



Travaux Pratiques

Par

Prof Edouard Ngor SARR

Enseignant Chercheur en Informatique

UFR SES-Université Assane SECK de Ziguinchor (UASZ)

Ziguinchor-SENEGAL

Email : edouard-ngor.sarr@univ-zig.sn

Site : <https://edouardngorsarr.com/>

Version 2.0 de Mars 2026

Atelier 1 : Environnement de travail pour le développement web

Durée : 2 heures

Objectif : Préparer les participants à configurer un environnement de développement web local, en leur permettant de comprendre le rôle de chaque outil et de créer un premier projet structuré.

Compétences visées :

- Être capable d'installer et configurer un environnement de développement local
- Comprendre le rôle de chaque outil utilisé dans un projet web
- Structurer un projet avec les fichiers de base (HTML, CSS, PHP)

Activités détaillées :

1. Téléchargement et installation des outils essentiels

Les participants installeront trois logiciels indispensables pour le développement web :

- **WAMP SERVER** : Serveur local permettant d'exécuter des scripts PHP et de gérer une base de données MySQL.
- **Visual Studio Code (VS Code)** : Éditeur de code moderne, léger et puissant pour écrire et organiser le code source.
- **FileZilla Client** : Client FTP qui permet de transférer des fichiers vers un serveur distant (introduction au déploiement).

2. Compréhension du rôle de chaque outil

Un moment sera consacré à l'explication de l'utilité de chaque logiciel :

- WAMP pour simuler un serveur local (Apache, PHP, MySQL)
- VS Code pour coder efficacement avec des extensions utiles
- FileZilla pour la gestion des transferts de fichiers dans un contexte professionnel

3. Lancement et prise en main de WAMP SERVER et de Visual Studio Code

Les participants apprendront à :

- Démarrer WAMP et vérifier que les services Apache et MySQL fonctionnent
- Présentation du dossier `www` où sont placés tous les projets web
- Ouvrir le dossier `www` dans VS Code
- Lancer Localhost dans le navigateur

4. Création d'un projet et des fichiers via Visual Studio Code

- Les participants créeront un nouveau projet nommé `PROJET1` dans le dossier `www`
- Ce dossier servira de base pour leurs futurs développements
- À l'intérieur du dossier `PROJET1`, les participants créeront quatre fichiers :
 - `page.html` : une page HTML
 - `style.css` : une feuille de style CSS
 - `module.php` : un fichier PHP
 - `controle.js` : un fichier java script

Atelier 2 : HTML

Durée : 04 h

Exercice 1 : Les balises de base

1. Dans www, Créer un dossier nommé **Atelier2**
2. Dans ce dossier, Créer un fichier **exercice1.html** contenant le code suivant :

```
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Voici mon site</title>
  </head>
  <body>
    <div>div est une balise de type "block"</div>
    <p>la balise p est une autre balise de type block</p>
    <span>En revanche, span est une balise de type "inline"</span>
    <a href="www.google.fr"></a>
    <h1>h1 est bien une balise de type "block"</h1>

    <form>
      <p>voici un champ de texte : <input type="text"/></p>
      <p>voici une checkbox <input type="checkbox"/></p>
      <button>Cliquez ici !</button>
    </form>
  </body>
</html>
```

3. Testez ce code avec votre navigateur
4. Interpréter chaque ligne du code

Exercice 2 : Notions avancés du HTML

1. Dans le dossier **Atelier2**, Créer un fichier **exercice2.html**
2. Implémentez-y le code HTML qui donnera l’affichage de la page ci-dessous.

MA PAGE HTML

Je peux le faire

Je suis un étudiant de la MIO 2, et j'adore le cours sur les Sites web Statiques de *Mr SARR*

Ce texte est très Cool car il se deplace de la gauche vers la droite

Je suis un titre de niveau 3 en italique

Je ne suis pas un lien mais un text souligné

~~Ce texte n'est pas bon~~

[Clique ici pour aller sur Youtube](#)

<https://www.youtube.com/watch?v=qsbkZ7gIKnc>

Signature, je suis à droite



Exercice 3 : Les listes

1. Dans le dossier **Atelier2**, Créer un fichier **exercice3.html**
2. Donnez le code HTML qui donnera l'affichage de la page ci-dessous.

Les mois du printemps

- avril
- mai
- juin

Les mois d'automne

1. octobre
2. novembre
3. décembre

Exercice 4 : les formulaires

1. Dans le dossier **Atelier2**, Créer un fichier **exercice4.html**
2. Réaliser un formulaire Html qui envoie les données via POST vers la cible **module.php** contenant les champs suivants :
 - Prenom champ de texte obligatoire
 - Nom champ de texte obligatoire
 - Date de naissance champ de DATE obligatoire
 - Sexe case à cocher avec comme valeurs M et F obligatoire
 - Adresse : zone de texte de 5 lignes non obligatoire
 - Téléphone : champ de chiffre de 12 caractère non obligatoire. Mettre 77 par défaut
 - Nationalité : un menu SENEGALAISE, MALIENNE ou GABONAISE obligatoire. MALIENNE est choisie par défaut
 - Email : champ de texte obligatoire

- Mot de passe Champ password obligatoire maximum 10 caractère

Exercice 5 : les vidéos

1. Dans le dossier **Atelier2**, Créer un fichier **exercice5.html**
2. Ajouter le code HTML pour y lire cette vidéo YouTube
<https://www.youtube.com/watch?v=68oSvuKVjeU>
3. Ecrire le HTML pour en y intégrant la géolocalisation Google MAP de UASZ.

Exercice 6 : les vidéos

1. Dans le dossier **Atelier2**, Créer un fichier **exercice6.html**
2. Ecrire le HTML pour en y intégrant la géolocalisation Google MAP de UASZ.

Atelier 3 : HTML + CSS

Durée : 2H

Activités :

1. Dans www, Créer un projet ATELIER3
2. Dans ce projet, créer un fichier page.html
3. Coller le code suivant dans votre fichier page.html

```
<!DOCTYPE html>

<html>

<head>

  <meta charset="UTF-8">

  <title>Ma premiere page HTML</title>

  <link rel="stylesheet" href="style.css">

</head>

<body>

  <header>

    

    <h1>Bienvenue sur mon <b>site</b></h1>

    <nav>

      <a href="accueil.html">Accueil</a>

      <a href="inscription.html">Inscription</a>

      <a href="validation.html">Validation</a>

      <a href="index.html">Se deconnecter</a>

    </nav>

    <marquee>

      <p>Un exemple de page <i>HTML</i> avec plusieurs balises</p>

    </marquee>

  </header>

  <!--je fais une liste des licences-->

  <ul>
```


Number 1 : We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications. We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications.

Number 1 : We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications. We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications.

Number 1 : We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications. We solicit high-quality original research papers (and significant work-in-progress papers) in any aspect of Big Data with emphasis on 5Vs (Volume, Velocity, Variety, Value and Veracity), including the Big Data challenges in scientificand engineering, social, sensor/IoT/IoE, and multimedia (audio, video, image, etc.) big data systems and applications.

<table>

<tr>

<th>Prenom</th>

<th>NOM</th>

</tr>

<tr>

<td>Ngor</td>

<td>sarr</td>

</tr>

<tr>

<td>Aliou Badara</td>

<td>FAYE</td>

</tr>

</table>

<footer>

<p>&copy; 2023 MonSiteWeb. Tous droits réservés.</p>

<address> Contact : contact@site.com

</address>

```

</footer>

</body>

</html>

```

4. Interpréter le code HTML
5. Lancer le navigateur
6. Taper au niveau de la barre d'Url Localhost/projet1/page.html
7. Créer un fichier CSS nommé style.css
8. Lier le CSS avec le HTML
9. Implémenter le CSS pour avoir le résultat suivant
 - a. Faire un HOVER pour le menu (balise a) et le texte (balise li)
 - b. Le texte « Un exemple de page HTML avec plusieurs balises » défile de la droite vers la gauche



Bienvenue sur mon site

[Accueil](#)
[Inscription](#)
[Validation](#)
[Se déconnecter](#)

Un exemple de page *HTML* avec plusieurs balises

- Numéro 1 : Nous sollicitons des articles de recherche originaux de haute qualité (et des articles importants en cours de travail) dans n'importe quel aspect du Big Data en mettant l'accent sur les 5V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et d'ingénierie, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.). Nous sollicitons des articles de recherche originaux de haute qualité (et des articles importants en cours de travail) dans n'importe quel aspect du Big Data en mettant l'accent sur les 5V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et d'ingénierie, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.).
- Numéro 1 : Nous sollicitons des articles de recherche originaux de haute qualité (et des articles en cours de réalisation significatifs) sur tous les aspects du Big Data, en mettant l'accent sur les 5 V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et techniques, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.). Nous sollicitons des articles de recherche originaux de haute qualité (et des articles en cours de réalisation significatifs) sur tous les aspects du Big Data, en mettant l'accent sur les 5 V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et techniques, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.).
- Numéro 1 : Nous sollicitons des articles de recherche originaux de haute qualité (et des articles en cours de réalisation significatifs) sur tous les aspects du Big Data, en mettant l'accent sur les 5 V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et techniques, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.). Nous sollicitons des articles de recherche originaux de haute qualité (et des articles en cours de réalisation significatifs) sur tous les aspects du Big Data, en mettant l'accent sur les 5 V (volume, vitesse, variété, valeur et véracité), y compris les défis du Big Data dans les systèmes et applications de Big Data scientifiques et techniques, sociaux, de capteurs/IoT/IoE et multimédias (audio, vidéo, image, etc.).

Prénom	NOM
Ngor	sarr
Aliou Badara	FAYE

© 2023 MonSiteWeb. Tous droits réservés.

Contact : contact@site.com

Atelier 4 : Les bases du PHP avec les formulaires

Durée : 2H

Objectifs :

- Maîtrise des bases du PHP
- Maîtrise des formulaires
- Maîtrise du GET et du POST

Activité 1 : Résolution d'une équation du second degré

Consignes : Écrire un programme PHP qui résout l'équation du second degré $ax^2+bx+c=0$ pour les coefficients donnés

1. Créer un projet nommé **ATELIER4**
2. Un formulaire PHP nommé **index.php**

Logiciel de resolution Equation du second degre

Delta positif donc deux solutions dans R

x² + x +

X1= X2=

3. Proposer un CSS pour le formulaire
4. Créer un fichier **equation.php** dans lequel vous allez
 - i. Calculer le discriminant ($\Delta=b^2-4ac$).
 - ii. Renvoyer les solutions selon les cas :
 1. Si $\Delta<0$: "Aucune solution réelle".
 2. Si $\Delta=0$: "Solution unique : x = [valeur]".
 3. Si $\Delta>0$: "Deux solutions : x1 = [valeur], x2 = [valeur]".

Atelier 5 : Notions avancées du PHP (Les fichiers)

Durée : 2H

Objectifs :

- Maîtrise des formulaires
- Maîtrise des fonctions sur les chaînes de caractères
- Lecture et écriture dans des fichiers

Activités

1. Créer un projet nommé ATELIER5
2. Créer le formulaire HTML nommé page.html qui envoie les valeurs suivantes : Prenom, Nom et Date Naissance vers un fichier mot_passe.php avec la méthode POST

GENERATEUR DE MOT DE PASSE

PRENOM

NOM DE FAMILLE

DATE DE NAISSANCE

MOT DE PASSE

3. Implémenter le fichier **generateur.php** qui :
 - a. Récupère les données envoyées : Prenom, Nom et Date Naissance
 - b. Génère le mot de passe à partir de ces données en respectant les conditions suivantes :
 - Les 03 premiers caractères sont les trois premières lettres du prénom mise en majuscule
 - Le 4em caractère est la 2eme lettre du Nom de famille mise en minuscule ; ^[SEP]
 - Les 5em et 6em caractères sont le nombre de caractère (nb) plus 1 (nb+1) lorsque l'on concatène le ^[SEP]Prénom avec le nom ^[SEP]
 - Les 7em et 8em caractères sont la somme entre le jour et le mois de naissance de l'utilisateur
 - Le 9em caractère est l'avant dernière lettre du Nom de famille ^[SEP]
 - Injecter un @ a la 3em position ^[SEP]
 - c. Enregistre les données (Prenom ; Nom ; Date Naissance ; Mot Passe) ligne par ligne sans écraser les données antérieures dans un **fichier** CSV nommé pwd.csv avec comme colonnes
4. Implémenter le fichier Liste.php qui affiche ligne par ligne les enregistrements du fichier pwd.csv

Exemple

GENERATEUR DE MOT DE PASSE

Mot de passe generé et enregistré

PRENOM

NOM DE FAMILLE

DATE DE NAISSANCE

MOT DE PASSE

Atelier 6 : PHP et les bases de données

Durée : 2H

Objectifs :

- Maîtrise des formulaires : HTML, CSS et JS
- Maîtrise des fonctions PHP sur les bases de données

Activités

1. Créer le projet ATELIER6
2. Implémenter le Formulaire clients.html suivant

The form is titled 'clients.html' and contains the following fields and controls:

- vous Nom d'utilisateur**: A text input field.
- Votre Mot de Passe**: A text input field.
- vous Profession**: A dropdown menu with options: 'Agent immobilier', 'Architecte', and 'Commerce et artisanat'.
- vous Pays**: A dropdown menu with the text 'Votre Choix ...'.
- vous Sexe**: Radio buttons for 'M' (Male) and 'F' (Female).
- vous Langue**: A dropdown menu with options: 'Arabe', 'Anglais', 'Espagnole', and 'Français' (highlighted).
- vous Loisir**: A group of checkboxes for 'Sport', 'Music', 'Internet', 'Voyage', 'Lecture', and 'Autre'.

At the bottom of the form are three buttons: 'Vider', 'Envoyer', and 'Rétablir'.

3. Créer la base de données MySQL nommée BDCLIENTS
4. Créer la table CLIENTS de la base de données en se basant sur le formulaire de la question 2
5. Implémenter le fichier PHP clients.php qui :
 - a. Récupère les informations venant du formulaire
 - b. Enregistre les données dans une table CLIENTS
6. Implémenter le fichier afficher.php qui récupère et affiche toutes les lignes de la table CLIENTS.

Atelier 7 : La POO et le MVC

Durée : 2H

Objectifs :

- Maîtrise des formulaires
- Maîtrise du MVC
- Maîtrise de la POO
- Maîtrise des fonctions PHP sur les bases de données

Activités

On souhaite créer une application de gestion des inscriptions des étudiants de la formation MIO avec HTML, PHP, JS et PDO MYSQL. Vous utiliserez le design Pattern MVC.

Voici les informations des étudiants nécessaires lors de l'inscription :

- Matricule : champ de texte obligatoire (Clé primaire)
- Civilité : Champ Radio qui contient les options Mme et M. ☐ ☐
- Nom : champ de texte obligatoire
- Prénom : champ de texte obligatoire
- Email : champ de texte Email obligatoire
- Téléphone : champ de texte commençant par +221
- Classe : Un menu des classes MIO1, MIO2, MIO3, MMIO1 et MMIO2
- Numero_Paiement : Champ numérique obligatoire
- Date_Paiement : Champ de type Data obligatoire
- Mot_passe : Champ de mot de passe obligatoire

Activité 1 : Créer la table Etudiants dans la base de données MySQL BDInscriptions

Activité 2 : Créer le Projet ATELIER7

Activité 3 : Dans le dossier CLASS du projet

- Créer le fichier de connexion PDO à la base de données bd.class.php
- Créer la classe Connexion_class.php avec la fonction inscription et connexion.
- Créer la classe Etudiants_class.php avec les fonctions ajouter, supprimer, modifier, lister et recherche et incluant le fichier de connexion.

Activité 4 : Dans le dossier VUES Créer les formulaires suivant et lier les avec les contrôleurs

- Formulaire connexion ;
- Formulaire d'Insertion ;
- Formulaire de Listing avec les boutons VOIR, SUPPRIMER, MODIFIER, ;
- Formulaire de Modification ;
- Formulaire Suppression ;
- Formulaire de Recherche.

Activité 5 : Dans le dossier CONTROLERS créer les contrôleurs

1. Etudiants_controller.php : gere les insertions, les modifications, les listings, les recherches et les suppressions
2. Comptes_controller.php : gère les connexion et les créations de comptes

Activité 6 : Lier les vues avec les contrôleurs

Activité 7 : Tester et déployer votre projet

ANNEXE 1 : BALISES HTML

Balise	Description	Exemple d'utilisation
<a>	Crée un lien hypertexte	Lien
<abbr>	Abréviation avec info bulle	<abbr title="HyperText Markup Language">HTML</abbr>
<address>	Donne une adresse de contact	<address>123 rue, Ville</address>
<area>	Zone cliquable dans une image	<area shape="rect" coords="34,44,270,350" href="link.html">
<article>	Contenu autonome (article de blog, etc.)	<article> Titre Contenu </article>
<aside>	Contenu complémentaire (barre latérale)	<aside>Infos connexes</aside>
<audio>	Intègre du son	<audio controls src="audio.mp3"></audio>
	Texte en gras sans importance sémantique	Important
<base>	Définit l'URL de base pour les liens relatifs	<base href="https://site.com/">
<bdi>	Isole du texte directionnel	<bdi>اسم</bdi>
<bdo>	Change la direction du texte	<bdo dir="rtl">Texte</bdo>
<blockquote>	Citation longue	<blockquote>Texte cité</blockquote>
<body>	Contenu principal de la page	<body>Contenu</body>
 	Saut de ligne	Ligne 1 Ligne 2
<button>	Bouton cliquable	<button>Cliquer</button>
<canvas>	Dessins dynamiques en JS	<canvas id="dessin"></canvas>
<caption>	Légende d'un tableau	<caption>Titre du tableau</caption>
<cite>	Titre d'œuvre (livre, film...)	<cite>Le Petit Prince</cite>
<code>	Affiche du code source	<code>console.log()</code>
<col>	Définit une colonne de tableau	<col style="background:yellow">
<colgroup>	Groupe de colonnes	<colgroup><col></colgroup>
<data>	Associe une valeur à un contenu	<data value="123">Produit</data>
<datalist>	Liste de suggestions	<datalist id="noms"><option>Edouard</option></datalist>

Balise	Description	Exemple d'utilisation
<dd>	Description dans une liste	<dd>Définition</dd>
	Texte supprimé	Ancien texte
<details>	Élément déroulant	<details><summary>Voir plus</summary> Contenu </details>
<dfn>	Définit un terme	<dfn>IA</dfn>
<dialog>	Fenêtre de dialogue	<dialog open>Boîte de dialogue</dialog>
<div>	Division générale	<div class="contenu"></div>
<dl>	Liste de définitions	<dl><dt>HTML</dt><dd>Langage web</dd></dl>
<dt>	Terme dans une liste de définitions	<dt>CSS</dt>
	Met en emphase (italique)	Important
<embed>	Contenu externe (PDF, etc.)	<embed src="file.pdf" type="application/pdf">
<fieldset>	Groupe de champs de formulaire	<fieldset><legend>Infos</legend></fieldset>
<figcaption>	Légende d'une figure	<figcaption>Image d'exemple</figcaption>
<figure>	Encapsule une image et sa légende	<figure><figcaption>Légende</figcaption></figure>
<footer>	Pied de page	<footer>© 2025</footer>
<form>	Formulaire	<form action="/submit"></form>
<h1> à <h6>	Titres de section (niveau 1 à 6)	Titre principal
<head>	Métadonnées de la page	<head><title>Titre</title></head>
<header>	En-tête de page ou de section	<header> Bienvenue </header>
<hr>	Ligne horizontale	<hr>
<html>	Racine du document HTML	<html lang="fr"></html>
<i>	Texte en italique sans emphase sémantique	<i>Italique</i>
<iframe>	Cadre intégré (autre page)	<iframe src="page.html"></iframe>
	Image	
<input>	Champ de saisie	<input type="text">
<ins>	Texte ajouté	<ins>Nouveau</ins>
<kbd>	Entrée clavier	<kbd>Ctrl</kbd> + <kbd>C</kbd>
<label>	Étiquette pour un champ	<label for="nom">Nom</label>
<legend>	Légende d'un fieldset	<legend>Informations</legend>
	Élément de liste	Élément

Balise	Description	Exemple d'utilisation
<link>	Lien vers une ressource externe	<link rel="stylesheet" href="style.css">
<main>	Contenu principal	<main> Accueil </main>
<map>	Carte image avec zones	<map name="plan"></map>
<mark>	Met en surbrillance	<mark>Texte surligné</mark>
<meta>	Métadonnées	<meta charset="UTF-8">
<meter>	Jauge	<meter value="0.6">60%</meter>
<nav>	Zone de navigation	<nav> Menu </nav>
<noscript>	Contenu si JS est désactivé	<noscript>JS requis</noscript>
<object>	Contenu externe	<object data="file.swf"></object>
	Liste ordonnée	1er
<optgroup>	Groupe dans une liste déroulante	<optgroup label="Groupe"><option>Choix</option></optgroup>
<option>	Option dans un menu	<option value="1">Un</option>
<output>	Affiche le résultat	<output id="resultat"></output>
<p>	Paragraphe	Texte
<param>	Paramètre d'un objet	<param name="autoplay" value="true">
<picture>	Image responsive	<picture><source srcset="img.webp" type="image/webp"></picture>
<pre>	Texte préformaté	<pre>Ligne 1\nLigne 2</pre>
<progress>	Barre de progression	<progress value="70" max="100"></progress>
<q>	Petite citation	<q>Citation</q>
<rp>	Fallback des rubis	<rp>(</rp>
<rt>	Texte de lecture ruby	<rt>prononciation</rt>
<ruby>	Annotation asiatique	<ruby>漢<rt>kan</rt></ruby>
<s>	Texte barré	<s>Ancien prix</s>
<samp>	Sortie d'un programme	<samp>Erreur 404</samp>
<script>	Code JavaScript	<script>alert("Hello")</script>
<section>	Section de contenu	<section> Titre </section>
<select>	Liste déroulante	<select><option>Choix</option></select>
<small>	Texte secondaire	Avertissement
<source>	Source média alternative	<source src="video.mp4" type="video/mp4">
	Conteneur inline	Texte
	Texte important	Important
<style>	CSS intégré	<style>body {color: red;}</style>
<sub>	Indice bas	H₂0
<summary>	Résumé d'un détail	<summary>Voir</summary>
<sup>	Exposant	x²
<table>	Tableau	<table><tr><td>Cellule</td></tr></table>

Balise	Description	Exemple d'utilisation
<tbody>	Corps de tableau	<tbody><tr></tr></tbody>
<td>	Cellule de tableau	<td>Donnée</td>
<template>	Contenu HTML caché	<template> Modèle </template>
<textarea>	Zone de texte	<textarea>Texte long</textarea>
<tfoot>	Pied de tableau	<tfoot><tr><td>Fin</td></tr></tfoot>
<th>	En-tête de cellule	<th>Titre</th>
<thead>	Tête de tableau	<thead><tr><th>Colonne</th></tr></thead>
<time>	Date/heure	<time datetime="2025-05-14">14 mai 2025</time>
<title>	Titre de la page (onglet)	<title>Ma page</title>
<tr>	Ligne de tableau	<tr><td>1</td></tr>
<track>	Sous-titres vidéo	<track src="subs.vtt" kind="subtitles" srclang="fr">
<u>	Souligné	<u>Souligné</u>
	Liste non ordonnée	Item
<var>	Variable	<var>x</var> = 10
<video>	Vidéo HTML	<video controls src="film.mp4"></video>
<wbr>	Suggère un point de coupure	Super<wbr>califragilistic

Annexe 2 : PROPRIETES CSS

Propriété	Description	Exemple
color	Couleur du texte	color: red;
background-color	Couleur de fond	background-color: yellow;
background-image	Image de fond	background-image: url("img.jpg");
background-size	Taille de l'image de fond	background-size: cover;
background-repeat	Répétition de l'image	background-repeat: no-repeat;
width	Largeur d'un élément	width: 100px;
height	Hauteur d'un élément	height: 200px;
max-width	Largeur maximale	max-width: 100%;
min-width	Largeur minimale	min-width: 300px;
display	Mode d'affichage	display: flex;
position	Positionnement de l'élément	position: absolute;
top	Distance du haut	top: 10px;
left	Distance de la gauche	left: 20px;
right	Distance de la droite	right: 0;
bottom	Distance du bas	bottom: 0;
z-index	Ordre d'empilement	z-index: 2;
margin	Marge externe	margin: 20px;
margin-top	Marge en haut	margin-top: 10px;
margin-bottom	Marge en bas	margin-bottom: 10px;
padding	Marge interne	padding: 15px;
padding-left	Rembourrage gauche	padding-left: 5px;
border	Bordure	border: 1px solid black;
border-radius	Bords arrondis	border-radius: 10px;
box-shadow	Ombre autour d'un élément	box-shadow: 2px 2px 5px grey;
text-align	Alignement du texte	text-align: center;
font-size	Taille du texte	font-size: 16px;
font-weight	Épaisseur du texte	font-weight: bold;
font-family	Police du texte	font-family: Arial, sans-serif;
line-height	Hauteur de ligne	line-height: 1.5;
letter-spacing	Espacement entre lettres	letter-spacing: 2px;
word-spacing	Espacement entre mots	word-spacing: 5px;
text-transform	Transformation du texte	text-transform: uppercase;
text-decoration	Décoration du texte	text-decoration: underline;
overflow	Gestion du débordement	overflow: hidden;
visibility	Visibilité de l'élément	visibility: hidden;
opacity	Opacité de l'élément	opacity: 0.5;
cursor	Apparence du curseur	cursor: pointer;
transition	Animation douce	transition: all 0.3s ease;
transform	Transformation 2D/3D	transform: rotate(45deg);
animation	Définir une animation	animation: slide 2s infinite;
flex-direction	Direction d'un conteneur flex	flex-direction: row;

Propriété	Description	Exemple
<code>justify-content</code>	Alignement horizontal (flex)	<code>justify-content: space-between;</code>
<code>align-items</code>	Alignement vertical (flex)	<code>align-items: center;</code>
<code>gap</code>	Espacement entre éléments (flex/grid)	<code>gap: 10px;</code>
<code>grid-template-columns</code>	Colonnes de la grille	<code>grid-template-columns: 1fr 2fr;</code>
<code>grid-gap</code>	Espace entre les cellules grid	<code>grid-gap: 20px;</code>
<code>float</code>	Position flottante	<code>float: right;</code>
<code>clear</code>	Supprime le float précédent	<code>clear: both;</code>
<code>list-style</code>	Style de liste	<code>list-style: none;</code>
<code>content</code>	Contenu généré (pseudo-élément)	<code>content: "→";</code>
<code>box-sizing</code>	Mode de calcul taille	<code>box-sizing: border-box;</code>
<code>clip-path</code>	Masquage de formes	<code>clip-path: circle(50%);</code>
<code>object-fit</code>	Ajustement des médias	<code>object-fit: cover;</code>
<code>pointer-events</code>	Activation des événements souris	<code>pointer-events: none;</code>
<code>filter</code>	Filtres visuels	<code>filter: grayscale(100%);</code>
<code>backdrop-filter</code>	Filtre sur arrière-plan	<code>backdrop-filter: blur(5px);</code>
<code>white-space</code>	Gestion des espaces blancs	<code>white-space: nowrap;</code>
<code>user-select</code>	Autorise la sélection de texte	<code>user-select: none;</code>
<code>resize</code>	Redimensionnement autorisé	<code>resize: both;</code>
<code>scroll-behavior</code>	Défilement doux	<code>scroll-behavior: smooth;</code>

Annexe 3 : Fonctions PHP

Fonction	Description	Exemple d'utilisation
<code>echo</code>	Affiche une chaîne de caractères	<code>echo "Bonjour";</code>
<code>print</code>	Affiche une chaîne (retourne 1)	<code>print "Salut";</code>
<code>strlen</code>	Longueur d'une chaîne	<code>strlen("chat") // 4</code>
<code>strtoupper</code>	Met en majuscules	<code>strtoupper("bonjour") // BONJOUR</code>
<code>strtolower</code>	Met en minuscules	<code>strtolower("BONJOUR") // bonjour</code>
<code>ucfirst</code>	Première lettre en majuscule	<code>ucfirst("chat") // Chat</code>
<code>ucwords</code>	Première lettre de chaque mot en majuscule	<code>ucwords("bonjour monde") // Bonjour Monde</code>
<code>substr</code>	Extrait une portion de chaîne	<code>substr("bonjour", 0, 3) // bon</code>
<code>strpos</code>	Position d'une sous-chaîne	<code>strpos("bonjour", "j") // 3</code>
<code>str_replace</code>	Remplace dans une chaîne	<code>str_replace("a", "o", "chat") // chot</code>
<code>trim</code>	Supprime les espaces	<code>trim(" bonjour ") // "bonjour"</code>
<code>ltrim</code>	Supprime les espaces à gauche	<code>ltrim(" bonjour")</code>
<code>rtrim</code>	Supprime les espaces à droite	<code>rtrim("bonjour ")</code>
<code>explode</code>	Coupe une chaîne en tableau	<code>explode(",", "a,b,c")</code>
<code>implode</code>	Fusionne un tableau en chaîne	<code>implode("-", ["a","b","c"])</code>
<code>is_array</code>	Vérifie si c'est un tableau	<code>is_array(\$tab)</code>
<code>in_array</code>	Cherche une valeur dans un tableau	<code>in_array("a", ["a","b"]) // true</code>
<code>array_push</code>	Ajoute à la fin du tableau	<code>array_push(\$tab, "x")</code>
<code>array_pop</code>	Supprime le dernier élément	<code>array_pop(\$tab)</code>
<code>array_shift</code>	Supprime le premier élément	<code>array_shift(\$tab)</code>
<code>array_unshift</code>	Ajoute au début du tableau	<code>array_unshift(\$tab, "x")</code>
<code>array_merge</code>	Fusionne deux tableaux	<code>array_merge(\$a, \$b)</code>
<code>array_keys</code>	Récupère les clés	<code>array_keys(\$tab)</code>
<code>array_values</code>	Récupère les valeurs	<code>array_values(\$tab)</code>
<code>array_slice</code>	Coupe une partie d'un tableau	<code>array_slice(\$tab, 1, 2)</code>
<code>count</code>	Compte les éléments	<code>count(\$tab)</code>
<code>sizeof</code>	Alias de <code>count</code>	<code>sizeof(\$tab)</code>
<code>sort</code>	Trie un tableau	<code>sort(\$tab)</code>
<code>rsort</code>	Trie en ordre inverse	<code>rsort(\$tab)</code>
<code>asort</code>	Trie par valeur, garde les clés	<code>asort(\$tab)</code>
<code>ksort</code>	Trie par clé	<code>ksort(\$tab)</code>
<code>shuffle</code>	Mélange un tableau	<code>shuffle(\$tab)</code>
<code>array_reverse</code>	Inverse un tableau	<code>array_reverse(\$tab)</code>
<code>isset</code>	Vérifie si une variable est définie	<code>isset(\$x)</code>
<code>empty</code>	Vérifie si vide	<code>empty(\$x)</code>
<code>is_null</code>	Vérifie si null	<code>is_null(\$x)</code>
<code>unset</code>	Détruit une variable	<code>unset(\$x)</code>
<code>gettype</code>	Type d'une variable	<code>gettype(\$x)</code>
<code>settype</code>	Change le type	<code>settype(\$x, "string")</code>
<code>print_r</code>	Affiche les tableaux/objets	<code>print_r(\$tab)</code>

Fonction	Description	Exemple d'utilisation
<code>var_dump</code>	Affiche le contenu et le type	<code>var_dump(\$x)</code>
<code>is_numeric</code>	Vérifie si numérique	<code>is_numeric("123") // true</code>
<code>is_string</code>	Vérifie si chaîne	<code>is_string(\$x)</code>
<code>is_int</code>	Vérifie si entier	<code>is_int(\$x)</code>
<code>is_bool</code>	Vérifie si booléen	<code>is_bool(\$x)</code>
<code>is_object</code>	Vérifie si objet	<code>is_object(\$x)</code>
<code>min</code>	Plus petit nombre	<code>min(1, 2, 3)</code>
<code>max</code>	Plus grand nombre	<code>max(1, 2, 3)</code>
<code>round</code>	Arrondir	<code>round(4.5)</code>
<code>ceil</code>	Arrondi supérieur	<code>ceil(4.1) // 5</code>
<code>floor</code>	Arrondi inférieur	<code>floor(4.9) // 4</code>
<code>abs</code>	Valeur absolue	<code>abs(-10) // 10</code>
<code>rand</code>	Nombre aléatoire	<code>rand(1, 100)</code>
<code>mt_rand</code>	Plus rapide que rand	<code>mt_rand(1, 100)</code>
<code>date</code>	Formatage de date	<code>date("Y-m-d")</code>
<code>time</code>	Timestamp actuel	<code>time()</code>
<code>strtotime</code>	Convertit date en timestamp	<code>strtotime("tomorrow")</code>
<code>mktime</code>	Timestamp personnalisé	<code>mktime(0, 0, 0, 12, 31, 2025)</code>
<code>sleep</code>	Pause en secondes	<code>sleep(2)</code>
<code>include</code>	Inclut un fichier	<code>include("fichier.php")</code>
<code>require</code>	Inclut (interrompt si erreur)	<code>require("config.php")</code>
<code>include_once</code>	Inclut une fois	<code>include_once("header.php")</code>
<code>require_once</code>	Comme require, mais une seule fois	<code>require_once("lib.php")</code>
<code>function_exists</code>	Vérifie si une fonction existe	<code>function_exists("myFunc")</code>
<code>call_user_func</code>	Appelle dynamiquement une fonction	<code>call_user_func("myFunc", \$arg)</code>
<code>explode</code>	Chaîne vers tableau	<code>explode(", ", "a,b")</code>
<code>implode</code>	Tableau vers chaîne	<code>implode(", ", ["a", "b"])</code>
<code>header</code>	Envoie un en-tête HTTP	<code>header("Location: index.php")</code>
<code>exit</code>	Arrête le script	<code>exit("Fin")</code>
<code>die</code>	Alias de exit	<code>die("Erreur")</code>
<code>fopen</code>	Ouvre un fichier	<code>fopen("fichier.txt", "r")</code>
<code>fread</code>	Lit un fichier	<code>fread(\$f, filesize("fichier.txt"))</code>
<code>fwrite</code>	Écrit dans un fichier	<code>fwrite(\$f, "Texte")</code>
<code>fclose</code>	Ferme un fichier	<code>fclose(\$f)</code>
<code>file_get_contents</code>	Lit un fichier (tout)	<code>file_get_contents("page.html")</code>
<code>file_put_contents</code>	Écrit dans un fichier	<code>file_put_contents("f.txt", "texte")</code>
<code>file_exists</code>	Vérifie si un fichier existe	<code>file_exists("photo.jpg")</code>
<code>is_file</code>	Vérifie si c'est un fichier	<code>is_file("notes.txt")</code>
<code>is_dir</code>	Vérifie si c'est un dossier	<code>is_dir("mon_dossier")</code>
<code>mkdir</code>	Crée un répertoire	<code>mkdir("nouveau")</code>
<code>rmdir</code>	Supprime un dossier	<code>rmdir("ancien")</code>

Fonction	Description	Exemple d'utilisation
<code>unlink</code>	Supprime un fichier	<code>unlink("fichier.txt")</code>
<code>basename</code>	Nom du fichier	<code>basename("/chemin/fichier.txt")</code>
<code>dirname</code>	Dossier du fichier	<code>dirname("/chemin/fichier.txt")</code>
<code>pathinfo</code>	Infos sur le fichier	<code>pathinfo("test.txt")</code>
<code>json_encode</code>	Convertit en JSON	<code>json_encode(\$data)</code>
<code>json_decode</code>	JSON vers tableau/objet	<code>json_decode(\$json)</code>
<code>htmlspecialchars</code>	Échappe les balises HTML	<code>htmlspecialchars("test")</code>
<code>htmlentities</code>	Convertit tous les caractères spéciaux	<code>htmlentities("é") // &eacute;</code>
<code>md5</code>	Hash en MD5	<code>md5("secret")</code>
<code>sha1</code>	Hash en SHA1	<code>sha1("secret")</code>
<code>password_hash</code>	Hash sécurisé	<code>password_hash("pass", PASSWORD_DEFAULT)</code>
<code>password_verify</code>	Vérifie un mot de passe hashé	<code>password_verify("pass", \$hash)</code>

Annexe 4 : Fonctions Java Script

Fonction	Description	Exemple
<code>alert</code>	Affiche une boîte d'alerte	<code>alert('Bonjour !');</code>
<code>console.log</code>	Affiche un message dans la console	<code>console.log('Debug');</code>
<code>prompt</code>	Demande une entrée utilisateur	<code>let name = prompt('Votre nom ?');</code>
<code>confirm</code>	Boîte de confirmation	<code>confirm('Êtes-vous sûr ?');</code>
<code>parseInt</code>	Convertit une chaîne en entier	<code>parseInt('42') // 42</code>
<code>parseFloat</code>	Convertit une chaîne en flottant	<code>parseFloat('3.14') // 3.14</code>
<code>isNaN</code>	Vérifie si ce n'est pas un nombre	<code>isNaN('abc') // true</code>
<code>isFinite</code>	Vérifie si c'est un nombre fini	<code>isFinite(10) // true</code>
<code>Number</code>	Convertit en nombre	<code>Number('10') // 10</code>
<code>String</code>	Convertit en chaîne	<code>String(123) // '123'</code>
<code>Boolean</code>	Convertit en booléen	<code>Boolean(0) // false</code>
<code>typeof</code>	Retourne le type	<code>typeof 'abc' // 'string'</code>
<code>Array.isArray</code>	Vérifie si c'est un tableau	<code>Array.isArray([1,2]) // true</code>
<code>Object.keys</code>	Retourne les clés d'un objet	<code>Object.keys({a:1}) // ['a']</code>
<code>Object.values</code>	Retourne les valeurs	<code>Object.values({a:1}) // [1]</code>
<code>Object.entries</code>	Paires clé/valeur d'un objet	<code>Object.entries({a:1})</code>
<code>Math.random</code>	Nombre aléatoire	<code>Math.random()</code>
<code>Math.floor</code>	Arrondit vers le bas	<code>Math.floor(4.7) // 4</code>
<code>Math.ceil</code>	Arrondit vers le haut	<code>Math.ceil(4.2) // 5</code>
<code>Math.round</code>	Arrondit à l'entier le plus proche	<code>Math.round(4.5) // 5</code>
...	...	...