

3.1 Arborescence

Quand on crée un site web, on manipule plusieurs fichiers :

- des pages .html;
- des images .png, .jpg, .webp;
- parfois d'autres dossiers (styles, documents, etc.).

Pour ne pas se perdre, on range tout dans des **dossiers**. L'ensemble des dossiers et des fichiers forme une **arborescence**.

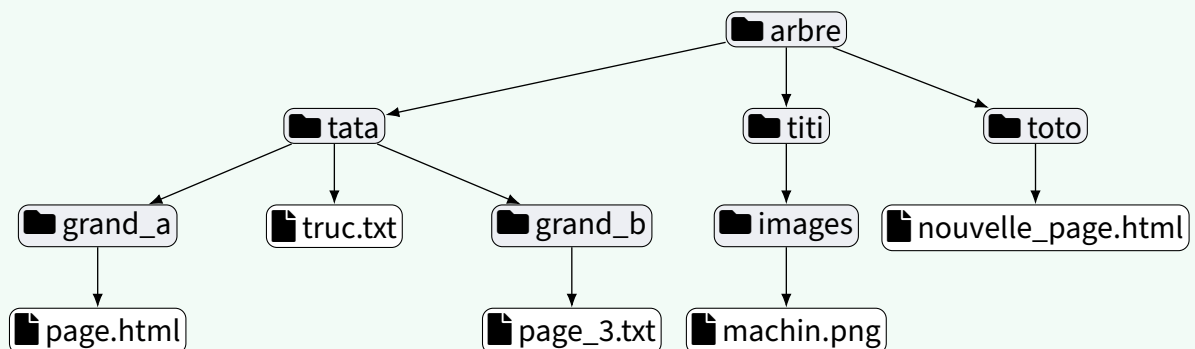
Définition 1 — Arborescence

Une **arborescence** est une organisation en dossiers et sous-dossiers. On peut repérer un fichier grâce à son **chemin**.

Exemple 1 — Exemple d'arborescence

On considère l'arborescence suivante :

Légende : 📁 = dossier 📄 = fichier



3.2 Chemins

Un **chemin** indique comment accéder à un fichier dans l'arborescence.

Définition 2 — Chemin absolu

Un **chemin absolu** donne le chemin complet **depuis la racine** de l'arborescence. Dans ce TP, on prendra la forme :

`/arbre/.../fichier.extension`

Définition 3 — Chemin relatif

Un **chemin relatif** est écrit **à partir de l'endroit où l'on se trouve**. Exemple : si on est dans le dossier **titi**, on peut écrire :

`images/machin.png`

💡 Remonter d'un cran

Pour **remonter d'un dossier** (faire « demi-tour »), on utilise :

`../`

Exercice 1 — Chemins dans l'arborescence

En utilisant l'arborescence donnée plus haut :

1. Quel est le dossier à la **racine** de cette arborescence?

.....

2. Donner le **chemin absolu** vers le fichier `nouvelle_page.html`.

.....

3. Donner le **chemin absolu** vers le dossier `grand_a/`.

.....

4. On se situe dans le dossier `titi/`. Donner le **chemin relatif** pour accéder à l'image `machin.png`.

.....

5. On se situe dans le dossier `grand_a/`. Donner le **chemin relatif** pour accéder à l'image `machin.png`.

.....

3.3 Créer un site sans capytale

Pour travailler en informatique, il est important de bien maîtriser l'environnement de travail, en suivant plusieurs étapes nous allons voir comment bien mettre en place cet environnement.

Exercice 2 — Créer le dossier et les fichiers

Dans cette partie, le but est de créer les dossiers et fichiers **à l'aide de l'Explorateur de fichiers de Windows**.

Étape 1 — Ouvrir l'Explorateur de fichiers

- Ouvrir l'Explorateur de fichiers (icône dossier jaune ou raccourci Win + E).
- Se rendre dans **Documents** ou sur le **Bureau** (au choix).

Étape 2 — Créer le dossier du projet

- Effectuer un clic droit dans la zone blanche.
- Choisir **Nouveau** → **Dossier**.
- Nommer ce dossier :

SNT_WEB_TP3

Étape 3 — Créer les fichiers HTML

- Entrer dans le dossier SNT_WEB_TP3.
- Effectuer un clic droit → **Nouveau** → **Document texte**.
- Renommer le fichier en :

index.html

- Vérifier que le fichier se termine bien par .html et non .txt.

Répéter la même opération pour créer le fichier :

page2.html

Étape 4 — Créer le dossier d'images

- Toujours dans le dossier SNT_WEB_TP3, effectuer un clic droit.
- Choisir **Nouveau** → **Dossier**.
- Nommer ce dossier :

img

À la fin de cette étape, le dossier SNT_WEB_TP3 doit contenir :

- index.html
- page2.html
- le dossier img/



Choisir un éditeur

Visual Studio Code (souvent appelé **VS Code**) est un **éditeur de code**.

Il sert à écrire et modifier des **fichiers texte**, par exemple des pages web en HTML. Contrairement à un navigateur, VS Code **n'affiche pas les pages web** : il permet d'écrire leur contenu.

VS Code facilite le travail grâce à une présentation claire des dossiers et des fichiers, à la coloration du code et à l'aide qu'il apporte lors de l'écriture. C'est un outil très utilisé pour créer des sites web et des programmes.

Exercice 3 — Ouvrir le dossier dans VS Code

Étape 1 — Ouvrir VS Code

Vous pouvez le trouver dans la barre de recherche Windows

Étape 2 — Ouvrir le dossier du projet

Dans VS Code :

1. Menu **Fichier** → **Ouvrir le dossier...**
2. Choisir SNT_WEB_TP3

Étape 3 — Vérifier

À gauche (Explorateur), on doit voir :

- index.html
- page2.html
- img/

Exercice 4 — Écrire et tester index.html

Maintenant que nos fichiers et dossiers sont prêts, nous pouvons commencer à créer le contenu de notre site. **Étape 1 — Écrire le code**

Dans index.html, écrire une page HTML complète contenant :

- un <title> : *Mon mini-site*
- un <h1> : *Bienvenue*
- un <p> : une phrase de présentation

Étape 2 — Sauvegarder

Utiliser **Ctrl+S**.

Étape 3 — Ouvrir dans le navigateur

Dans l'explorateur de fichiers, double-cliquer sur index.html.

Étape 4 — Aller-retour

Modifier le paragraphe dans VS Code, sauvegarder, puis **rafraîchir** le navigateur (**F5**).

Exercice 5 — Relier deux pages avec <a>

Désormais nous souhaitons pouvoir passer d'une page à l'autre grâce à un lien **hypertexte**

Étape 1 — Écrire page2.html

Dans page2.html, écrire une page HTML complète contenant :

- un <title> : *Page 2*
- un <h1> : *Deuxième page*
- un <p> : une phrase

Étape 2 — Ajouter un lien dans index.html

Dans index.html, ajouter un lien cliquable vers page2.html.

Étape 3 — Ajouter un lien retour dans page2.html

Dans page2.html, ajouter un lien cliquable vers index.html.

Étape 4 — Tester

Ouvrir index.html dans le navigateur et vérifier que les liens fonctionnent.

**Exercice 6 — Afficher une image avec **

Dans cet exercice, le but est de d'abord télécharger une image avant de la mettre sur le site !

Étape 1 — Récupérer une image

Télécharger une image (ex : une planète) sur Internet.

Étape 2 — Placer l'image

Mettre cette image dans le dossier img/ du projet.

Étape 3 — Renommer

Renommer l'image en image.png (ou image.jpg selon le format).

Étape 4 — Afficher l'image

Dans index.html, ajouter une balise qui :

- utilise src avec un **chemin relatif** vers img/ ;
- contient un alt décrivant l'image en une courte phrase.