

Conception de documents et d'interfaces numériques

TP n° 3 : Web

Durée : 6h

Système d'exploitation : au choix.

Objectifs du TD :

- Comprendre la structure d'une page HTML ;
- Se familiariser avec les balises HTML de mise en forme ;
- Savoir comment créer et inclure une feuille CSS ;
- Comprendre le système de blocs et le positionnement CSS ;
- Utiliser le validateur W3C ;
- Comprendre la conception de site Web adaptatifs.

Rendu : Voir les consignes en fin de sujet.

I) Bases de HTML

Dans toute cette section, il ne faut pas trop se soucier du résultat affiché, seulement s'assurer que vous appliquez les bons éléments (la bonne **structure**). On améliorera la présentation en CSS plus tard.

Question 1 (Création). Choisissez un éditeur de texte brut (gedit, notepad...) et créez un document `cv.html`.

→ Lisez les parties 4.2 et 4.3 du polycopié et reproduisez la structure de base d'une page Web (avec éléments `html`, `head`, `body` et l'encodage indiqué dans `<meta charset = "..." />`).

→ Écrivez un texte quelconque dans l'élément `body`, puis ouvrez la page dans un navigateur pour constater le résultat.

⚠ Il faudra recharger cette page à chaque fois pour constater le résultat des changements faits dans le document HTML. Si vos changements n'ont aucun effet, vérifiez que :

1. vous avez bien sauvegardé votre document HTML ;
2. le document que vous éditez est le même que la page ouverte.

Question 2 (Découpage en sections). Nous allons commencer par définir les sections principales du CV.

→ Créez un titre principal « CV de (votre nom) ». Ce sera un élément `h1` (c'est-à-dire du texte compris entre des balises `<h1>` et `</h1>`).

→ Créez quatre sections (éléments `section`) intitulées « À propos de moi », « Formation », « Compétences » et « Contact ». Ces titres se placent dans des éléments « titres de niveau 2 ».

⚠ On a ici le premier exemple d'élément sans effet sur l'affichage. Ce type d'élément donne la structure du document et leur présentation pourra être changée plus tard en CSS (séparation structure / présentation). Parfois, ces éléments servent aussi pour l'extraction automatique d'information (moteurs de recherche...)

Question 3 (Section « À propos de moi »). Pour cette section, écrivez quelques phrases sur votre origine, âge, votre formation actuelle, le métier auquel vous aimeriez vous destiner, etc.

→ Séparez ces phrases en (au moins) deux éléments paragraphes. Constatez l'effet sur l'affichage.

→ Mettez en valeur un ou deux points qui vous paraissent importants (par exemple, votre lieu de formation) à l'aide d'un élément adapté.

Question 4 (Section « Formation »). Cette section parle de vos études.

→ Commencez la section par une phrase indiquant votre lieu de formation. Elle contiendra un lien vers la description de la formation de l'IUT, par exemple `http://www.iut-orsay.u-psud.fr/fr/formations/dut/info_dut_fi.html`. Attention : cette phrase est un paragraphe.

→ Faites une liste (non numérotée) indiquant les cours que vous suivez à l'IUT.

Question 5 (Section « Compétences »). Dans cette section on résume vos compétences en différents langages, avec années d'expérience et niveau d'expertise.

→ Créez un tableau (élément `table`). Le résultat final doit ressembler à :

Langage	Expérience	Niveau
Français	30 ans	Débutant
HTML	1h	Expert

⚠ Pour le moment votre tableau n'aura pas de bordure. On s'en occupera en CSS.

Question 6 (Section « Contact »). Cette section comprend les informations nécessaires pour vous contacter (nom, téléphone, adresse email...)

→ Essayez de trouver l'élément HTML servant à indiquer les indications de contact pour l'auteur de la page dans un site de référence tel que (`https://developer.mozilla.org/fr/docs/Web/HTML/Element`).

⚠ En général, on fait les sauts de ligne en changeant de paragraphe : `</p> <p>`. Mais pour les lignes d'une adresse, qui ne sont pas des paragraphes différents, on peut utiliser la balise de saut de ligne `
` (*line break*).

Question 7 (Ajout d'une photo). Nous allons ajouter une photo en haut à gauche de la page. Vous pouvez utiliser au choix une photo de vous que vous auriez sous la main, ou l'image `Portrait.jpg` que l'on peut trouver sur le Moodle.

→ Insérez l'image après le titre en utilisant l'élément ``. L'attribut `src` désigne le chemin vers l'image depuis là où votre document HTML est enregistré.

⚠ `img` est une balise sans contenu ; elle vient seule. Aussi, on écrit `` au lieu de ` `.

→ Indiquez la taille (en pixels) de l'image via les attributs `width` et `height`. Pour obtenir la taille de l'image, faites **clic droit** → **Propriétés** dans l'explorateur de fichiers.

⚠ Vous devez bien indiquer la taille originale de l'image dans ces attributs. Ils ne sont pas là pour mettre l'image à la bonne taille pour la mise en page : on fera ça en CSS. Le but est que le navigateur connaisse la taille de l'image avant de la télécharger, pour éviter que la page « bouge » lors du chargement.

→ Indiquez sous la photo son origine : par exemple « Portrait de l'artiste », Vincent Van Gogh, Musée d'Orsay (domaine public). Pour cela, placez après la photo un élément `figcaption` (légende), et mettez le tout dans un élément `figure`.

Question 8 (Navigation dans le CV). De même qu'un sommaire sert à se retrouver et à naviguer dans un document texte, les documents HTML ont différents outils qui aident à la navigation dans le document. Nous allons utiliser l'attribut générique `id` pour permettre la navigation dans la page.

→ Associez un attribut `id` différent à chaque élément `section`, avec un nom descriptif.

→ Dans la barre d'adresse, ajoutez à la fin de l'url de la page un dièse et un nom d'identifiant (par exemple : ... `file.html#Contact`). Que se passe-t-il ?

⚠ S'il ne se passe rien, essayez de zoomer sur la page (`Ctrl+`) et réessayez.

→ Créez un menu de navigation placé dans l'en-tête (vérifiez les éléments correspondants). Dans ce menu, créez des liens vers les différentes sections de votre CV : utilisez des dièses dans les attributs `href` (par exemple `href="#Formation"`).

II) Un peu de CSS

Question 9 (Création de la feuille de style). Créez une feuille de style vide au format CSS (texte brut avec extension `.css`).

→ Dans la feuille CSS, ajoutez une ligne `h1 {color:valeur;}`, où `valeur` est un nom de couleur.

⚠ Pour choisir une valeur pour `color`, vous pouvez vous référer à cette liste de noms de couleurs prédéfinies : https://developer.mozilla.org/fr/docs/Web/CSS/Type_color.

→ Reliez cette feuille de style à votre page HTML en utilisant dans l'élément `head` la balise `<link rel="stylesheet" href="chemin/vers/votre/feuille/CSS"/>`. Constatez le résultat.

⚠ Le chemin part de l'endroit où est rangé votre document HTML. Si la feuille CSS est dans le même dossier, il suffira d'écrire `href="titre.css"`.

Question 10 (Mise en forme des titres). Modifiez la police utilisée dans certains titres (`h1` ou `h2`, à votre choix). Pour choisir la police, référez-vous à https://www.w3schools.com/cssref/css_websafe_fonts.asp. Fixez aussi la taille que vous désirez avec `font-size`.

Question 11 (Mise en forme de l'image). Nous allons redimensionner l'image et modifier l'apparence de la légende. On attendra la section suivante pour la positionner à droite ou à gauche du texte.

→ Appliquez à l'image les propriétés `width:x`; `height:auto`; en choisissant une bonne valeur pour `x` (en cm, ou px...). La valeur pour `height` garantit que l'image n'est pas déformée.

→ Modifiez la légende pour que la police soit plus petite que la normale et en italique. Utilisez l'aide sur Moodle ou cherchez sur Internet pour trouver les propriétés CSS correspondantes.

Question 12 (Mise en forme de la liste). Nous allons modifier les choix de puces dans la liste de cours suivis.

→ Allez vous renseigner sur la propriété CSS `list-style-type` et les valeurs qu'elle peut prendre sur <https://developer.mozilla.org/fr/docs/Web/CSS/list-style-type>.

→ Remplacez les puces de base par votre option préférée en appliquant le CSS à l'élément `` qui constitue la liste.

→ Donnez une puce particulière au cours que vous préférez et mettez son nom en valeur. Pour cela, on donnera à l'élément `li` correspondant un attribut `class` pour pouvoir le sélectionner dans le CSS.

⚠ Si vous êtes perdus, allez lire dans le polycopié la section 4.3.2 sur les sélecteurs.

Question 13 (Mise en forme du tableau). Pour finir, on termine la mise en forme du tableau. Pour rappel :

Langage	Expérience	Niveau
Français	30 ans	Débutant
HTML	1h	Expert

→ Pour créer la ligne horizontale, modifiez la propriété `border-bottom` des cases de la première ligne (le **header** de la table).

→ Pour créer les lignes verticales, modifiez les propriétés `border-right` et `border-left` des cellules de la table.

→ Regardez le résultat. Pour ne pas avoir de bordures en double, appliquez la propriété `border-collapse: collapse;` à la table.

→ Améliorez le rendu visuel de la table grâce aux propriétés `text-align`, `padding-right` et `padding-left`.

III) Validation W3C

Question 14 (Validation W3C). Le W3C propose un validateur HTML et CSS en ligne pour vérifier que votre code respecte les standards.

→ Allez sur <https://validator.w3.org/> et choisissez **Validate by File Upload** pour faire valider votre CV. Tentez de comprendre les erreurs affichées et faites les corrections nécessaires, puis recommencez jusqu'à ce qu'il n'y ait plus d'erreurs.

→ Faites la même opération avec votre feuille CSS sur <https://jigsaw.w3.org/css-validator/>.

⚠ Le code HTML et CSS que vous rendrez pour ce TP, ainsi que pour le reste de l'année, devra passer cette validation sans erreur.

IV) Propriété float

Question 15 (Flottants). On voudrait que l'image soit placée sur le côté du texte.

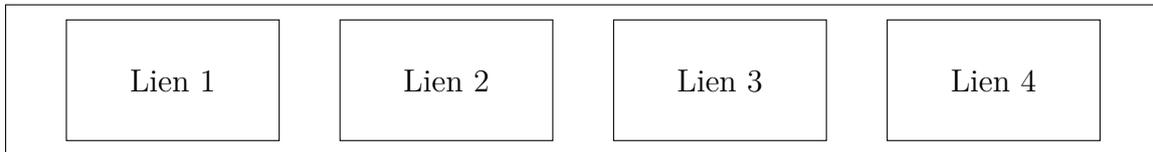
→ Utilisez la propriété CSS `float` avec une valeur `left` ou `right`.

⚠ Réfléchissez bien à quel élément il vaut mieux appliquer ces changements.

→ Si vous voulez rajouter une marge entre l'image et le texte, on peut appliquer la propriété CSS `margin-right` ou `margin-left` à l'image pour lui donner un peu de place.

V) Marges et propriétés display

Question 16 (Menu de navigation). Les liens du menu de navigation ressemblent à des liens normaux. On voudrait entourer chaque lien d'une boîte, et que le lien soit suivi quand on clique dans la boîte - l'apparence habituelle d'un menu de navigation.



→ Modifiez le style des liens pour qu'ils ne soient pas soulignés ni en bleu ; pour créer une boîte visible, donnez-leur une `border` et une `background-color` de votre choix.

⚠ Comme `border` est une propriété raccourcie, elle permet de modifier plusieurs propriétés : `border-width`, `border-style` et `border-color`. N'hésitez pas à regarder la page des propriétés raccourcies pour voir les valeurs possibles.

⚠ Attention au contraste entre la couleur du texte et celle du fond. Ça ne s'applique pas qu'aux Powerpoint !

→ Modifiez leurs propriétés `width` et leur `height` pour faire des boîtes d'une taille agréable. Que se passe-t-il ? Et si vous leur appliquez les propriétés `display:block` ou `display:inline-block` ?

⚠ Pour résumer, la propriété `display` permet de changer le comportement des éléments entre `inline` (texte normal), `block` (bloc normal), `inline-block` (se comporte comme inline mais garde des propriétés de bloc), et `none` (non affiché).

→ Améliorez l'apparence de votre menu en donnant une bordure et une couleur de fond au menu de navigation, et en utilisant les propriétés suivantes :

- `margin` et `padding` dans chaque boîte `a`, pour ajouter de l'espace dans et entre les boîtes ;
- `text-align` pour centrer le menu ;
- `width` et `height` pour fixer la hauteur des boîtes.

⚠ `margin` comme `padding` sont des propriétés raccourcies qui contiennent quatre valeurs : `margin-top`, `margin-bottom`, `margin-left` et `margin-right`. On peut aussi modifier les valeurs individuellement. Essayez de comprendre la différence entre les deux effets et trouvez des valeurs qui vous conviennent.

Question 17 (Pied de page). Dans le pied de page (`footer`), on va rajouter un petit bloc qui indique, par exemple, la technologie utilisée pour créer le site Web. On voudrait un petit bloc centré, d'une couleur différente du fond, tout en bas de la page.

→ Définissez un bloc et ajoutez-y un contenu « Powered by HTML » (ou autre chose). Choisissez une couleur de fond. Donnez-lui une largeur et une hauteur égales, soit absolue (px, em, cm), soit en pourcentage.

→ Le texte passe à la ligne automatiquement. Pour améliorer la mise en page, vous pouvez choisir un alignement (`text-align`) et placer des sauts de lignes `
` à l'endroit que vous préférez. Ajoutez un peu de `padding` à votre goût.

⚠ Encore une fois, il ne faut pas utiliser `
` quand vous êtes en train d'écrire des paragraphes de texte (`<p>`).

→ Donnez à certains de vos blocs (celui-ci ou dans le menu de navigation) des bordures arrondies grâce à la propriété `border-radius`.

→ Pour centrer le bloc, `text-align` ne marche pas car il ne s'agit pas d'un élément inline. À la place, il faut utiliser un autre « truc » : Appliquer au bloc `margin:auto` ; qui place des marges égales des deux cotés du bloc.

⚠ On pourrait vouloir centrer le texte en vertical dans le bloc ; cependant c'est un problème notoirement difficile à résoudre en CSS pur.

VI) Positionnement CSS

Question 18 (Positionnement fixé). Nous allons faire en sorte que l'en-tête de navigation soit toujours visible. C'est une méthode courante pour les sites avec beaucoup de contenu.

→ Appliquez la propriété `position:fixed` au header. Pour voir le résultat, zoomez sur la page avec `Ctrl+` et faites défiler la page.

→ Comme le bloc a été sorti du flot normal, l'espace qui lui était réservé a été enlevé. Pour résoudre ce problème, une méthode est de :

- ajouter de la marge supérieure (`margin-top`) au bloc `body` contenant toute la page, puis ;
- positionner le menu avec `top` dans l'espace libéré.

→ Suivant l'ordre dans lequel les blocs sont introduits, l'en-tête peut passer « sous » certaines images. Ce problème peut être corrigé en lui attribuant un `z-index` de 1 ou plus (le `z-index` correspond à la « hauteur » de l'élément).

Question 19 (Positionnement absolu). Le positionnement absolu (`position:absolute`) permet de sortir un bloc de sa position normale et de le placer à un endroit quelconque de la page. On va l'utiliser pour ajouter une petite image par-dessus votre portrait.

⚠ Il ne faut pas mettre ce positionnement partout, car en plus de rendre les bloc pénibles à placer précisément, cela cassera l'adaptativité du site. On l'utilise souvent pour placer un objet par-dessus un autre.

→ Insérez après votre portrait une image de votre choix - par exemple `quality.png` et redimensionnez-la en CSS pour qu'elle soit assez petite.

→ Appliquez à cette image les propriétés CSS `position:absolute; top:10px; left:20px;` (positionnement à 10 pixels du bord haut et 20 pixels du bord gauche), et regardez le résultat. Modifiez les valeurs de `top` et `left` (ou (`bottom` et/ou `right`)) de sorte que la nouvelle image vienne sur votre portrait.

→ Déplacez la figure (par exemple en lui appliquant `float:right`). Constatez le problème. Pour améliorer la situation :

1. Placez l'image dans la même figure que le portrait ;
2. Appliquez `position:relative` à la figure ;
3. Choisissez des bonnes valeurs pour `top`, `left`, `bottom`, `right` pour placer votre figure.

⚠ La raison pour l'étape 2 est que le positionnement absolu se fait par rapport à l'élément parent si cet élément est en positionnement relatif; sinon, il se fait par rapport à la page entière. Le positionnement relatif sert (en dehors de ce cas-là) à décaler légèrement un élément depuis son emplacement normal, toujours avec `top`, `left`, `bottom`, `right`.

VII) Conception adaptative

Cette partie vous fait utiliser du CSS pour que votre site s'adapte correctement à toutes les tailles d'écran. Nous utiliserons les requêtes `media`, qui sont expliquées dans la partie 4.6.3 du polycopié.

Question 20 (Adaptativité de l'image). Nous allons améliorer la manière dont l'image s'affiche à toutes les tailles.

→ Dans le CSS, donnez à l'image une largeur en pourcentage et utilisez les propriétés `min-width` et `max-width` pour que l'image se redimensionne et reste à une taille appropriée pour toutes les tailles d'écrans.

→ Faites en sorte que, pour les très petits écrans (mobiles), l'image n'ait pas de texte à sa gauche ou à sa droite; autrement dit, désactivez sa propriété `float`.

Question 21 (Requêtes `media`). En utilisant les requêtes `media`, on peut afficher une information différente pour chaque taille.

→ Sous le titre, mettez un texte qui indique :

- “Version mobile” si l’écran fait moins de 640 pixels ;
- “Version tablette” si l’écran est entre 641 et 1000 pixels ;
- “Version bureau” si l’écran fait plus de 1001 pixels.

Indice : Utilisez la propriété CSS `display:none` pour cacher un bloc.

Question 22 (Adaptation de la mise en page). Pour terminer, améliorons l’affichage du menu de navigation.

→ Faites en sorte que les liens dans le menu soient l’un au-dessus de l’autre dans la version mobile et sur la même ligne pour la version tablette et bureau.

→ En utilisant les pourcentages, arrangez les propriétés de `padding` et de `margin` pour que l’affichage soit bon à toutes les tailles.

→ Bonus : comment faire pour que le menu contienne deux lignes de deux liens en version tablette ?

VIII) CV secret

Question 23 (CV secret). On va créer une version « sombre » du CV.

→ Créez une copie de votre CV à l’identique. Dans chaque version, placez un lien qui dirige vers l’autre page quand on clique sur le portrait. Il suffit de placer des balises `<a>` autour de l’image.

→ Créez une nouvelle feuille CSS pour votre deuxième version (ne copiez pas le contenu de la première feuille). Vous lui donnerez un arrière-plan très sombre et du texte clair.

→ Reliez les deux feuilles CSS à la version sombre. Essayez de le faire dans un sens et dans l’autre pour voir celui qui convient.

IX) Bonus

Question 24 (Bonus). Si vous avez terminé le reste du TP, vous pouvez aller vous renseigner sur Flexbox ou Bootstrap et refaire le menu de navigation. Dans ce cas-là, ne modifiez que le CV secret pour que nous puissions voir les deux versions. Vous pourrez utiliser ce que vous avez appris lors du projet.

Rendu : Le site est à déposer sur Moodle sous la forme d’un dossier compressé `S1-NomPrenom.zip` qui contient votre site entier (HTML, CSS, ressources). Votre site doit fonctionner sans modifications. Penser à valider votre site avec le validateur W3C (HTML et CSS) avant le rendu.

Si vous avez le temps, vous pouvez apprendre à mettre votre site en ligne. Les instructions sont sur <https://codin.iut-orsay.fr/>.

<https://validator.w3.org/>

<https://jigsaw.w3.org/css-validator/>