

# Technologies du Web - TP 1 : HTML, PHP, CSS

## Prise en main de l'environnement de développement

### Éditeur de texte

Le code informatique est du simple texte. Pour écrire un programme dans des langages tels que HTML, CSS ou PHP, il suffit d'un **éditeur de texte simple**. Attention, les outils comme LibreOffice, OpenOffice ou Word ne sont pas des éditeurs de texte simple !

Le plus basique des éditeurs de texte fera l'affaire, mais certains d'entre-eux ont des fonctionnalités dédiées à la programmation : coloration syntaxique, indentation automatique etc. On vous recommande pour ce TP :

- vscode (linux, mac, windows)
- atom (linux, mac, windows),
- gedit ou mousepad (linux)

### Navigateur

C'est le **navigateur Web** qui permet de visualiser des pages HTML. Les plus connus sont Firefox, Chrome, Edge, Internet Explorer, Safari.

Pour ce TP on utilisera Firefox car il est libre, gratuit et disponible sur toutes les plateformes matérielles et logicielles. En outre il inclut des **outils de développement** Web avancés qui facilitent la vie du développeur que vous êtes.

## Programmation des exercices du TD

### Préambule

- Ouvrir Firefox, puis ouvrir le panneau d'outils de développement de Firefox : cliquer sur l'icône du menu -> développement Web -> outils de développement.
- Se rendre dans "Console". Vérifier que tout est activé: Réseau, CSS, JS, Sécurité, sauf Journal (la première fois, cliquer une ou deux fois sur chaque bouton afin de s'en assurer). Cliquer sur "Vider la console" pour faire place nette.
- Taper dans la barre d'adresse : <http://fr.wikipedia.org/>. La page Wikipedia s'affiche; observer les différentes étapes du traitement et les avertissements affichés par la Console.

## Exercice 1 - HTML

- Créer un nouveau fichier texte nommé tp1.html et y saisir le code HTML de la tarte aux prunes. Enregistrer. L'éditeur de texte colore le code : observer quels sont les termes / éléments / classes d'éléments mis en valeur.
- Ouvrir ce fichier dans Firefox. Au choix :
  - fichier -> ouvrir
  - taper dans la barre d'adresse "file://" afin d'explorer le système de fichiers depuis Firefox
  - clic droit sur le fichier -> ouvrir avec -> Firefox
- Utiliser la console pour observer d'éventuels avertissements; le cas échéant, les corriger.

## Exercice 2 - PHP

PHP est un langage exécuté côté serveur. Usuellement, le serveur est une machine distante connectée au réseau; dans ce TP c'est votre machine qui sera à la fois le client et le serveur.

À l'UM, Apache est préconfiguré pour servir des pages se trouvant dans votre répertoire personnel, dans "**public\_html**". Si ce dossier n'existe pas **vous devrez le créer**. **Les pages PHP que vous créerez dans ce TP devront être placées dans ce dossier, ou dans un de ses sous-dossiers.**

Le protocole utilisé pour communiquer avec un serveur Web est **HTTP**. Pour parler à votre propre machine en tant que serveur Web, rendez-vous à l'adresse suivante, qui est dépendante de votre login : <http://localhost/~votrelogin> Si votre login est "adurand" et si vous avez créé dans "public\_html" une page nommée "tp1.php", vous y accéderez donc par <http://localhost/~adurand/tp1.php> et ainsi de suite.

- Renommer votre fichier en tp1.php et y ajouter le code PHP de l'exercice 2 du TD 1, pour afficher plusieurs images à l'aide d'une boucle.
- Ouvrir ce fichier avec le protocole HTTP et vérifier qu'il s'exécute correctement.

## Exercice 3 - CSS

- Créer une feuille de style, par exemple style.css et y ajouter les règles de style de l'exercice 3 du TD.
- Lier cette feuille de style à votre page PHP; actualiser la page dans le navigateur; vérifier que le style est bien appliqué
- Dans les outils de développement, cliquer sur "Inspecteur" et explorer les relations entre HTML et CSS, dans la partie droite de l'outil (Règles, Mise en page).

## Annexe 1 - commandes utiles du navigateur

Actualiser (rafraîchir) la page : F5 ou Ctrl+R

Actualiser la page et vider le cache : Ctrl+F5 ou Ctrl+Maj+R

Ouvrir la console : Ctrl+Maj+K ou F12

Ouvrir un nouvel onglet : Ctrl+T

Ouvrir un lien dans un nouvel onglet : clic central (enfoncer la molette) sur le lien

## Annexe 2 - bonnes pratiques

### Indentation

L'indentation consiste à décaler vers la droite certaines lignes de code avec des tabulations ou des espaces, en fonction de leur niveau d'imbrication, pour rendre les blocs de code plus lisibles.

Le code HTML par exemple présente une forte imbrication : chaque portion de code comprise entre deux balises (une ouvrante et une fermante) est une imbrication, qu'on indentera.

Exemple de HTML indenté:

```
<div> <!-- aucune indentation -->
    <ul> <!-- indentation : une tabulation -->
        <li>premier élément</li> <!-- indentation : deux tabulations -->
        <li>second élément</li>
    </ul> <!-- indentation : une tabulation -->
</div>
```

Exemple de PHP indenté:

```
if (file_exists($path)) { // aucune indentation
    if (is_readable($path)) { // une tabulation
        $ok = true; // deux tabulations
    }
}
```

}

## Annexe 3 - configuration de PHP

Par défaut , les erreurs de votre code PHP se traduisent à la fac par une page blanche, qui ne vous aide pas du tout à comprendre votre erreur. C'est excessivement énervant.

Dans le dossier "public\_html", créez un fichier nommé ".htaccess" (attention, il y a un point au début). Comme tous les fichiers commençant par un point, il ne s'affiche pas dans l'explorateur de fichier de Linux; pour le voir, vous devez afficher les fichiers cachés en appuyant simultanément sur les touches "Ctrl" et "H" de votre clavier, ou en passant par le menu "affichage".

Éditez ce fichier à l'aide de votre éditeur de texte préféré, et placez-y les lignes suivantes :

```
php_flag display_errors on  
php_flag html_errors off
```