-clic sur le fichier HTML devrait suffire).

## Exercice 6.

Créer une page personnalisée.

1. Allez dans le dossier *web* et créez un dossier dans votre répertoire appelé *un\_site\_web*.
2. Suivez le lien [http://unemainlavelautre.net/site\\_exemple/exemple.zip](http://unemainlavelautre.net/site_exemple/exemple.zip) puis téléchargez l'archive et extrayez-la.
3. Ouvrez le logiciel notepad (ou bloc-notes). Puis ouvrez la page HTML *page\_exemple.html* téléchargée à la question précédente.
4. Double-cliquez sur votre fichier *page\_exemple.html* pour l'ouvrir avec un navigateur.

*Si le fichier ne s'ouvre pas normalement demandez l'aide de l'enseignant.*

5. Modifiez, dans notepad, la page : titre, texte, image et lien hypertexte. Vérifiez que cela fonctionne dans le navigateur web en actualisant régulièrement la page (F5).

La page d'accueil d'un site internet s'appelle toujours « *index.html* ». Ce nom lui confère un statut spécial. Elle est affichée par défaut pour une requête sur le serveur HTTP même si aucune URL n'est spécifiée.

## IV Publier sa page HTML sur internet.

Installer un serveur apache le configure l'activer, le rendre accessible.

## V Les CSS.

Le style graphique d'une page (polices utilisées, tailles, couleurs) peut être précisé par un *fichier de style* (en anglais stylesheet) au format CSS. Un fichier CSS (cascading stylesheet) contient des règles de mise en forme.

Par exemple la règle « `h1{color :blue;}` » indique que tous les titres de niveau 1 doivent être affichés en bleu. Il est inutile de préciser sur chaque ligne de la page HTML la présentation qu'il faudra choisir.

Il est ainsi possible de personnaliser chaque bloc (entre des balises).

Il est également possible de créer une classe : `.maclasse{color:pink;}`. C'est une personnalisation qui peut s'appliquer à n'importe quel bloc de balises. Typiquement : `<div class="maclasse"> </div>`. Tout le texte entre ces balise sera en rose.

Les instructions CSS peuvent être indiquées entre les balises `<head></head>` de la page HTML mais le plus souvent elles sont dans un autre fichier qui prend le



nom d'extension « .css » et qui est placé à côté du fichier HTML.

L'usage des fichiers CSS contribue à homogénéiser la présentation d'une page web (et même d'un site) et à séparer le fond, c'est-à-dire le contenu (le HTML), de la forme (le CSS). De plus si l'on souhaite changer la présentation de toutes les pages du site il suffit de changer les CSS. Par exemple ce site propose de tester différentes feuille de style : <http://www.csszengarden.com/>.

Voici un exemple de CSS.

```
/*Personalise la balise html */

html {
  background-color:dodgerblue;
  font-family:"Open Sans",sans-serif;
  padding: 5px 25px;
  font-size: 18px;
  margin:0;
  color:#400;
}

/*Personnalise les liens hypertexte */

a {
  color:red;
}

/*Personnalise les titres */

h1 {
  font-family:"Merriweather",serif;
  font-size:48px;
  text-align: center;
  margin: auto;
}

/* Création d'une classe. Une personnalisation qui
peut s'appliquer à n'imorte quel bloc: div, body,
h1... */

.blockpersonel{
  max-width: 800px;
  margin: auto;
  background-color: rgba(255,255,255,1);
}
}
```

Cliquez [ici](#) pour voir la page dans un navigateur web.

Cliquez [ici](#) pour voir l'importance des CSS dans l'aspect d'un site.

## Exercice 7.

Personnaliser la feuille de style.

1. Allez dans le dossier web et créez un dossier appelé *site\_web\_personnalise*.
2. Téléchargez et enregistrez dans le dossier *site\_web\_personnalise* la page suivante : [http://unemainlavelautre.net/site\\_exemple/exemple\\_avec\\_css.zip](http://unemainlavelautre.net/site_exemple/exemple_avec_css.zip). Puis extrayez l'archive.
3. Ouvrez la page *page\_exemple\_avec\_css.html* avec un navigateur web. Vous pourrez remarquer une mise en page différente pour la page de l'exercice 6.
4. Avec un éditeur de texte comme notepad ouvrez les fichiers *page\_exemple\_avec\_css.html* et *css\_exemple.css*.
5. Faites les modifications suivantes de mise en page (essentiellement dans le fichier CSS). Vous vous aiderez des instructions déjà présentes dans le CSS et vous vérifierez que vos modifications fonctionnent en actualisant la page dans le navigateur web.
  - (a) Changez pour du vert la couleur d'arrière plan de la page toute entière (balise *html*).  
Pour choisir une couleur vous pouvez utiliser le rgb ou : [https://www.w3schools.com/cssref/css\\_colors.asp](https://www.w3schools.com/cssref/css_colors.asp).
  - (b) Modifiez la police (*font* en anglais) du titre pour quelque chose de plus excentrique.  
Pour changer de police : [https://www.w3schools.com/Css/css\\_font.asp](https://www.w3schools.com/Css/css_font.asp).
  - (c) Créez une classe *.paragraphe* que vous appliquerez à la balise `<p>` et telle que :
    - la police à une taille de 24 pixel,
    - l'arrière plan est orange,
    - la largeur maximale du paragraphe est de 600 pixels,
    - le paragraphe est au centre de la page car les marges sont réglées automatiquement.

## VI HTTP et HTTPS.

### HTTP.

Les échanges d'informations entre ordinateurs sur internet se font suivant un protocole (ensemble de règle de communication) appelé TCP.

Via ce protocole TCP deux ordinateurs peuvent échanger des informations. Ainsi l'ordinateur hébergeant un site web et l'ordinateur d'un internaute peuvent se mettre en relation. Dans ce cas particuliers les règles de communications entre ces deux ordinateurs sont appelées HTTP.

1. Le client (l'ordinateur de l'internaute) demande une URL précise,

2. le serveur envoie à l'ordinateur de l'internaute le HTML, le CSS et toutes les images ou vidéos liées à l'URL demandée,
3. le client reçoit toutes les informations et les transmet au navigateur qui les affiche sur l'écran.

Voici un exemple pour les deux premières étapes.

1. Le client envoie au serveur la requête.

```
GET /index.html HTTP/1.1
Host : unemainlavelautre.net
```

2. Si la page est introuvable le serveur répond

En ligne de commande :

```
GET /index.html HTTP/1.1 Host : unemainlavelautre.net/test.html
```

```
HTTP/1.1 404 Not Found
```

Sinon le serveur répond

En ligne de commande :

```
GET /index.html HTTP/1.1 Host : unemainlavelautre.net
```

```
HTTP/1.1 200 OK
Date : Mon, 4 Mar 2019 18 :24 :23 GMT
Content-Type : text/html ; charset=UTF8
Content-Length :983
Last-Modified : Wed, 8 Jan 2019 05 :10 :35
```

```
<html>
<head> ...
```

### Exercice 8.

Le client envoie au serveur la requête.

```
GET /index.html HTTP/1.1
Host : unemainlavelautre.net
```

Si la page est introuvable le serveur répond

```
HTTP/1.1 404 Not Found
```

Sinon le serveur répond

```
HTTP/1.1 200 OK
Date : Mon, 4 Mar 2019 18 :24 :23 GMT
Content-Type : text/html; charset=UTF8
Content-Length :983
Last-Modified : Wed, 8 Jan 2019 05 :10 :35

<html>
<head> ...
```

Expliquez le rôle des différentes instructions dans cet exemple de requêtes HTTP.

### Exercice 9.

Lorsque le client et le serveur communiquent ils échangent des informations sur ce qu'ils sont.

Suivez [ce lien](#) et indiquez qu'elle sont les informations que vous communiquez au site internet sur lequel vous souhaitez vous connecter.

### Le cache.

Pour accélérer la navigation les navigateurs enregistrent (temporairement) les pages visitées sur l'ordinateur du client. On dit alors que les pages sont *mises en cache*. C'est ce qui permet d'utiliser les boutons précédents et suivants du navigateur. Les pages mises en cache constituent des traces de notre navigation.

L'ensemble des toutes les pages mises en cache constituent ce qu'on appelle l'historique de navigation qui est le plus souvent consultable sur le navigateur.

### Exercice 10.

Allez sur votre navigateur internet et identifiez dans les préférences les réglages possibles concernant l'historique de navigation.

Notez les différents réglages possibles.

## Les cookies.

Le protocole HTTP prévoit que le site web puisse enregistrer des informations du client sur l'ordinateur du client dans de petits fichiers textes appelés les *cookies*. Ainsi les préférences de présentation, d'affichage choisis par le client, sont renvoyés dans un cookie par le serveur à chaque fois qu'il se connecte au serveur.

Les cookies peuvent servir à enregistrer toutes les démarches du clients sur le serveur.

De nos jours les cookies sont utilisés pour garantir des sessions d'utilisation de client sur les sites commerciaux. Lorsque l'utilisateur du site donne ses identifiants et mots de passe un cookie est créé sur son ordinateur avec un mot de passe temporaire pour identifier l'utilisateur pendant toute sa visite du site.

Les cookies ont une très mauvaise réputation car ils ont autrefois servis à enregistrer beaucoup d'informations sur les clients et conservaient aussi des mots de passe et des informations personnelles.

### Exercice 11.

Allez sur votre navigateur internet et identifiez dans les préférences les réglages possibles concernant les cookies.

Notez les différents réglages possibles.

## HTTPS.

À l'heure du commerce sur internet (et donc des échanges d'argent) il est devenu nécessaire de sécuriser davantage les communications HTTP. Il s'agit notamment de protéger le client de deux types d'attaques classiques : le phishing et le man-in-the-middle.

Faire des schéma pour les attaques et pour le HTTPS : communication entre client et serveur demande de carte d'identité (certificat) (contre le phishing), et pour que personne ne puisse entendre notre communication nous allons utiliser une langue secrète (chiffrement SSL) (contre le man-in-the-middle).

Le phishing consiste pour un serveur escroc à se faire passer pour un site internet valide. Les clients y font des versements ou laisse leurs informations bancaires et sont volés.

Il fallait donc une manière de vérifier l'identité d'un site internet. Le protocole HTTPS prévoit donc un système de certificat qui est décerné à un site pour garantir son authenticité.

Il est possible de créer soi-même un certificat pour son site web mais dans ce cas il ne sera pas reconnu tout de suite par le client : lors de la première connexion le client doit choisir de faire confiance ou non au serveur.

### Faire un essai en ssl sur mon site.

Les sites commerciaux en général paient un organisme de certification qui vérifie que le site n'est pas frauduleux et dans ce cas le client fait automatiquement confiance au serveur.

Une fois l'identité du serveur vérifiée il faut, pour éviter l'attaque par man-in-the-middle, que les échanges entre le client et le serveur restent secrets. Pour cela le protocole HTTPS prévoit de chiffrer (utiliser un code secret) pour rendre les échanges incompréhensible pour une tierce personne. Ce code est appelé le TLS (Transport Layer Security) il repose sur un codage asymétrique construit à partir du certificat du site internet.

## VII PHP et Javascript.

Les pages web que nous avons créés pour l'instant diffusent les mêmes informations à tous les internautes il s'agit donc de sites web statiques.

Il est possible de personnaliser le contenu d'un site web en fonction de chaque utilisateur ce sont des sites dits dynamiques.

Concrètement il ne s'agit plus d'afficher du texte (avec des liens hypertextes) mais d'exécuter des programmes qui permettront d'effectuer des affichages actualisés (afficher l'heure) ou des actions à la demande de l'utilisateur (changer la présentation).

Mais s'il s'agit d'exécuter un programme il y a deux possibilités. Soit il s'agit d'un programme exécuté par l'ordinateur client dans ce cas le langage de programmation est Javascript, soit le programme est exécuté par l'ordinateur serveur dans ce cas le langage de programmation est PHP.

Remarque PHP est un acronyme récursif :

Voici un exemple de page HTML intégrant des éléments PHP et Javascript.

Fichier à tester avec [ceci](#).

```
<!DOCTYPE html>

<html>

  <head>
    <title>
      Heure locale et du serveur.
    </title>
  </head>

  <body>

    <script type="text/javascript">
      var ladate=new Date()
      document.write("Heure sur l'ordinateur client <->
      grâce à javascript: ");
      document.write(ladate.getHours()+" ":"<->
      +ladate.getMinutes()+" ":"+ladate.getSeconds())
    </script>

    <br>

    <?php
      $heure=date("H:i:s");
      Print("Heure sur l'ordinateur serveur grâce à PHP:
      $heure");
    ?>

  </body>

</html>
```

Fichier PHP sans passage à la ligne pour copier coller :

```
<!DOCTYPE html>

<html>

<head>
<title>
```



Heure locale et du serveur.

```
</title>
</head>

<body>

<script type="text/javascript">
var ladate=new Date()
document.write("Heure sur l'ordinateur client grâce à javascript: ");
document.write(ladate.getHours()+":"+ladate.getMinutes()+":"+ladate.getSeconds());
</script>

<br>

<?php
$heure=date("H:i:s");
Print("Heure sur l'ordinateur serveur grâce à PHP: $heure");
?>

</body>

</html>
```

Une très belle et très simple horloge javascript

```
<div id="div_horloge"></div>

<script type="text/javascript">
window.onload=function() {
horloge('div_horloge');
};

function horloge(el) {
if(typeof el!="string") { el = document.getElementById(el); }
function actualiser() {
var date = new Date();
var str = date.getHours();
str += ':'+(date.getMinutes()<10?'0':'')+date.getMinutes();
str += ':'+(date.getSeconds()<10?'0':'')+date.getSeconds();
```

```
el.innerHTML = str;
}
actualiser();
setInterval(actualiser,1000);
}
</script>
```

### Exercice 12.

1. Consultez la page : [http://unemainlavelautre.net/site\\_exemple/exemple\\_php\\_et\\_javascript.php](http://unemainlavelautre.net/site_exemple/exemple_php_et_javascript.php)
2. Créer un dossier php\_javascript dans le répertoire SNT.
3. Téléchargez la page précédente : [lien de téléchargement](#) puis extrayez l'archive.
4. Après avoir ouvert avec Notepad la page obtenue à la question précédente copiez puis collez son contenu sur ce site internet (dans la partie de gauche) : [https://www.w3schools.com/php/phptryit.asp?filename=tryphp\\_intro](https://www.w3schools.com/php/phptryit.asp?filename=tryphp_intro) puis appuyez sur Run.
5. Proposez une explication pour les différences d'affichage obtenus aux questions 1 et 4.

### Exercice 13.

Exemple et explication de PHP (questionnaire)

### Exercice 14.

Quels sont a priori les problèmes de sécurité que peuvent susciter Javascript d'une part et PHP de l'autre.

Si un serveur php est installé sur l'ordinateur il faut ouvrir le fichier PHP en passant par le navigateur.

Pour tester du code php : [lien](#).

Les variables d'environnement (celles concernant client et serveur) : [lien](#)

Créer un formulaire : <https://www.php.net/manual/fr/tutorial.forms.php>

Enregistrer les données d'un formulaire dans un fichier : [lien](#).

Ouvrir un text avec php : [lien](#)

Ouvrir un fichier text en php : <https://phpsources.net/tutoriel-fichiers.htm>

Problème de droit sur un fichier pour qu'il s'ouvre avec php : <https://openclassrooms.com/fr/courses/918836-concevez-votre-site-web-avec-php-et-mysql/913492-li>

Ajouter pi dans le groupe serveur PHP et autoriser l'écriture sur un fichier :  
<https://forum.ubuntu-fr.org/viewtopic.php?id=1623371>  
Récupérer les données d'un formulaire : [lien](#).

## VIII Les moteurs de recherche.

Google, Bing, Yahoo, Baidu, Qwant, Startpage, Duckduckgo.

### L'indexation.

Il s'agit de sites web qui répertorient les pages web et les classes en fonction de leur popularité. Pour cela les moteurs de recherches parcourent en permanence le web.

Pour que le temps de réponse pour une recherche soit court les moteurs de recherche enregistrent des copies de toutes les pages web et procède à leur indexation, c'est-à-dire le classement de la page en fonction de mots clefs.

Des robots (de petits programmes) utilisés par les moteurs de recherche parcourent le web en permanence à la recherche de nouvelles pages en suivant les liens qui apparaissent sur les pages. La plupart des créateurs de sites web installent eux-même ces petits logiciels sur leur site afin que celui-ci soit indexé par le moteur de recherche et donc plus facilement accessible.

La plus grande part du web reste inaccessible et n'est d'ailleurs pas indexé. On parle de « deep web ». Il s'agit de toutes les pages protégées par des mots de passe et en général en PHP. Les pages web de votre environnement Métrice en sont un exemple.

### Les métadonnées et les mots clefs.

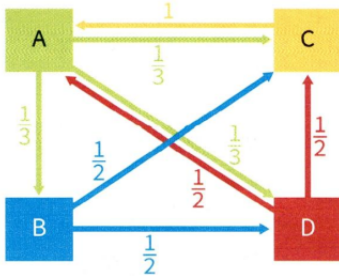
### Les algorithmes.

Chaque moteur de recherche développe normalement ses propres algorithmes de référencement même si ceux-ci tendent à s'uniformiser.

Dans un premier temps référencement naturel :

Pour calculer un score, l'algorithme :

1. part du principe que chaque site possède un « vote » égal à 1. Ce vote se répartit également sur le nombre de liens sortants (qui pointent vers d'autres sites) ;
2. additionne le score total de chaque site ;
3. fait un classement.



Le score du site A est de 2,5, celui du site B de 1,33, celui du site C de 2,33 et celui du site D de 1,83. On obtient donc le classement ou l'ordre d'affichage suivant : site A – site C – site D – site B.

Puis page rank :

## IX Les navigateurs et leurs configurations.

Les bulles de filtre.

## X Les acteurs du web.

- W3C
- ICANN