**FICHE DE REVISION**

**Web - Introduction**

**Ce qu’il faut savoir**

● Le "World Wide Web", plus communément appelé "Web" a été développé au début des années 90 pour faciliter les échanges d’informations avec un système hypertexte permettant de consulter des documents à partir d’un document en cliquant sur des mots clés, les hyperliens

● le web se base sur trois choses : le protocole HTTP (HyperText Transfert Protocol), les URL (Uniform Resource Locator) et le langage de description HTML (HyperText Markup Language).

● il ne faut pas confondre Internet et le Web : Internet est un "réseau de réseaux" s'appuyant sur le protocole IP alors que le web est la combinaison de trois technologies : HTTP, URL et HTML. Le Web “utilise” Internet (sans Internet le Web n’aurait sans doute jamais vu le jour, alors qu’Internet existait avant le Web)

**Web – Les URL**

**Ce qu’il faut savoir**

● Une URL (Uniform Resource Locator) permet d'identifier une ressource (par exemple un fichier) sur un réseau. Il indique « l’endroit » où se trouve une ressource sur un ordinateur.

● Pour indiquer la position d'un fichier (ou d'un dossier) dans une arborescence, il existe 2 méthodes : indiquer un chemin absolu ou indiquer un chemin relatif.

● Chemin absolu : doit indiquer « le chemin » depuis la racine. Exemple: L’URL du fichier « fichier3.jpg » sera : /dossier2/dossier3/fichier3.jpg 🡪 on démarre bien depuis la racine = « / »

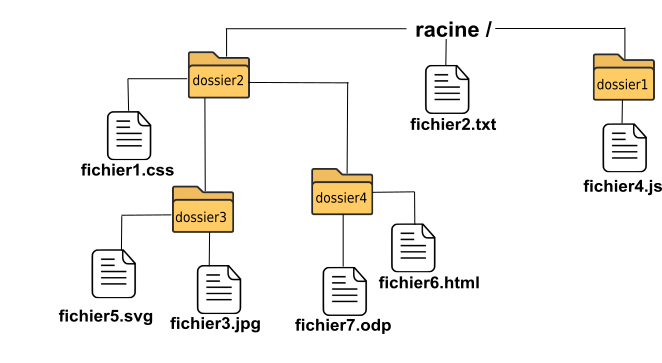
● Chemin relatif : on le distingue du chemin absolu par l’abscence de « / » au début. Il permet d’indiquer le chemin non pas depuis la racine mais depuis un dossier qui accueille un fichier. Exemple: le fichier fichier1.css fait appel au fichier3.jpg. On écrit alors : dossier3/fichier3.jpg 🡪absence du / au début qui montre qu’on ne commence pas par la racine

● « Remonter d’un niveau » : Pour remonter d’un cran, il faut utiliser deux points 🡪 .. Pour remonter de plusieurs crans, il suffit de mettre plusieurs foix les deux points 🡪 ../../

Remarque : la façon d'écrire les chemins (avec des slash (/) comme séparateurs) est propre aux systèmes dits « UNIX », par exemple GNU/Linux ou encore Mac OS. Sous Windows, ce n'est pas le slash qui est utilisé, mais l'antislash (\). Pour ce qui nous concerne ici, les chemins réseau (et donc le web), pas de problème, c'est le slash qui est utilisé.

**Ce qu’il faut savoir faire**

Être capable de donner le chemin permettant d’atteindre un fichier (chemin absolu et chemin relatif) pour une arborescence donnée.



**structure en arborescence**

**lien du site d’informations:**  <https://pixees.fr/informatiquelycee/n_site/nsi_prem.html>

**Web – HTML et CSS**

**Ce qu’il faut savoir**

● Le HTML est un langage de balisage conçu pour représenter des pages web.

● Dans une page HTML on trouve uniquement des balises ouvrantes ( exemple <p>) et fermantes (exemple </p>) ou des balises à la fois ouvrantes et fermantes (exemple <img/>)

● Les navigateurs web (Firefox, Chrome, …) sont des logiciels capables d’interpréter du HTML et d’afficher la page web correspondante.

● Chaque balise a une signification qu’il faut bien respecter (on parle de la sémantique des balises).

● Le HTML n’a pas été conçu pour gérer la mise en page. Le HTML s’occupe uniquement du contenu et de la sémantique, pour tout ce qui concerne la mise en page et l’aspect « décoratif » ( on parle du style de la page), on utilisera un autre langage, le CSS ( Cascading Style Sheets ).

● Il est possible d’ajouter des attributs dans les balises (exemple <p id = « monPara »>, <p class= « mesPara »>).

**Ce qu’il faut savoir faire**

● Savoir utiliser un certain nombre de balises : la balise « paragraphe » <p> ; les balises titres <h1>, <h2>, … ; la balise lien <a> ; la balise image <img> ; la balise formulaire <form> ; la balise <input> ; la balise <button> et les balises génériques <div> et <span>

● Savoir à partir d’un code HTML et d’un code CSS être capable de « deviner » l’aspect d’une page web (couleur d’un texte, police de caractère, s