

Table des matières

Mise en situation.....	3
Séquence 1 : la page d'accueil (balises de base).....	4
Objectifs.....	4
Moyens disponibles.....	4
Travail préliminaire.....	4
Contraintes.....	4
Structure.....	5
Travail demandé.....	7
Séquence 2 : la page produits (les tableaux).....	10
Objectifs.....	10
Structure.....	10
Travail demandé.....	11
Séquence 3 : la page contact (les formulaires).....	13
Objectifs.....	13
Structure.....	13
Contraintes.....	14
Travail demandé.....	15
Séquence 4 : la charte graphique (feuille de style CSS).....	16
Objectifs.....	16
Contraintes.....	16
Screenshot.....	17
Travail demandé.....	17
Séquence 5 : la page services (multimédia).....	18
Objectifs.....	18
Screenshot.....	18
Travail demandé.....	19
Annexe 1 : Mise en page des documents web.....	20
Annexe 2 : les balises pour le multimédia.....	28
Annexe 3 : quelques formats d'image.....	30
Annexe 4 : quelques formats vidéos.....	31

D'autres ressources utiles sont disponibles à cette adresse : <http://tvaira.free.fr/>

Mise en situation

La société SFCV (Spécialiste Français n°1 des Caves à Vin) désire mettre en place un site web.

La planification de la réalisation du site sera la suivante :

◆ étape n°1 : réalisation des pages HTML

Cette étape a pour but de découvrir et de prendre en main le langage *HTML* qui est interprété côté client par le navigateur. L'objectif principal de cette étape est la réalisation des pages **statiques** du site : menu, affichage des résultats, formulaires pour une demande de catalogue, effectuer une recherche, etc ...

◆ étape n°2 : mise en place de l'interactivité côté client

Cette étape a pour but de découvrir et de prendre en main le langage *JavaScript* qui s'exécute côté client (donc par le navigateur). L'objectif principal de cette étape est d'assurer l'intégrité des données envoyées vers le serveur par les formulaires. En effet, il serait inutile de surcharger le serveur avec l'envoi de données manquantes, incomplètes ou incorrectes. Cette vérification peut donc se faire côté client par le langage *JavaScript*. D'autre part, ce langage permet d'améliorer l'interactivité côté client.

◆ étape n°3 : réalisation des pages dynamiques

Cette étape a pour but de découvrir et de prendre en main le langage *PHP* qui s'exécute côté serveur. Les objectifs principaux de cette étape sont :

- de traiter les données reçues par les différents formulaires et de les insérer dans la base de données par exemple
- d'interroger la base de données et de fournir un résultat adapté aux requêtes clientes.

Séquence 1 : la page d'accueil (balises de base)

Objectifs

Cette première séquence a pour objectif de réaliser la première page HTML du site. Elle met en œuvre :

- la structure des pages en utilisant quelques balises de base
- les liens vers les autres pages (menu)
- l'utilisation des listes
- l'insertion d'image pour le logo de l'entreprise

Moyens disponibles

Un éditeur de texte pour la réalisation de la page HTML

Un navigateur (*Firefox* ou *Chrome* par exemple) pour visualiser des pages HTML réalisées

Travail préliminaire

Créer un répertoire **cave** pour le stockage des fichiers du site. Dans le répertoire **cave**, créer un sous-répertoire **images** dans lequel on stockera toutes les images nécessaires pour le site. Les pages HTML porteront l'extension **.html**.

Contraintes

A . Pour la réalisation des pages HTML de ce site, on utilisera **HTML 5** en spécifiant la déclaration de type de document suivante :

```
<!DOCTYPE html>
```

B . Pour la réalisation des pages HTML de ce site, on utilisera l'encodage de caractères **UTF-8** :

```
<meta charset="utf-8">
```

Ou :

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Remarques : Il est important de comprendre qu'il faut obtenir une cohérence entre l'encodage utilisé pour créer le document (l'éditeur de texte) et celui pour l'interpréter (le navigateur) :

- Les outils d'édition (par exemple, un éditeur de texte) peuvent coder des documents HTML avec l'encodage de caractères de leur choix. Ces outils peuvent utiliser tout encodage commode lequel couvre la plupart des caractères contenus dans le document, pourvu que l'encodage soit correctement étiqueté. Les caractères occasionnels, qui ne sont pas contenus dans cet encodage, peuvent tout de même être représentés par des références de caractères (numérique ou entité). Le choix de l'encodage se fait généralement au moment de l'enregistrement du fichier.
- Le paramètre « charset » identifie un encodage de caractères, qui représente une méthode pour convertir une séquence d'octets en une séquence de caractères. Cette conversion s'intègre naturellement au système de l'activité du Web : les serveurs envoient des documents HTML aux agents utilisateurs sous la forme d'un flux d'octets, les agents utilisateurs les interprètent comme séquence de caractères.

Exemples de références de caractères :

- < → < (permet d'éviter la confusion avec la balise ouvrante)
- > → > (permet d'éviter la confusion avec la balise fermante)
- © → ©

D . On précisera la langue "fr" en attribut dans la balise racine **html**.

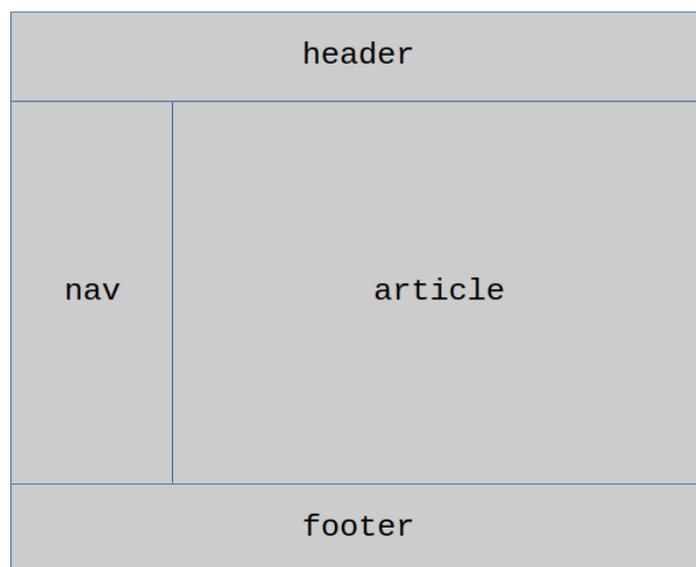
C . Respecter les noms de fichier indiqués dans le TP.

Remarques : dans tous les cas, ne pas utiliser d'espace, de majuscules et de caractères spéciaux (pas d'accents notamment) dans les noms de fichiers.

Structure

Le premier travail est de définir une structure à son document (le **fond**) et de ne pas se focaliser sur son apparence dans le navigateur (la **forme**). La mise en forme du document est une tâche qui sera réalisée plus tard (en séquence 4) notamment grâce à l'utilisation des feuilles de style et en respectant une charte graphique.

Dans cette première séquence, ce travail est important car les choix faits seront réutilisés dans les autres pages. En quelque sorte, cette première page va servir de *squelette* pour les autres. On va définir la structure suivante pour les documents :



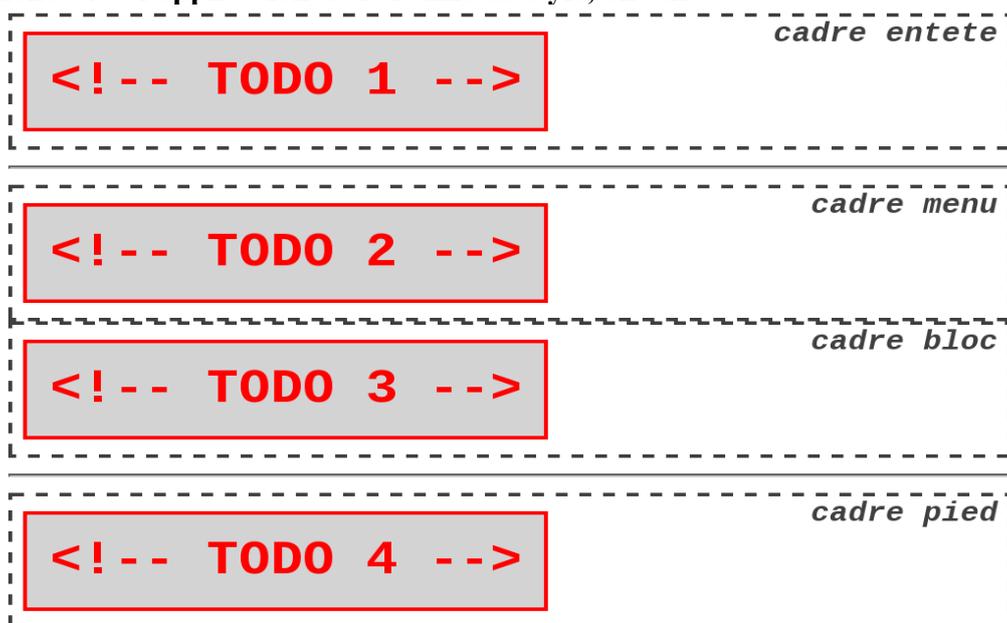
Le squelette de la page **index.html** sera donc le suivant :

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title><!--TODO--></title>
  </head>
  <body>
    <header><!--TODO--></header>
    <hr>
    <nav><!--TODO--></nav>
    <article><!--TODO--></article>
    <hr>
    <footer><!--TODO--></footer>
  </body>
</html>
```

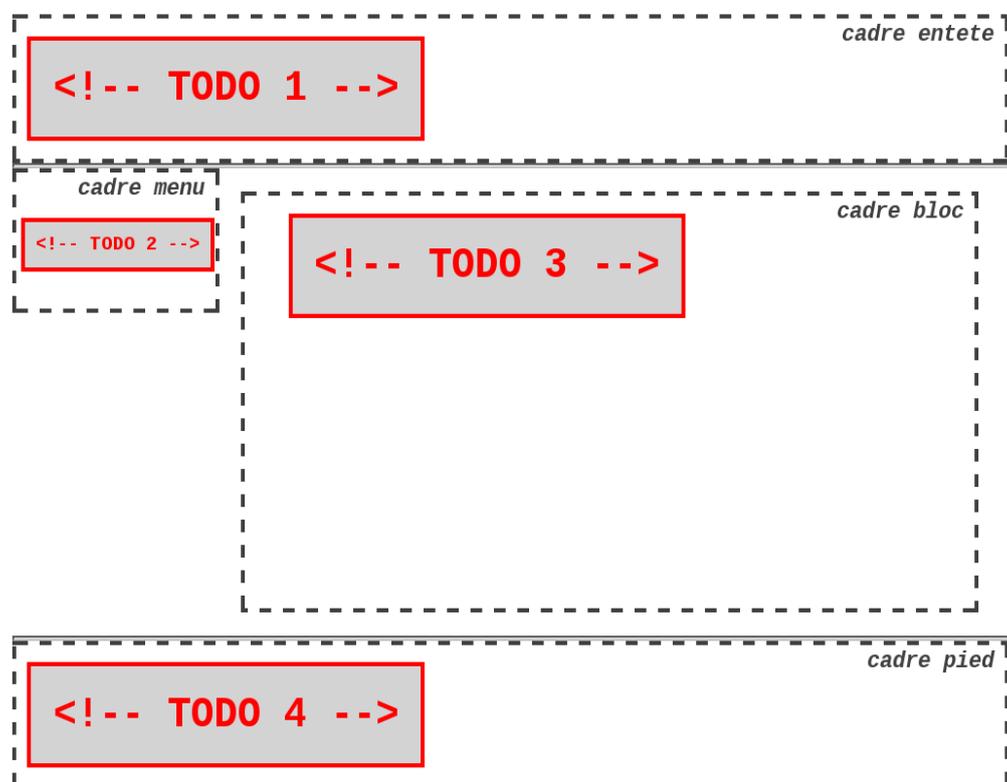
Remarque : Les balises `<!--TODO-->` sont des **commentaires** en HTML. Ils indiquent ici les parties à réaliser (TODO) dans les questions du travail demandé.

Lire l'annexe 1 sur la mise en page des sites web.

Avant la séquence 4 et l'application de la feuille de style, on aura :



Après la séquence 4 et l'application de la feuille de style, on obtiendra :



Travail demandé

Documentation : <https://html.spec.whatwg.org/multipage/>

Voici un aperçu des balises de base utilisées dans cette séquence :

cadre entete



- balise

SFCV

- balise <h1>

N°1 des caves à vins

- balise <h3>

- balise <hr>

cadre menu

- balise
- balise <a> - balise

cadre bloc

Spécialiste français n°1 des caves à vin

- balise <h2>

Déclaration des droits de la bouteille

- balise <p>

- balise
- balise
- balise
- balise

Avec une gamme complète commercialisée par les plus grands distributeurs nationaux, notre société vous propose son savoir-faire incomparable pour concevoir des caves à vin de très haute qualité.

- balise
 - balise <p>

Liens

- balise <h2>

La passion du vin au féminin : [www.femivin.com](#)

Vinosafe : [www.vinosafe.com](#)

cadre pied

© SFCV - [Mentions légales](#)

L'abus d'alcool est dangereux pour la santé, à consommer avec modération

Ceci définit la **structure** de notre page web.

1. Titre (balise <title>)

→ Créer la page d'accueil **index.html** demandée (voir contraintes) en précisant le titre du document :
SFCV : Spécialiste Français n°1 des Caves à Vin

2. Le bloc article : en-têtes (balises <h>), paragraphes (balises <p> et
) et listes (balises , et)

On tiendra compte des indications suivantes :

- l'en-tête « Spécialiste français n°1 des caves à vin » est de type <h2>
- on utilisera des paragraphes (<p>) dont un utilise une liste numérotée ().

→ Modifier le document **index.html** (dans le cadre *bloc* <!--TODO3-->) qui reproduit la figure ci-dessous :

Spécialiste français n°1 des caves à vin

Déclaration des droits de la bouteille

1. Article 1 : Quelle que soit l'origine, on a le droit au respect
2. Article 2 : Quelle que soit la couleur, on a le droit à un abri sûr
3. Article 3 : Quelle que soit l'âge, on a le droit à s'épanouir
4. Article 4 : Quelle que soit le lieu, on a le droit à s'intégrer

Avec une gamme complète commercialisée par les plus grands distributeurs nationaux, notre société vous propose son savoir-faire incomparable pour concevoir des caves à vin de très haute qualité.

Attention : vous avez tendance à « abuser » des balises
 pour respecter la mise en page demandée. Ne faites pas ça ! Et conservez seulement les sauts de lignes nécessaires dans le texte d'un paragraphe. On vous rappelle que la présentation du document à l'écran sera réalisée à la fin par l'application d'un style (CSS). Pour l'instant, consacrez vous à la structure de vos pages plus qu'à leur visualisation.

3. Les liens absolus et relatifs (balise <a>)

On tiendra compte des indications suivantes :

- l'en-tête « Liens » est de type <h2>
- on déclarera deux paragraphes pour chaque lien créé
- le lien "Mentions légales" pointera sur un document **mentions.html**

→ Modifier le document **index.html** (dans le cadre *bloc* <!--TODO3--> et *pied* <!--TODO4-->) qui reproduit la figure ci-dessous :

Liens

La passion du vin au féminin : www.femivin.com

Vinosafe : www.vinosafe.com

© SFCV - [Mentions légales](#)

L'abus d'alcool est dangereux pour la santé, à consommer avec modération

4. Les images (balise)

On tiendra compte des indications suivantes :

- l'en-tête pour le nom de la société sera de type <h1>
- on utilisera un autre en-tête de type <h3> pour " N°1 des ... "
- le chemin du fichier image est : **images/logo.png** (on ajoute un texte alternatif pour cette image)

→ Modifier le document **index.html** (dans le cadre *entete* <!--TODO1-->) qui reproduit la figure ci-dessous :



SFCV

N°1 des caves à vins

5. Le menu (nav)

On tiendra compte des indications suivantes pour créer le menu de gauche :

- on utilisera une liste non numérotée de type `` donc chaque élément de liste `` contiendra un élément (lien) du menu
- le lien " Accueil " pointera sur un document **index.html**
- le lien " Produits " pointera sur un document **produits.html**
- le lien " Services " pointera sur un document **services.html**
- le lien " Contact " pointera sur un document **contact.html**

Remarque : ces choix correspondent au menu « classique » d'une société commerciale.

→ Modifier le document **index.html** (dans le cadre *menu* `<!--TODO2-->`) qui reproduit la figure ci-dessous :



SFCV
N°1 des caves à vins

- [Accueil](#)
- [Produits](#)
- [Contact](#)

Spécialiste français n°1 des caves à vin

Déclaration des droits de la bouteille

1. Article 1 : Quelle que soit l'origine, on a le droit au respect
2. Article 2 : Quelle que soit la couleur, on a le droit à un abri sûr
3. Article 3 : Quelle que soit l'âge, on a le droit à s'épanouir
4. Article 4 : Quelle que soit le lieu, on a le droit à s'intégrer

Avec une gamme complète commercialisée par les plus grands distributeurs nationaux, notre société vous propose son savoir-faire incomparable pour concevoir des caves à vin de très haute qualité.

Liens

La passion du vin au féminin : www.femivin.com

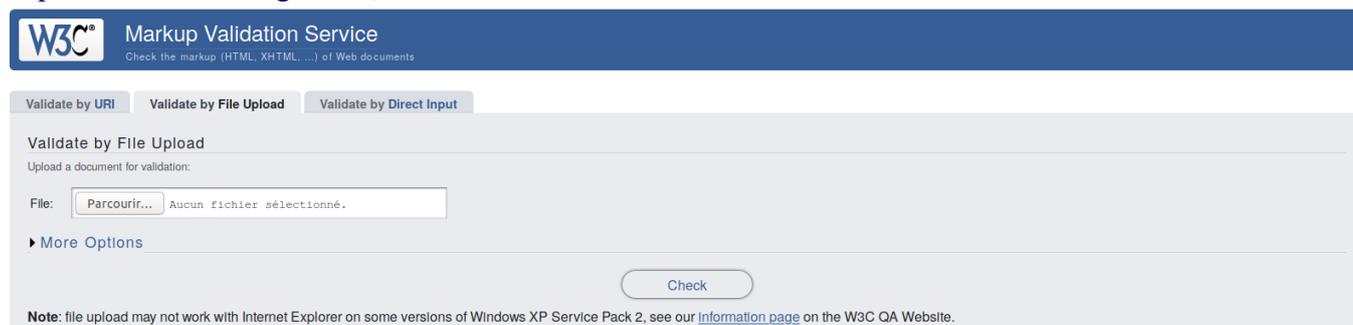
Vinosafe : www.vinosafe.com

© SFCV - [Mentions légales](#)

L'abus d'alcool est dangereux pour la santé, à consommer avec modération

6. Validation

→ Vous devez valider la page **index.html** en utilisant le service en ligne du W3C : <http://validator.w3.org/>. Puis, tester les liens externes.



W3C Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI | **Validate by File Upload** | Validate by Direct Input

Validate by File Upload

Upload a document for validation:

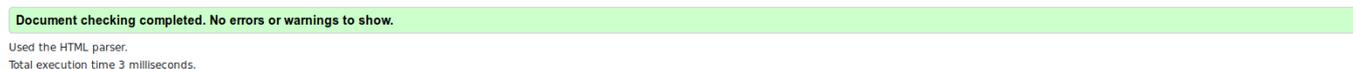
File: Aucun fichier sélectionné.

► More Options

Check

Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our [information page](#) on the W3C QA Website.

On doit obtenir :



Document checking completed. No errors or warnings to show.

Used the HTML parser.
Total execution time 3 milliseconds.

Séquence 2 : la page produits (les tableaux)

Objectifs

Cette séquence a pour objectif de réaliser la page HTML qui permet de lister l'ensemble des produits commercialisés par la société SFCV. Elle met en oeuvre l'utilisation des **tableaux**. Pour naviguer plus facilement dans le document, on utilisera aussi des **liens internes** (avec des ancres).

Structure

Évidemment, on retrouve la même structure, seul le *bloc* central droit change de contenu :



SFCV
N°1 des caves à vins

- [Accueil](#)
- [Produits](#)
- [Contact](#)

Caves de vieillissement

AV 175	Pratique, le thermomètre en façade et le nouveau registre de cave international pour tenir à la croix la gestion des bouteilles présentes. Confortable les clayettes collecteur réversibles à empreintes en bois massif.		
	Contenance	Système froid	Dimensions (h x l x p)
	178 (en 75cl)	Compresseur	144 x 82 x 87
	Accessoires : 3 Clayettes réversibles et 1 Modulable (en option payante), Régulation électromécanique, Afficheur Digital de température		Noir

AV 205	Astucieux, sa lampe de lecture pour lire les étiquettes dans un environnement obscur.		
	Contenance	Système froid	Dimensions (h x l x p)
	186 (en 75cl)	Compresseur	128 x 70 x 88
	Accessoires : Serrure à clé, 1 Clayette coulissante (possibilité de supplémentaires en option payante), Régulation électronique, Dynamique DataDisplay		Noir

[Remonter](#)

Caves polyvalentes

CV 182	Élegante, la poignée de porte en bois massif.		
	Contenance	Système froid	Dimensions (h x l x p)
	170 (en 75cl)	Compresseur	125 x 70 x 87
	Accessoires : Serrure à clé, Clayettes bois		Brun foncé

Bientôt de nouvelles caves disponibles ...

© SFCV - [Mentions légales](#)
L'abus d'alcool est dangereux pour la santé, à consommer avec modération

On va utiliser des tableaux pour la présentation des produits de la société.

Travail demandé

1. Titre (balise <title>)

→ Créer la page d'accueil **produits.html** demandée en précisant le titre du document :
SFCV : les caves à vin

→ Copier/Coller la structure du fichier **index.html** réalisé à la séquence 1. Seul le cadre *article* sera personnalisé pour cette page.

2. Tableaux (balises <table>, <tr>, <th> et <td>)

a. Pour commencer simplement avec les tableaux, on va réaliser un document **tarifs.html** (qui servira plus tard à la séquence 5) qui reproduit la figure ci-dessous :

Modèle	Tarif
AV 175	<i>veuillez nous contacter</i>
AV 205	<i>veuillez nous contacter</i>
CV 183	<i>veuillez nous contacter</i>

Remarque : la page **tarifs.html** n'utilisera pas la structure *header*, *nav* et *footer*. Le tableau sera juste placé dans le corps de la page (balise *body*).

b. Pour la mise en page des produits, on tiendra compte des indications suivantes :

- les différentes catégories de caves seront de type <h2>
- on utilisera le fusionnement de cellules (colspan et rowspan) et on intégrera une image dans le tableau

→ Modifier le document **produits.html** qui reproduit la figure ci-dessous (images/AV175.jpg) :

Caves de vieillissement				
AV 175	Pratique, le thermomètre en façade et le nouveau registre de cave International pour tenir à la craie la gestion des bouteilles présentes. Confortable les clayettes collector réversibles à empreintes en bois massif.			
	Contenance	Système froid	Dimensions (hxlxp)	Porte
	178 (en 75cl)	Compresseur	144 x 62 x 67	Pleine
	Accessoires : 2 Clayettes réversibles et 1 Modulable (en option payante), Régulation électromécanique, Afficheur Digital de température			Noir

Attention : n'utilisez pas les balises
 pour respecter la mise en page demandée. On vous rappelle que la présentation du document à l'écran sera réalisée à la fin par l'application d'un style (CSS). Pour l'instant, consacrez vous à la structure de vos pages plus qu'à leur visualisation.

Même chose pour ces deux autres produits :

- **AV 205** (images/AV205.jpg)
- Astucieux, sa lampe de lecture pour lire les étiquettes dans un environnement obscur.
- 196 (en 75cl) / Compresseur / 139 x 70 x 68 / Pleine / Noir
- Accessoires : Serrure à clé, 1 Clayette coulissante (possibilité de supplémentaires en option payante), Régulation électronique, Dynamique DataDisplay

et une cave polyvalente :

- **CV 183** (images/CV183.jpg)
- Élégante, la poignée de porte en bois profilé.
- 170 (en 75cl) / Compresseur / 125 x 70 x 67 / Pleine / Brun foncé
- Accessoires : Serrure à clé, Clayettes bois

3 . Les liens internes et les ancres (balise <a>)

On réalisera les opérations suivantes :

- créer une ancre en haut du document
- placer un lien " Remonter " entre chaque catégorie et qui pointera sur l'ancre

Exemple :

```
<a name="caves_v"></a>
<h2>Caves de vieillissement</h2>

<!-- ... -->

<a href="#caves_v">Remonter aux caves de
vieillissement</a>
```

Crée une ancre nommée caves_v

Crée un lien vers l'ancre nommée caves_v

→ Modifier le document **produits.html** pour intégrer au moins deux liens internes.

4 . Validation et test

→ Vous devez valider la page **produits.html** en utilisant le service en ligne du W3C : <http://validator.w3.org/>. Puis, tester les liens internes.

Séquence 3 : la page contact (les formulaires)

Objectifs

Cette séquence a pour objectif de réaliser la page HTML qui permet de demander un catalogue gratuit à la société SFCV. Elle met en oeuvre l'utilisation des **formulaires**.

Structure

Une nouvelle fois, on retrouve la même structure, seul le *bloc* central droit change de contenu :



SFCV

N°1 des caves à vins

- [Accueil](#)
- [Produits](#)
- [Contact](#)

Demande de catalogue

Demandez-nous vite notre catalogue gratuit !
Il vous sera livré dans les meilleurs délais.

Civilité

Nom

Prénom

Adresse

Ville Code postal :

je souhaite recevoir la newsletter - courriel :

Ce site a fait l'objet d'une déclaration auprès de la Commission Nationale Informatique et Libertés (CNIL). Conformément aux dispositions de la loi Informatiques et libertés du 6/01/1978, vous disposez d'un droit de modification, de rectification et d'opposition/suppression des données vous concernant, ce par simple demande à : info@sfcv.com

N'hésitez pas à prendre contact avec l'équipe SFCV. Nous répondrons à toutes vos questions sur l'aménagement de votre cave ou le choix de votre armoire à vin. Tous les conseils de professionnels pour conserver vos meilleurs vins en toute sécurité pour le plaisir de les déguster en toute convivialité.

© SFCV - [Mentions légales](#)

L'abus d'alcool est dangereux pour la santé, à consommer avec modération

Contraintes

A. Les données du formulaire :

Les données du formulaire seront envoyées à un destinataire précisé par l'attribut **action** de la balise **<form>**. On peut indiquer, dans l'attribut **action**, soit une adresse email (action="<mailto:@email>") soit l'url d'une page (le plus souvent vers un CGI ou un script PHP).

Toutes les données des éléments du formulaire doivent être nommées (attributs **name** et **id**) afin d'être identifiables par le programme de traitement destinataire (notion de variables : le nom donné dans le formulaire correspondra au nom de la variable dans le programme serveur) ou même par le destinataire du *mail*. On n'acceptera pas les accents et on n'utilisera que les minuscules pour les attributs **name** et **id**.

D'autre part, le concepteur du formulaire doit souvent fixer les valeurs par défaut associées aux éléments (attribut **value**) :

- le formulaire : name="demande", id="demande"
- la civilité : name="civilité" – value="1" pour Monsieur, etc ...
- le nom : name="nom" – value="" (saisie de l'utilisateur)
- le prénom : name="prenom" – value="" (saisie de l'utilisateur)
- l'adresse : name="adresse" (saisie de l'utilisateur)
- la ville : name="ville" – value="" (saisie de l'utilisateur)
- le code postal : name="codepostal" – value="" (saisie de l'utilisateur)
- choix de recevoir la newsletter : name="newsletter" – value="oui"
- son adresse email : name="email" – value="" (saisie de l'utilisateur)
- l'envoi du formulaire : name="action" – value="Envoyer"

Remarque : l'utilisateur a la possibilité de donner son adresse *email* s'il veut recevoir la lettre d'information (*newsletter*)

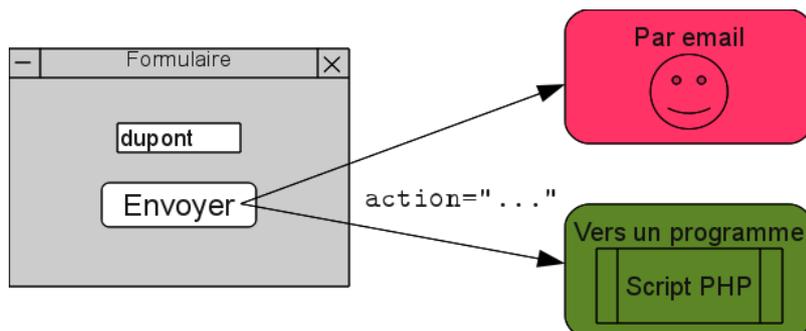
B. L'envoi du formulaire :

En utilisant l'attribut **method**, on distingue deux méthodes (GET ou POST) d'envoi des données du formulaire :

- Avec method=**POST**, les données envoyées au serveur se trouvent dans le **corps de la requête HTTP**.
- Avec method=**GET**, les données envoyées au serveur se trouvent dans l'**URL** précisée dans l'attribut **action** (cela donnera par exemple *url?month=fevrier&day=24*).

Il faut aussi indiquer la manière avec laquelle les données du formulaire seront encodées en renseignant l'attribut **enctype** (*text/plain, multipart/form-data, ...*).

Il y a en général deux destinations d'envoi possibles (la plus utilisée étant celle vers un programme) :



Travail demandé

1. Titre (balise <title>)

→ Créer la page d'accueil **contact.html** demandée en précisant le titre du document :
■ SFCV : Contact

2. Formulaire (balises <form>, <input> et <textarea>)

→ Réaliser la page demandée en respectant les contraintes énoncées ci-dessus.

3. Validation et test

→ Vous devez valider la page **contact.html** en utilisant le service en ligne du W3C :
<http://validator.w3.org/>.

→ Tester unitairement l'envoi du formulaire pour chacune des balises **<form>** suivantes :

a . `<form action="mailto:votreadresse@xxx.fr" name="contact" method="POST" enctype="text/plain">`
remplacer `votreadresse@xxx.fr` par votre adresse email si vous voulez recevoir les données dans votre boîte

b . `<form action="test.html" name="contact" method="GET" enctype="text/plain">`

c . `<form action="test.html" name="contact" method="POST" enctype="text/plain">`

d . `<form action="http://tvaira.free.fr/report.php" name="report" method="POST">`
`<form action="http://tvaira.free.fr/report.php" name="report" method="GET">`

→ Fournir les résultats obtenus.

Séquence 4 : la charte graphique (feuille de style CSS)

Objectifs

Cette séquence a pour objectif d'appliquer une charte graphique à l'ensemble des pages HTML réalisées. Elle met en oeuvre l'utilisation des feuilles de style CSS (*Cascading Style Sheets*). Le principe est simple : il s'agit de redéfinir le rôle d'une balise html en lui imposant de nouvelles propriétés.

Contraintes

A. Fichier externe unique

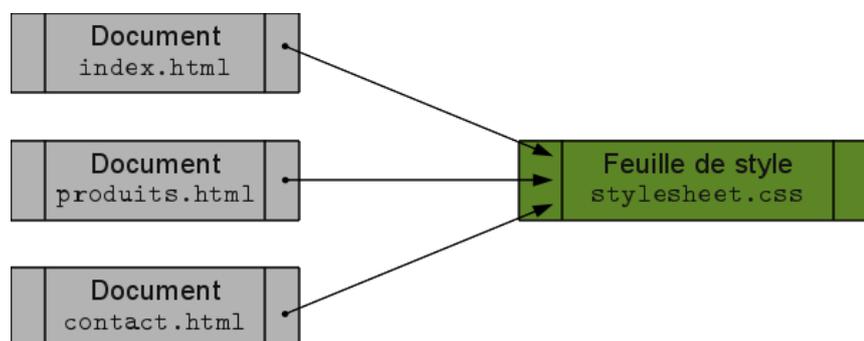
Les **feuilles de styles CSS** améliorent la qualité des mises en page sur le web (positionnement des éléments au pixel près), diminuent le temps d'affichage et factorisent le code, le rendant ainsi plus lisible et plus facile à maintenir. Le simple changement d'une feuille de styles permet de modifier la présentation du document. Dans le fichier CSS, il faut aussi faire la séparation entre structure et décoration :

- la **structure** contiendra le positionnement des éléments dans la (ou les) page(s) **ET**
- la **décoration** s'occupera notamment de la couleur et du graphisme

Remarque : en pratique cela pourrait donner deux fichiers CSS : un pour la structure, et un autre pour la présentation. Dans notre cas, on utilisera qu'un SEUL fichier dans lequel on placera de grandes barres de commentaires pour séparer les différentes parties (*/* ... */*). Dans tous les cas, il est conseillé faire la séparation.

On utilisera donc un fichier externe unique **stylesheet.css** pour redéfinir le style des balises utilisées dans les pages HTML du site. Ce fichier CSS ne doit pas contenir de code HTML. Chaque page du site devra donc indiquer qu'elle utilise ce fichier CSS de la manière suivante :

```
<head>
  <title>Le titre de la page</title>
  <link rel="stylesheet" href="stylesheet.css" type="text/css">
</head>
```



B. La charte graphique

Les séquences précédentes nous ont amenés à définir une structure aux documents créés. Il reste à définir, grâce aux feuilles de style, l'interprétation visuelle faite par le navigateur.

On découpera ce travail en deux parties :

- la structure visuelle des éléments : dimensions, positionnement (flottant, absolu) et leurs relations
- la décoration visuelle des éléments : police, couleur, ...

L'objectif est d'obtenir une cohérence (graphique) pour l'ensemble des pages et de faciliter les tâches de maintenance (modifications).

B. Quelques éléments de structure à respecter :

- Taille et couleur de la police : 12px blanche (pour tout le document)
- Taille et couleur de la police : 22px grise (pour l'élément h1)
- Taille et couleur de la police : 16px blanche (pour l'élément h2)
- Taille et couleur de la police : 14px grise (pour l'élément h3)
- Taille et couleur de la police : 10px grise (pour l'élément h3 du bloc *entete*)
- Modification du texte : en majuscules (pour l'élément h3 du bloc *entete*)
- Taille de la bordure (gauche) : 2px (pour le bloc *bloc*)
- Type de la bordure (gauche) : solid (pour le bloc *bloc*)
- Couleur de la bordure (gauche) : grise (pour le bloc *bloc*)

→ Récupérer le fichier http://tvaira.free.fr/web/src/stylesheet2_todo.css et le renommer en **stylesheet.css**.

Screenshot

La page d'accueil **index.html** (un exemple de ce que l'on peut obtenir grâce à la structure définie avant) :



The screenshot displays a web browser window with the developer tools console open. The website content includes a navigation menu with 'Accueil', 'Produits', 'Services', and 'Contact'. The main heading is 'Spécialiste français n°1 des caves à vin'. Below this, there is a section titled 'Déclaration des droits de la bouteille' with four bullet points. The footer contains the text '© SFCV - Mentions légales' and 'L'abus d'alcool est dangereux pour la santé, à consommer avec modération'. The developer tools console shows the HTML structure, including the doctype, the root element with lang='fr', and the body containing header, nav, article, and footer elements.

Travail demandé

- 1 . Compléter le fichier **stylesheet.css** fourni en respectant les contraintes énoncées ci-dessus.
- 2 . Mettre au point sa feuille de style en utilisant la console d'erreurs des outils de développement de votre navigateur (Firefox ou Chrome)
- 3 . Valider sa feuille de style en utilisant l'outil de validation du W3C : <http://jigsaw.w3.org/css-validator/>
- 4 . Intégrer, dans toutes les pages du site, la feuille de style CSS créée.
- 5 . Pour terminer, réaliser la page **mentions.html**. Le contenu est disponible dans le fichier mentions.txt. Utiliser des ancres pour naviguer facilement dans cette page.

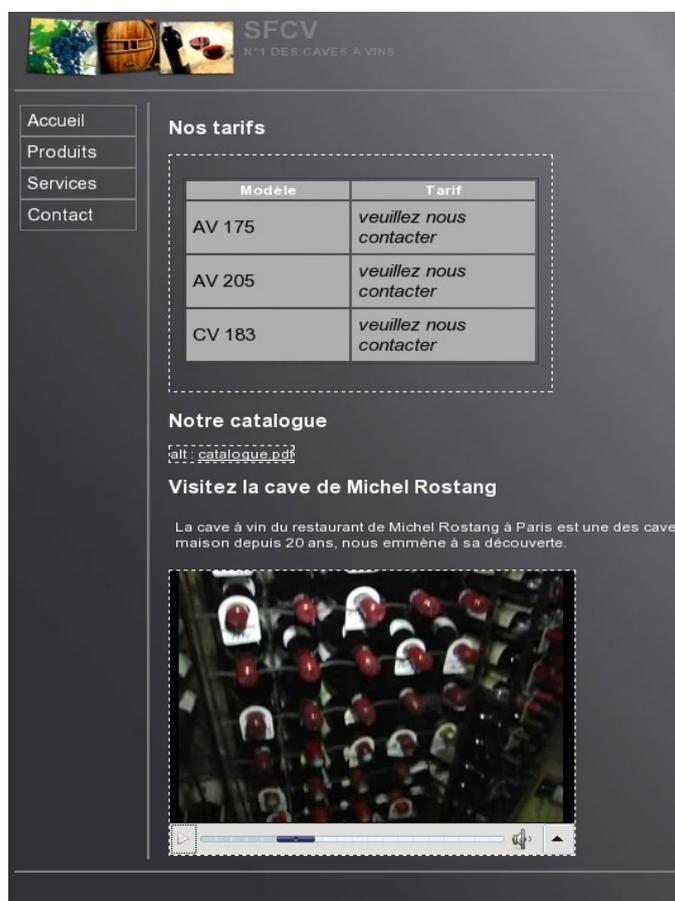
Séquence 5 : la page services (multimédia)

Objectifs

Cette séquence a pour objectif d'intégrer des ressources multimédias dans une page web. Elle met en oeuvre l'utilisation de balises décrites dans l'annexe 2.

Screenshot

La page d'accueil `services.html` :



Lire l'annexe 2 sur les balises multimédia et les exemples fournis.

Travail demandé

1. Titre (balise <title>)

→ Créer la page d'accueil **services.html** demandée en précisant le titre du document :
■ SFCV : les services

2. Contenu multimédia (balise <object>)

→ Réaliser la page demandée (voir *screenshot*).

La page (le cadre *bloc*) contiendra trois parties :

- Nos tarifs (titre h2) : incorporation dans la page du fichier **tarifs.html** (réalisé à la séquence 2)
- Notre catalogue (titre h2) : incorporation du catalogue au format PDF (**catalogue.pdf**)
- Visite d'une cave (titre h2) : incorporation d'une vidéo au format MP4 (**visite.mp4**)

3. Validation et test

→ Vous devez valider la page **services.html** en utilisant le service en ligne du W3C :
<http://validator.w3.org/check>.

→ Tester la visualisation de la vidéo.

Il est conseillé de lire les annexes sur les formats d'image et vidéo.

Annexe 1 : Mise en page des documents web

Tout d'abord, il faut savoir que le langage HTML d'origine ne prévoyait pas véritablement d'éléments pour assurer sa mise en page. Donc différentes techniques utilisées par les *webmasters* se sont succédées :

- les cadres (*frames*)
- les tableaux
- et maintenant les feuilles de styles CSS (et la balise conteneur `<div>`)

Les cadres (*frames*)

Les cadres découpent la fenêtre principale en autant de petits cadres, chacun d'eux jouant le même rôle qu'une fenêtre : un document HTML propre avec barres de défilement indépendantes...

Ces cadres peuvent être chargés de manière totalement indépendante : un cadre sera modifié et l'autre maintenu. Ils peuvent interargir entre eux : un choix dans un premier cadre pourra provoquer le chargement d'une nouvelle page dans un deuxième.

Les cadres permettent donc de créer rapidement une mise en page simple pour un site. L'utilisation classique consiste ainsi à présenter sur un cadre à gauche une liste de liens vers les différentes parties d'un site, qui s'affichent dans la fenêtre principale au centre. Cela facilite de plus la maintenance des différentes pages.

Cependant, les moteurs de recherche indexent les pages à l'intérieur des cadres autant que les autres. Il y a donc un risque pour qu'une page indexée de la sorte soit "orpheline", et ne donne pas accès au reste du site. Plus généralement, une telle page ne se suffit souvent pas à elle-même (il y manque parfois les outils de navigation à l'intérieur du site). Enfin, la présence de cadres s'accompagne souvent de celles de barres de défilement horizontales.

Il faut savoir qu'au début des *frames* sur le Web, leur utilisation posait toutes sortes de problèmes. Par exemple, il n'existait aucun moyen de masquer les bordures séparant les différentes zones dans la fenêtre du navigateur, ce qui fait que l'écran était rempli de traits gris et de barres de défilement. Dans la mesure où la place est limitée sur une page Web, on comprend que l'utilisateur ne se réjouisse pas de voir la fenêtre du navigateur ainsi remplie. C'est l'une des raisons pour lesquelles les *frames* ont (eu) mauvaise réputation. Aujourd'hui pourtant, il est possible de rendre invisibles les bordures entre *frames*.

Remarque: les responsables de sites ont constaté que la plupart des internautes choisissent la version sans *frame* lorsqu'elle est proposée.

Avantages

Pour le *webmaster*: gestion et maintenance des pages plus simple et plus rapide

Pour l'internaute: sommaire toujours présent à l'écran, permettant de naviguer plus rapidement d'une page à l'autre sans revenir en arrière.

Inconvénients

Pour le *webmaster*:

- ◆ Pages orphelines (chaque page HTML composant un *frameset* peut être appelée directement par son adresse URL, indépendamment de son environnement *frameset*).
- ◆ Se retrouver avec des *frames* dans les *frames* (un site à base de *frames* peut appeler un autre site à base de *frames* etc ...)

Pour l'internaute:

- ◆ Sauvegarde ou impression d'une page problématique
- ◆ Non accessibilité à l'information pour les aveugles et mal-voyants. (<http://www.brailenet.jussieu.fr/education/livreblanc/>)
- ◆ Espace réduit pour le contenu
- ◆ Complexité de la navigation: les *frames* peuvent constituer un véritable labyrinthe, notamment quand le *webmaster* a concocté un *frameset* encapsulant des sites externes, ou des sommaires en cascades.

Exemple

```
<HTML>
  <HEAD>
    <TITLE>Un document avec deux cadres verticaux</TITLE>
  </HEAD>
  <FRAMESET cols="25%,75%" frameborder="0">
    <FRAME src="menu.html" name="menu">
    <FRAME src="accueil.html" name="principal">
  </FRAMESET>
</HTML>
```

Les tableaux

Originellement prévus dans le but de mettre en forme des données, les tableaux ont rapidement été détournés par les développeurs web pour combler un vide qui faisait cruellement défaut au HTML à l'époque : la mise en page.

En effet, dans ce qu'il convient d'appeler la préhistoire du web, point de CSS ni de séparation entre le contenu et la présentation. Les développeurs pensaient avoir trouvé la parade avec les tableaux, et ont donc commencé à utiliser massivement ces derniers pour du positionnement et autres mises en page complexes.

Les tableaux sont franchement inadaptés aux exigences d'aujourd'hui, à cause de leurs limites notamment en termes d'accessibilité, de facilité d'utilisation et de maintenance.

Remarque : il est important de noter que les tableaux remplissent parfaitement leur rôle quand il s'agit de structurer des données.

Inconvénients

- ◆ Lenteur d'affichage : un tableau est un élément complexe à traiter, notamment pour définir les tailles de chaque cellule et ligne, et nécessite d'analyser toutes les cellules ... surtout si les tableaux sont imbriqués
- ◆ Page illisible pour les navigateurs non-graphique (ne pas oublier l'existence de navigateurs spécialisés pour les handicapés, avec plage braille ou synthétiseur vocal, qui ne peuvent pas restituer les informations graphiques)
- ◆ Impression problématique : changer complètement le *design* d'une page pour permettre une meilleure impression est impossible ce qui amène souvent à faire une page séparée pour l'impression, et avec les coûts associés à sa création et à sa maintenance, et cela se transforme en un gâchis d'énergie et de ressources considérables.
- ◆ Complexe et coûteux à produire (même avec des éditeurs WYSIWYG)
- ◆ Lourdeur des pages : beaucoup de balises pour peu de contenu !

Les feuilles de style CSS (Cascading Style Sheets)

Les feuilles de style CSS permettent la mise en forme du contenu des pages Web.

Avantages

Les **feuilles de styles CSS** améliorent la qualité des mises en page sur le web (positionnement des éléments au pixel près), diminuent le temps d'affichage et factorisent le code, le rendant ainsi plus lisible et plus facile à maintenir. Le simple changement d'une feuille de styles permet de modifier la présentation du document. Une seule feuille de style permet d'appliquer la présentation d'un site web.

Syntaxe

Cet exemple basique devrait vous permettre de comprendre l'utilisation des feuilles de style.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta content="text/html; charset=UTF-8" http-equiv="content-type">
    <title>Exemples</title>
    <link rel="stylesheet" href="style.css" type="text/css">
  </head>
  <body>
    <h3>Application d'un style global aux balises body, h1, h2 et p :</h3>
    <div><code>body { color: #4B0082; }<br>
    h1 { font-family: Arial; font-size: 22px; color: blue; }<br>
    h2 { font-family: Arial; font-size: 16px; color: green; }<br>
    p &nbsp;{ font-family: Arial; font-size: 12px; color: red; }<br>
    </code></div>
    <h1>&lt;h1&gt;Je suis un titre de niveau 1&lt;/h1&gt;</h1>
    <h2>&lt;h2&gt;Je suis un titre de niveau 2&lt;/h2&gt;</h2>
    <p>&lt;p&gt;Je suis un paragraphe&lt;/p&gt;</p>
    <h1>&lt;h1&gt;Je suis un autre titre de niveau 1&lt;/h1&gt;</h1>
    <hr>
    <h3>Application d'un style individuel (id) aux balises h1, h2 et p :</h3>
    <div><code>
    #id_h1 { font-family: Courier; color: black; } /* on ne touche pas à la taille */<br>
    #id_h2 { font-family: Courier; color: black; }<br>
    #id_p &nbsp;{ font-family: Courier; color: black; }<br>
    </code></div>
    <h1 id="id_h1">&lt;h1 id="id_h1"&gt;Je suis un titre de niveau 1 (avec un
id)&lt;/h1&gt;</h1>
    <h2 id="id_h2">&lt;h2 id="id_h2"&gt;Je suis un titre de niveau 2 (avec un
id)&lt;/h2&gt;</h2>
    <p id="id_p">&lt;p id="id_p"&gt;Je suis un paragraphe (avec un id)&lt;/p&gt;</p>
    <h1 id="id_h1">&lt;h1 id="id_h1"&gt;Je suis un autre titre de niveau 1 (sans id)&lt;/h1&gt;</h1>
    <div><em>Remarque : h1, h2 et p ont hérité de la taille (font-size) définie globalement pour ces
balises.<br>Attention, l'id doit être défini et utilisé qu'une fois, car un id(entifiant) est toujours
unique.</em></div>
    <hr>
    <h3>Application d'une classe de style (class) commune aux balises h1, h2 et p :</h3>
    <div><code>
    .my_class { font-family: Tahoma; font-size: 16px; } /* on ne touche pas à la couleur */<br>
    </code></div>
    <h1 class="my_class">&lt;h1 class="my_class"&gt;Je suis un titre de niveau 1 (avec
my_class)&lt;/h1&gt;</h1>
    <h2 class="my_class">&lt;h2 class="my_class"&gt;Je suis un titre de niveau 2 (avec
my_class)&lt;/h2&gt;</h2>
    <p class="my_class">&lt;p class="my_class"&gt;Je suis un paragraphe (avec
my_class)&lt;/p&gt;</p>
    <h1 class="my_class">&lt;h1 class="my_class"&gt;Je suis un autre titre de niveau 1 (sans my_class)&lt;/h1&gt;</h1>
    <div><em>Remarque : h1, h2 et p ont hérité de leur couleur (color) définie globalement pour ces
balises.</em></div>
    <hr>
  </body>
</html>
```

La feuille de style **style.css** :

```
/* toutes les balises body, h1, h2 et p auront ce style */
body { color: #4B0082; }
h1 { font-family: Arial; font-size: 22px; color: blue; }
h2 { font-family: Arial; font-size: 16px; color: green; }
p { font-family: Arial; font-size: 12px; color: red; }
code { font-family: Courier; }

/* les balises identifient leur style individuel et unique (id → #) */
#id_h1 { font-family: Courier; color: black; } /* on ne touche pas à la taille */
#id_h2 { font-family: Courier; color: black; }
#id_p { font-family: Courier; color: black; }

/* pour toutes les balises utilisant cette classe de style (class → .) */
.my_class { font-family: Tahoma; font-size: 16px; } /*on ne touche pas à la couleur*/
```

Ce qui donne :

Application d'un style global aux balises **body**, **h1**, **h2** et **p** :

```
body { color: #4B0082; }
h1 { font-family: Arial; font-size: 22px; color: blue; }
h2 { font-family: Arial; font-size: 16px; color: green; }
p { font-family: Arial; font-size: 12px; color: red; }
```

<h1>Je suis un titre de niveau 1</h1>

<h2>Je suis un titre de niveau 2</h2>

<p>Je suis un paragraphe</p>

<h1>Je suis un autre titre de niveau 1</h1>

Application d'un style individuel (id) aux balises **h1**, **h2** et **p** :

```
#id_h1 { font-family: Courier; color: black; } /* on ne touche pas à la taille */
#id_h2 { font-family: Courier; color: black; }
#id_p { font-family: Courier; color: black; }
```

<h1 id="id_h1">Je suis un titre de niveau 1 (avec un id)</h1>

<h2 id="id_h2">Je suis un titre de niveau 2 (avec un id)</h2>

<p id="id_p">Je suis un paragraphe (avec un id)</p>

<h1>Je suis un autre titre de niveau 1 (sans id)</h1>

*Remarque : h1, h2 et p ont hérité de la taille (font-size) définie globalement pour ces balises.
Attention, l'id doit être défini et utilisé qu'une fois, car un id(entifier) est toujours unique.*

Application d'une classe de style (class) commune aux balises **h1**, **h2** et **p** :

```
.my_class { font-family: Tahoma; font-size: 16px; } /* on ne touche pas à la couleur */
```

<h1 class="my_class">Je suis un titre de niveau 1 (avec my_class)</h1>

<h2 class="my_class">Je suis un titre de niveau 2 (avec my_class)</h2>

<p class="my_class">Je suis un paragraphe (avec my_class)</p>

<h1>Je suis un autre titre de niveau 1 (sans my_class)</h1>

Remarque : h1, h2 et p ont hérité de leur couleur (color) définie globalement pour ces balises.

Exemple

```
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <title>Exemple - CSS</title>
  <link rel="stylesheet" href="./style.css" type="text/css">
</head>
<body>
  <header>
    L'en-t&ecirc;te ...
  </header>

  <div id="left">
    Le c&ocirc;nt&eacute; gauche ...
  </div>

  <div id="right">
    Le c&ocirc;nt&eacute; droit ...
  </div>

  <div id="middle">
    Le milieu ...
  </div>

  <footer>
    Le pied de page ...
  </footer>
</body>
</html>
```

La feuille de style `style.css` :

```
/* Structure */

body
{
  margin: 10px;
  padding: 0px;
}

header
{
  margin: 0px;
  padding: 1%;
  clear: both;
  height: 5%;
}

div#left
{
  margin: 0px;
  padding: 1%;
  float: left;
  width: 20%;
  height: auto;
}

div#right
{
```

```
    margin: 0px;
    padding: 1%;
    float: right;
    height: 80%;
    width: 10%;
}

div#middle
{
    float: left;
    width: 60%;
    margin: 1%;
    padding: 1%;
}

footer
{
    margin: 0px;
    padding: 1%;
    height: auto;
    clear: both;
}

/* Décoration */

header
{
    border: 2px dashed #FF8822;
    color: #FF8822;
}

div#left
{
    border: 2px dashed rgb(51, 204, 255);
    color: rgb(51, 204, 255);
}

div#right
{
    border: 2px dashed #2288FF;
    color: #2288FF;
}

div#middle
{
    border: 2px dashed blue;
    color: blue;
}

footer
{
    border: 2px dashed red;
    color: red;
}
```

On obtient :



Liens pour en savoir plus

Spécification HTML 5 : <https://html.spec.whatwg.org/multipage/>

Spécification DOM : <http://xmlfr.org/w3c/TR/REC-DOM-Level-1/>

Spécification HTML 4 : <http://www.la-grange.net/w3c/html4.01/>

Spécification XHTML : <http://www.la-grange.net/w3c/xhtml1/>

Spécification CSS2 : <http://www.yoyodesign.org/doc/w3c/css2/cover.html>

Les standards du Web: <http://www.openweb.eu.org/>

CSS pratique: <http://pompage.net/pompe/csspratique/>

Annexe 2 : les balises pour le multimédia

L'élément **OBJECT**

L'élément HTML **<object>** représente une ressource externe qui peut être interprétée comme une image, un contexte de navigation imbriqué ou une ressource à traiter comme un *plugin*.

Les deux attributs importants sont (au moins un attribut **data** et un attribut **type** doivent être définis) :

- **data** : l'adresse de la ressource, une URL valide..
- **type** : le type MIME de la ressource définie par **data**.

Incorporer un document HTML

```
<object data="datas/test.html" type="text/html" width="300" height="200"> alt : <a href="datas/test.html">test.html</a></object>
```

Incorporer un document pdf

```
<object data="datas/tpHTML.pdf" type="application/pdf" width="300" height="200"> alt : <a href="datas/tpHTML5.pdf">tpHTML5.pdf</a></object>
```

L'élément **EMBED**

L'élément HTML **<embed>** représente aussi un point d'intégration pour une application externe ou pour du contenu interactif (autrement dit, pour un *plugin*).

Remarque : Il faut garder à l'esprit que la plupart des navigateurs ont dépréciés voire retirer la prise en charge des plugins. Aussi, mieux vaut ne pas utiliser sur <embed> si on souhaite qu'un site ou une application soit fonctionnelle sur un navigateur moyen.

L'élément **AUDIO**

L'élément HTML **<audio>** est utilisé afin d'intégrer un contenu sonore dans un document. Il peut contenir une ou plusieurs sources audio représentées avec l'attribut **src** ou l'élément **<source>**. S'il y a plusieurs sources, le navigateur choisira celle qui convient le mieux.

```
<figure>
  <figcaption>Écoutez Kaamelott</figcaption>
  <audio controls src="datas/mortel.mp3" width="200" height="20">
    Votre navigateur ne supporte pas la balise <code>audio</code>.
  </audio>
</figure>
```

L'élément **VIDEO**

L'élément HTML **<video>** intègre un contenu vidéo dans un document.

Remarque : Les navigateurs ne prennent pas en charge l'ensemble des formats vidéo et il est possible de fournir plusieurs sources grâce à des éléments <source>, le navigateur utilisera la première ressource dont il connaît le format.

Les attributs importants sont :

- **autoplay** : un booléen qui indique que la vidéo doit automatiquement être lancée dès qu'elle peut être jouée sans être arrêtée par le chargement des données.
- **controls** : si cet attribut est présent, le navigateur affichera des contrôles pour permettre à l'utilisateur de contrôler la lecture de la vidéo, le volume et la mise sur pause.

- **loop** : un booléen, qui, lorsqu'il est présent, indique que la vidéo doit être jouée en boucle.

Lien : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/video>

Incorporer un document vidéo

```
<video controls width="320" height="255">
  <source src="datas/test.mp4" type="video/mp4">
  <source src="datas/test.ogv" type="video/ogg">
  <source src="datas/test.webm" type="video/webm">
  <p>Désolé, votre navigateur ne prend pas en charge les vidéos intégrées.</p>
</video>
```

Des exemples sont aussi disponibles sur :

- HTML4 : <http://tvaira.free.fr/web/src/objects.html>
- HTML5 : <http://tvaira.free.fr/web/src/multimedia.html>

Annexe 3 : quelques formats d'image

PNG (Portable Network Graphics)

C'est un format ouvert d'images numériques, qui a été créé pour remplacer le format GIF, à l'époque propriétaire et dont la compression était soumise à un brevet. Le PNG est un format non destructeur spécialement adapté pour publier des images simples. Il a été normalisé par l'ISO.

PNG est une spécification pour Internet et l'objet d'une Recommandation W3C et d'une RFC. Il a été créé pour contourner la licence existante sur le format GIF, le plus en vogue à la fin des années 1990, CompuServe refusant de laisser utiliser son format gratuitement. PNG a alors été défini mais en augmentant les capacités de GIF.

GIF (Graphics Interchange Format)

C'est un format d'image numérique couramment utilisé sur le web.

GIF a été mis au point par CompuServe en 1987 pour permettre le téléchargement d'images en couleur. Ce format utilise l'algorithme de compression sans perte. En 1989, le format GIF a été étendu pour permettre le stockage de plusieurs images dans un fichier. Ceci permet de créer des diaporamas ou des animations.

JPEG (JPEG File Interchange Format ou JFIF)

C'est le format de fichier le plus utilisé pour contenir les images enregistrées avec la compression JPEG. L'extension de fichier .jpeg ou .jpg indique un fichier au format JFIF.

Le format JFIF est complémentaire du GIF et du PNG pour la publication d'image sur le World Wide Web. Bien que plus intéressante pour des photographies ou des images lourdes, la compression JPEG provoque une perte d'information (algorithme de compression destructif) pouvant aboutir à une perte de qualité visible si l'utilisateur privilégie un taux de compression élevé, particulièrement lorsque l'image contient des changements nets de couleur ou peu de couleurs (par exemple des logos, captures d'écran, diagrammes, ...). Le format JFIF ne gère ni les animations ni la transparence.

Actuellement une société détentrice de certains brevets tente une grande vague de procès contre les éditeurs de logiciels qui utilisent la compression JPEG.

WebM

WebM est un format multimédia ouvert qui a été lancé par Google (après rachat de la société On2 Technologies). L'utilisation est en libre et gratuite.

Lien : <http://fr.wikipedia.org/wiki/WebM>

SVG (Scalable Vector Graphics)

SVG est un format de données conçu pour décrire des ensembles de graphiques vectoriels et basé sur XML. Ce format est spécifié par le World Wide Web Consortium.

Le SVG est très utilisé dans le monde de la cartographie et du téléphone portable.

Lire : <http://fr.wikipedia.org/wiki/Svg>

Annexe 4 : quelques formats vidéos

Actuellement, les formats les plus répandus sont : WebM, MP4 et Ogg Theora. Vous pouvez ffmpeg utiliser **ffmpeg** (<http://ffmpeg.org/>) pour réaliser les conversions de format.

H.264/MP4

H.264 est supporté par le *Moving Picture Experts Group*. C'est un format non-libre (soumis à brevets) et non-gratuit. Toutefois, il est gratuit dans certaines utilisations (la diffusion gratuite de vidéos par des sites Web par exemple).

Les fichiers MP4 utilisant le *codec* H.264 sont lisibles nativement sur les navigateurs Apple (Safari, Safari Mobile), Opera, Google Chrome et Firefox.

Remarque : Il est recommandé de placer ce format en premier dans une liste car il est supporté par les iPhone, iPad et autres iPod.

Lien : <https://fr.wikipedia.org/wiki/H.264>

OGG/Theora

Theora est un format de compression vidéo open-source, sans brevets. Ceci donne le droit à tous d'utiliser Theora (à des fins non commerciales tout comme à des fins commerciales) sans devoir payer de redevance au consortium MPEG.

OGG/Theora est lisible sur Firefox, Opera, et Google Chrome.

Lien : <http://fr.wikipedia.org/wiki/Theora>

WebM/VP8

WebM est un format multimédia ouvert qui a été lancé par Google (après rachat de la société On2 Technologies). L'utilisation est en libre et gratuite.

Lien : <http://fr.wikipedia.org/wiki/WebM>

MPEG (Moving Picture Experts Group)

C'est un groupe de travail du comité technique de l'ISO et de la CEI pour les technologies de l'information. Ce groupe d'experts est chargé du développement de normes internationales pour la compression, la décompression, le traitement et le codage de la vidéo, de l'audio et de leur combinaison, de façon à satisfaire une large gamme d'applications.

MPEG a développé notamment les normes suivantes :

- MPEG-1 : première norme audio et vidéo utilisé plus tard pour les Vidéo CD. Elle comprend le populaire format audio MPEG-1 partie 3 audio couche 3 (MP3).
- MPEG-2 : norme applicable au codage de l'audio et la vidéo, ainsi que leur transport pour la télévision numérique : télévision numérique par satellite, télévision numérique par câble, télévision numérique terrestre, et (avec quelques restrictions) pour les vidéo-disques DVD ou SVCD. C'est notamment le format utilisé jusqu'à présent pour la TV sur ADSL.
- MPEG-4 : norme qui permet, entre autres, de coder des objets vidéo/audio, le contenu 3D et inclut le DRM. La partie 2 de MPEG-4 a connu du succès grâce à la mise en application DivX ainsi que dans les téléphones mobiles.

AVI (Audio Video Interleave)

C'est un format de fichier conçu pour stocker des données audio et vidéo. Il a été présenté par Microsoft en novembre 1992, en tant qu'élément de la vidéo pour la technologie de Windows.

Dans un fichier AVI, chaque composante audio ou vidéo peut être compressée par n'importe quel codec.

Le format DivX est souvent utilisé comme codec vidéo, et le format mp3 comme codec audio, mais d'autres codecs peuvent également être utilisés, par exemple XviD ou MPEG pour la vidéo, et mp2, WAV etc. pour l'audio.

MOV (QuickTime)

Un fichier QuickTime contient une ou plusieurs pistes, chacune comporte un type de données particuliers : audio, vidéo, effet ou texte (pour des sous-titres par exemple). Ces fichiers ont le plus souvent l'extension .mov.

QuickTime est un framework multimédia développé par Apple. QuickTime est utilisé par de nombreuses personnes pour la lecture en streaming, ce qui en fait un des logiciels majeurs pour cet usage.

WMV (Windows Media Video)

C'est le nom d'une famille de codecs vidéo propriétaires développé par Microsoft. Sur Internet, il est (était) fréquent de rencontrer ce type de fichiers vidéo, que ce soit en téléchargement ou en streaming.

FLV (Flash Video)

C'est un format de fichier utilisé sur Internet pour diffuser des vidéos via le lecteur Adobe Flash Player versions 6, 7, 8, 9 et 10, de manière à pouvoir l'incorporer aux animations Flash. Ce format est notamment utilisé par les sites de partage de vidéos sur Internet (youtube par exemple).