

HTML & CSS — Récapitulatif (Version lisible)

Objectif

Ce document résume les concepts et bonnes pratiques HTML & CSS que vous avez vus : structure HTML, éléments de texte, listes, tableaux, formulaires, CSS de base, Flexbox, Grid, media queries et bonnes pratiques.

HTML — Structure et éléments essentiels

- `<!DOCTYPE html>`, `<html>`, `<head>`, `<body>`
- Balises structurantes : `header`, `main`, `footer`, `section`, `article`, `nav`, `aside`
- Titres : `h1` ... `h6`
- Texte : `p`, `span`, `strong`, `em`, `mark`, `br`
- Liens : `a href`, `target`, `id` pour ancrer, `mailto:`
- Images : `img src alt` (toujours définir `alt`)
- Listes : `ul`, `ol`, `li`
- Tableaux : `table`, `thead`, `tbody`, `tfoot`, `tr`, `td`, `th`, `rowspan`, `colspan`, `caption`
- Formulaires : `form`, `input`, `label`, `textarea`, `select`, `option`, `fieldset`, `legend`
- Attributs importants : `name`, `value`, `placeholder`, `required`, `type`, `maxlength`, `autofocus`

CSS — Fondamentaux

- Sélecteurs : élément, `.classe`, `#id`, descendant, pseudo-classes (`:hover`, `:visited`, `:active`...)
- Box Model : `content` ↔ `padding` ↔ `border` ↔ `margin` (penser au reset `* { box-sizing: border-box }`)
- Couleurs : hex (`#RRGGBB`), `rgb()`, `rgba()` (avec alpha)
- Polices : `font-family`, `font-size`, `font-style`, `font-weight`, `line-height`
- Texte : `text-align`, `text-decoration`, `text-shadow`, `letter-spacing`

Mise en page moderne

Flexbox (pour aligner une ligne/colonne)

- `display: flex`
- `flex-direction`, `justify-content`, `align-items`, `flex-wrap`, `gap`, `order`, `align-self`

Grid (pour layouts complexes)

- `display: grid`
- `grid-template-columns`, `grid-template-rows`
- `grid-column`, `grid-row`, `grid-area`
- `repeat()`, `fr` units, `gap`

Positionnement

- `position: static` | `relative` | `absolute` | `fixed` | `sticky`
- `top`, `left`, `right`, `bottom`, `z-index`

Images, backgrounds, et médias

- `background-image`, `background-size` (`cover` | `contain`), `background-position`, `background-repeat`
- `opacity` (attention : affecte tout l'élément)
- responsive images : `srcset`, `sizes` (next step)
- media queries : `@media` (`max-width: 1024px`) – `mobile-first` vs `desktop-first`

Formulaires et interactions

- Styliser `input`, `textarea`, `select`
- Pseudo-classes pratiques : `:required`, `:focus`, `:checked`
- `Fieldset` + `legend` pour regrouper des champs

- Validation : required, type=email, pattern (regex)
- Envoi : method GET vs POST, action (URL)

Bonnes pratiques (raccourci)

- Utiliser des classes pour le style (éviter les IDs pour CSS)
- Organiser le CSS (reset, base, layout, components, utilities)
- Utiliser des unités relatives : rem / em / % / fr
- Tester le responsive avec les devtools (device toolbar)
- Éviter position: absolute/global pour des éléments courants (ex: a)
- Rendre les images accessibles (alt) et utiliser des tailles adaptées

Prochaines étapes recommandées

- Construire un mini-portfolio (index.html, projects.html, about.html)
- Pratiquer Flexbox + Grid sur des composants (cards, hero, navbar)
- Introduire JavaScript pour DOM (menu mobile, filtres de projets)
- Ensuite apprendre TypeScript pour frontend structuré

Fin du document. Bon travail — tu as de solides bases.