

## Développement Front-End

### LP - AW

### TP 2 (CSS)

L'objectif du TP est de vous faire travailler sur les aspects suivants

- Propriétés css (fonts, couleurs, arrière-plan)
- Adapter votre style aux tableaux, listes, menus
- Factoriser votre code
- Appliquer du style en fonction d'évènements (click, hover, etc.)

---

### Exercice d'application

En modifiant le code debut\_css.html fourni sur Chamilo, ajouter directement dans le code de la page (en section head) les directives CSS nécessaires pour que :

- le texte de toutes les cellules de lignes de classe bestOf dans la table d'identifiant lesMeilleursVendeurs soit affiché en vert (sans changer celles de la table d'identifiant lesMeilleursProduits).
- le texte de toutes les cellules d'une ligne de classe worstOf, quel que soit son parent, s'affiche en rouge.
- le texte de la première ligne ayant la classe bestOf s'affiche en gras sans qu'elle devienne prioritaire sur les directives précédentes, ajouter UNE directive pour que les lignes paires des tableaux aient un texte bleu.

Les développeurs de la page se rendent compte qu'ils ont oublié d'indiquer l'unité monétaire du chiffre d'affaire :

- créer une directive CSS pour rajouter "€" derrière les chiffre d'affaire des vendeurs et des produits (mais pas ceux des services).

Dans le formulaire de contact, rajouter de l'ergonomie en...

- regroupant les champs : ajouter un fieldset contenant mail et téléphone, avec comme légende "Vous répondre", et un second fieldset contenant objet et message, avec comme légende "Votre message" ;
- alignant les libellés à droite et en haut (textarea) sur une petite surface d'affichage, les libellés doivent s'afficher au dessus des champs et non à côté
- centrant le bouton "Envoyer"
- rendant obligatoire les champ "mail" et "message" ; mettez leur couleur de fond en rouge pâle lorsqu'ils sont vides pour les rendre évidents (à faire en UNE directive CSS avec UN sélecteur, qui doit marcher pour tous les champs obligatoires, présents et à venir)

- trouver trois directives différentes (trois types de sélecteurs) pour que le titre de la section d'identifiant "services" soit affiché en jaun

---

## Application à votre site

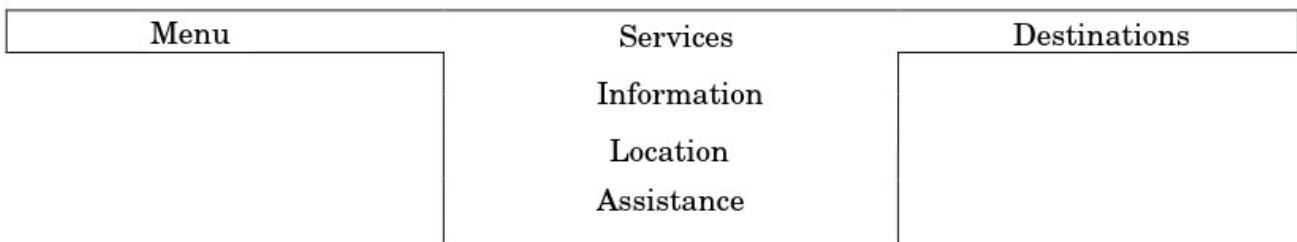
Votre chef de projet est ravi de votre travail sur les pages html du site. Néanmoins, il trouve que cela manque d'esthétique.

- Dans votre répertoire de travail (qui porte votre nom et prénom), créer un nouveau répertoire nommé TP2. Recopiez les pages html du TP1 dans celui-ci.
1. Ajouter des directives CSS à vos pages. Essayez de les rendre à la fois attractives et avec un code efficace (compact, mais maintenable, c'est-à-dire factorisé quand c'est intelligent et commenté...). Vous êtes libre du style, mais vous devez avoir au minimum :
    - Des styles pour les titres
    - Des styles différents pour des rubriques différentes
  2. Reprenez le tableau dans votre page "destinations/animaux/joueurs" afin de lui appliquer les styles suivant :
    - Une bordure
    - Une couleur de fond pour les cellules d'entête.
    - Une couleur de fond qui change toutes les deux lignes.
    - Un effet différent (changement de couleur de la ligne) lors du passage de la souris.
    - Une taille uniforme pour toutes les images dans votre tableau
    - Vous pouvez consulter l'image "exemple tableau.png" disponible sur Chamilo à titre d'exemple sur la mise en forme souhaitée.
  3. Reprenez votre menu afin qu'il apparaisse sur chaque page. Ce menu doit
    - Être placé en haut de chaque page, occupant toute la largeur, et avoir un fond d'une couleur qui masque ce qui est derrière lui
    - Il doit rester à cette place même si l'on fait défiler la page (le menu doit rester visible)
    - Avoir l'option active (le lien Accueil lorsque la page d'accueil est affichée !)
    - Visuellement identifiable.
    - Etudiez ensuite comment « automatiser » ce rendu visuel en ayant pour le menu le même code HTML et les mêmes directives CSS dans toutes les pages (l'idée : peut-on externaliser le menu et son style pour les factoriser ?)

- Vous pouvez vous inspirer de la structure HTML suivante :

```
<div class="menuItem" id="identifiantDuMenu1">
  Entête du menu 1
  <ul>
    <li class="menuElement"><a href="...">1er élément du menu1</a></li>
    <li class="menuElement"><a href="...">2ième élément du menu1</a></li>
    ...
  </ul>
</div>
<div class="menuItem" id="identifiantDuMenu2">
  Entête du menu 2
  <ul>
    <li class="menuElement"><a href="...">1er élément du menu2</a></li>
    <li class="menuElement"><a href="...">2ième élément du menu2</a></li>
    ...
  </ul>
</div>
```

4. Menu déroulant : transformer le menu statique du haut du CV en menu déroulant en utilisant uniquement du CSS. Exemple :



Votre code CSS ne doit pas dépendre du nombre d'options ou de sous-options du menu et ne doit pas être spécifique pour une option donnée (en clair : pas de directive dépendant d'un id).

5. Proposez un menu vertical pour la partie connexion, recherche, réservation. Celui-ci doit apparaître à gauche de chaque page, dès qu'on passe la souris sur l'onglet menu du menu horizontal (image ci-dessus).

*N.B : tous les styles doivent être écrits en CSS ! (balises <b>, <i>, ... interdites). Les éléments de même nature doivent pouvoir être modifiés en un seul changement dans les directives CSS. Par exemple, toutes les images doivent dépendre de la / des mêmes directives CSS. De même, les destinations doivent pouvoir être modifiées toutes d'un seul changement dans le CSS. A l'inverse, des rubriques différentes (activités et services) peuvent avoir le même rendu mais doivent pouvoir être stylisées différemment en ne modifiant QUE le CSS.*