

HTML5



HTML Tutorial

IMAGES, AUDIO, VIDEO

HTML Tutorial

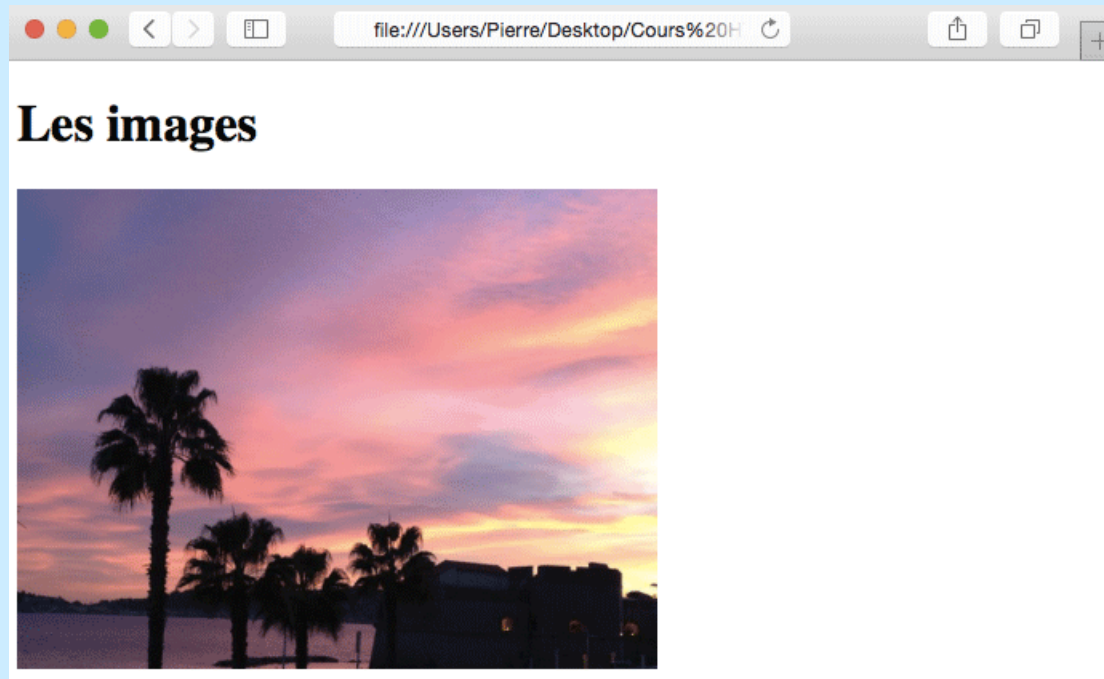
LES IMAGES EN HTML ET EN CSS

Redimensionner une image en CSS

- Après avoir vu comment insérer une image grâce à la balise **img**, il s'agit maintenant d'apprendre à ajuster et positionner cette image dans la page web grâce au CSS.
- Pour redimensionner une image en CSS, on utilise les propriétés **width** (pour la largeur) et **height** (pour la hauteur).
- Une petite astuce ici : en ne modifiant que l'un des deux paramètres de l'image, le second va se mettre automatiquement à l'échelle.

```
img{  
  width: 400px;  
}
```

Redimensionner une image en CSS



Positionner une image en CSS

- L'élément `img` est un élément HTML de type **inline**. Cela signifie que les images, si elles en ont la place, vont se positionner à côté d'autres contenus HTML.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les images</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

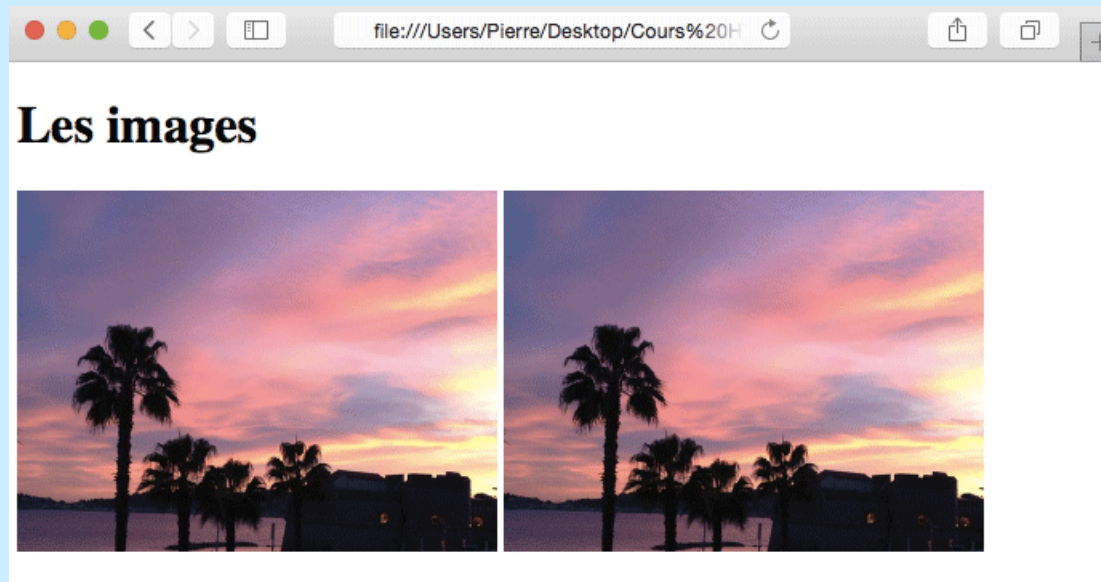
  <body>
    <h1>Les images</h1>

    
    
  </body>
</html>
```

Positionner une image en CSS

- (Suite)

```
img{  
  width: 300px;  
}
```



Positionner une image en CSS

- Souvent, lorsque l'on crée un site, on voudra changer ce comportement par défaut et faire en sorte que les images se positionnent sur leur propre ligne.
- Il y a trois manières simples de faire cela :
 - soit en encadrant notre image avec un élément de type block,
 - soit en utilisant la propriété display,
 - soit en utilisant les propriétés CSS float et clear.
- Première méthode, on peut déjà commencer par placer nos images au sein d'un élément div ou p, tous deux de type block :

Positionner une image en CSS

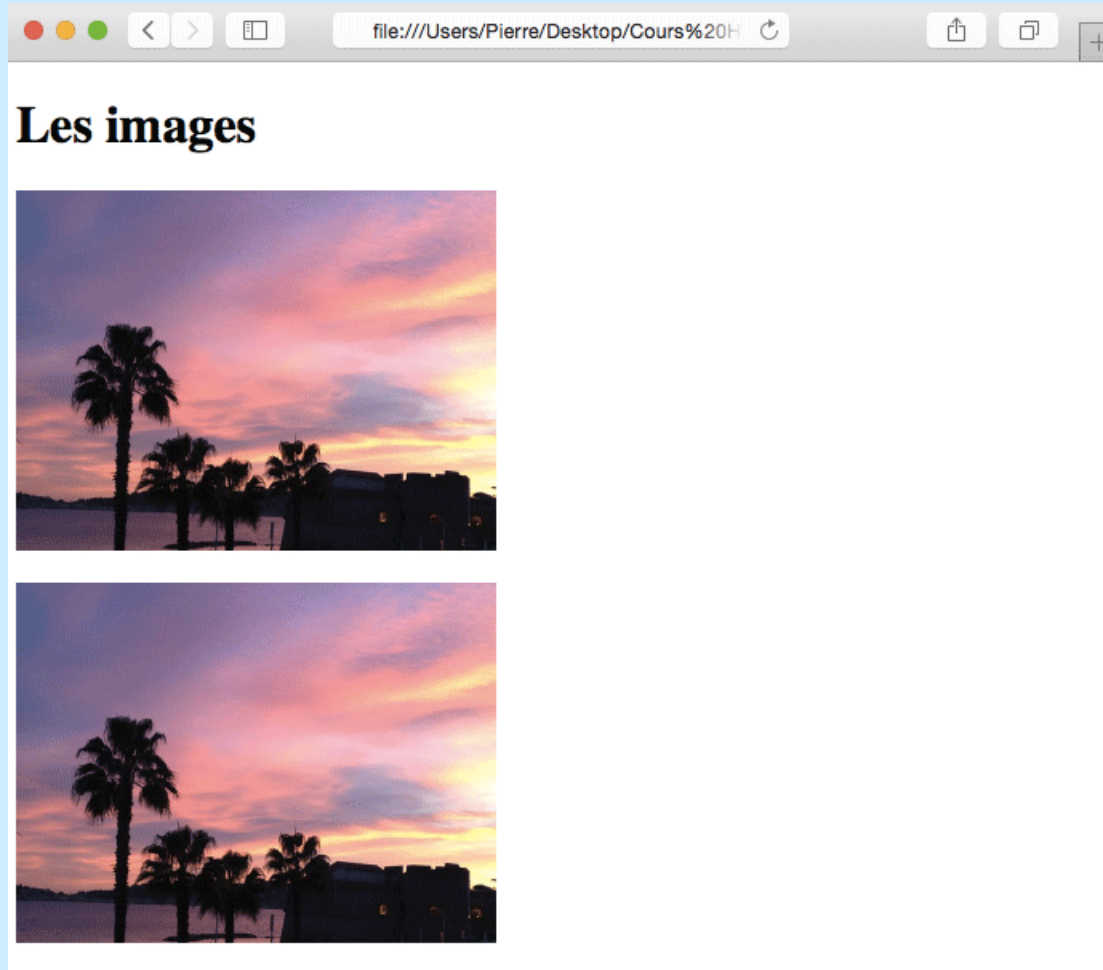
- Première méthode, plaçons les images au sein d'un élément div ou p, tous deux de type block :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les images</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>Les images</h1>

    <p></p>
    <p></p>
  </body>
</html>
```

Positionner une image en CSS



Positionner une image en CSS

- Deuxième méthode, en utilisant la propriété display :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Les images</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

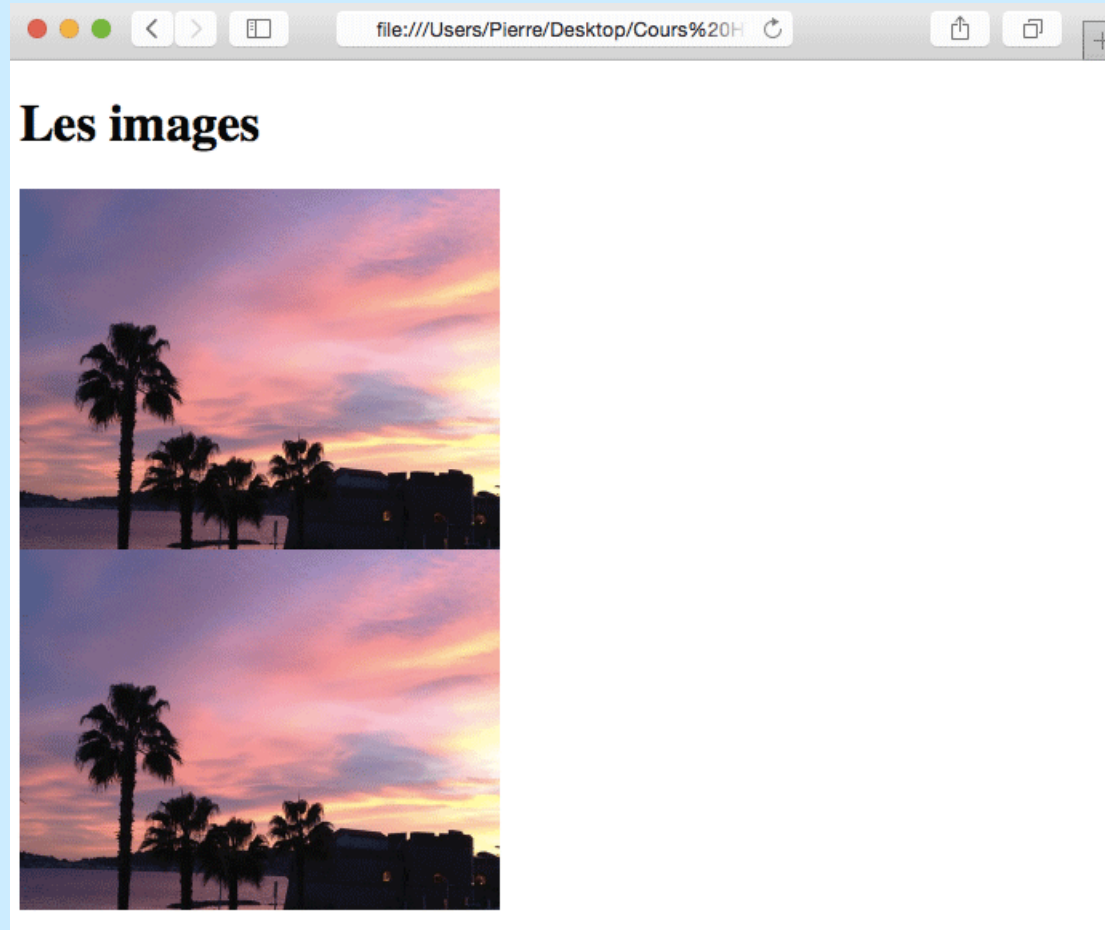
  <body>
    <h1>Les images</h1>

    
    
  </body>
</html>
```

```
img{
  width: 300px;
  display: block;
}
```

Positionner une image en CSS

- (Suite)

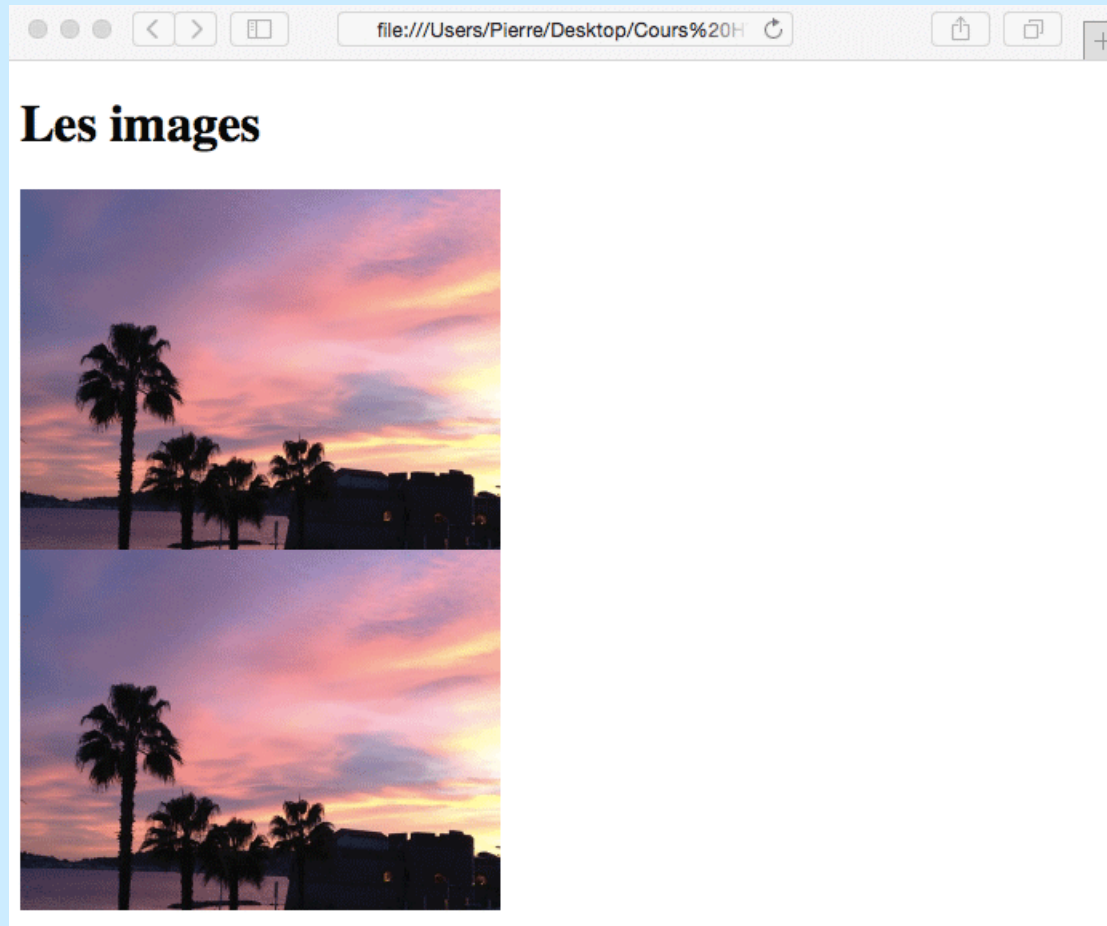


Positionner une image en CSS

- Dernière méthode, en utilisant les propriétés float et clear :

```
img{  
  width: 300px;  
  float: left;  
  clear: both;  
}
```

Positionner une image en CSS



Créer un cadre autour des images

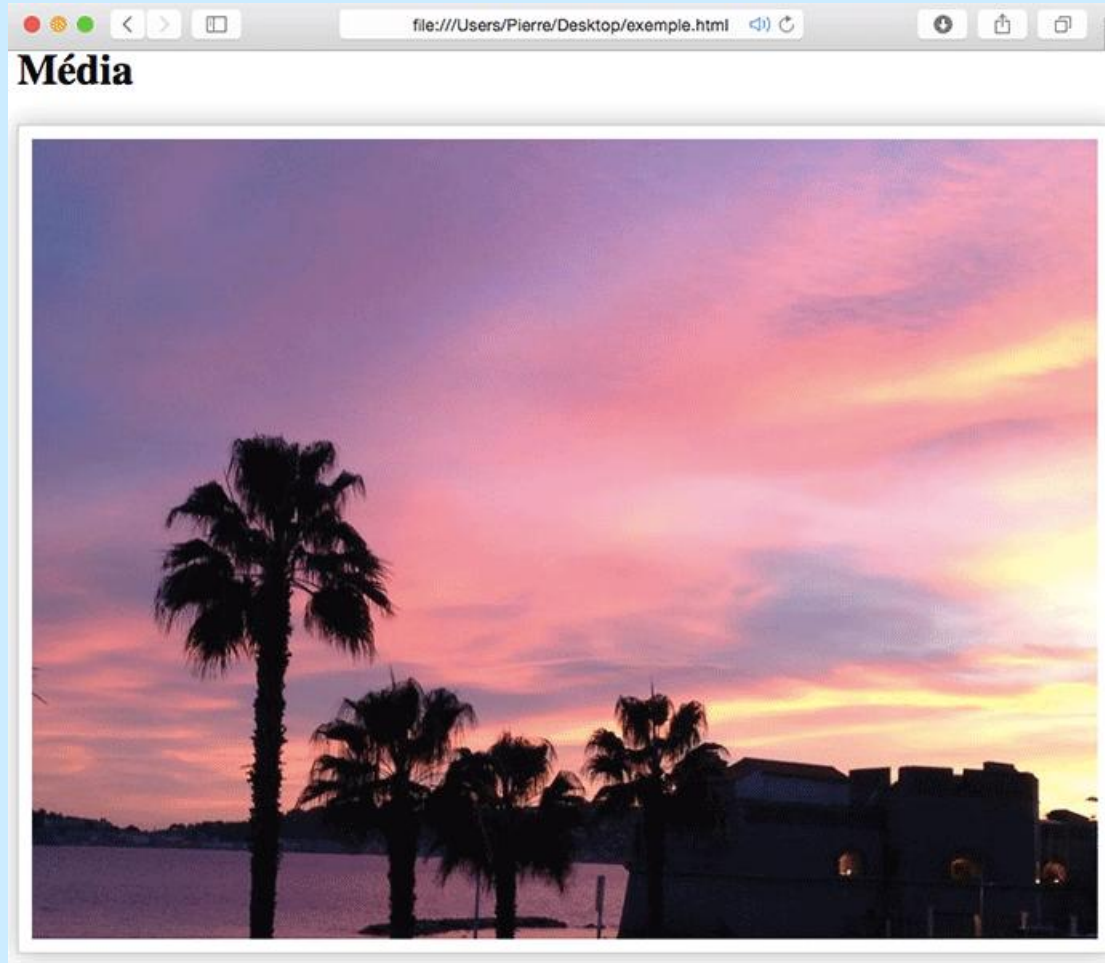
- Afin d'améliorer l'esthétique des images, on peut faire appel aux propriétés vues dans la partie concernant le modèle des boîtes pour créer un cadre autour des images et les faire ressortir.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Média</title>
    <meta charset='utf-8'>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>Média</h1>
    
  </body>
</html>
```

```
img{
  padding: 10px;
  border: 1px solid #CCC;
  box-shadow: 0px 0px 17px #BBB;
}
```

Créer un cadre autour des images

- (Suite)



HTML Tutorial

INSERER DU CONTENU AUDIO EN HTML

Un problème de formats

- Pour **insérer du contenu audio** dans un document HTML, on dispose de l'élément HTML **audio**.
- Mais problème, la prise en charge de l'audio par le HTML est assez récente (cela date du HTML5).
- Ainsi, tandis qu'il existe différents formats audio (tout comme il existe différents formats d'image), aucun de ces formats n'est reconnu de façon universelle par les vieilles versions des principaux navigateurs.
- Il va donc falloir insérer plusieurs fois la même piste audio sous différents formats afin d'avoir une meilleure prise en charge du fichier par les principaux navigateurs.

Un problème de formats

- Pour information, ci-après la liste des formats audio supportés par l'élément HTML audio et leur prise en charge ou non par les dernières versions des navigateurs les plus courants.
- Comme on peut le voir, les dernières versions des navigateurs les plus connus supportent toutes le mp3.
- Mais il reste conseillé de présenter au moins deux formats pour tenir compte d'anciennes versions.

Un problème de formats

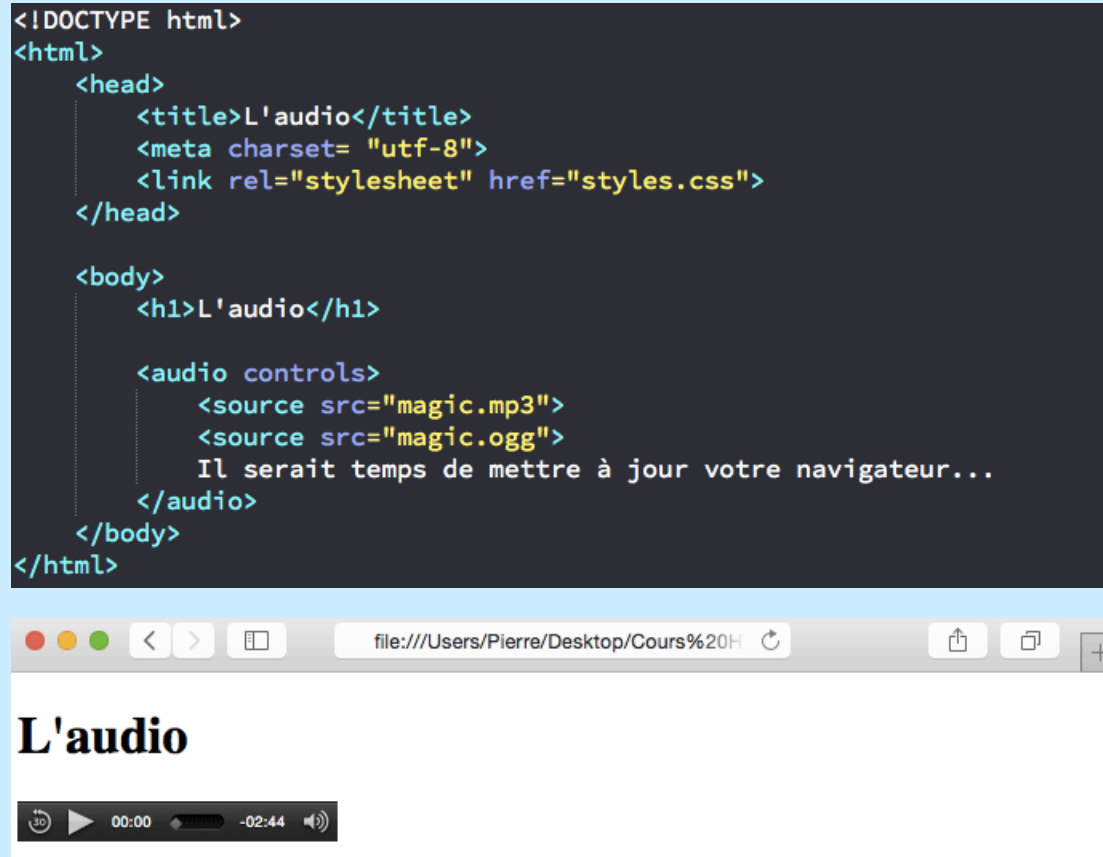
Navigateur	MP3	Ogg	Wav
Chrome	Oui	Oui	Oui
Safari	Oui	Non	Oui
Firefox	Oui (seulement depuis la version 21)	Oui	Oui
Internet Explorer	Oui	Non	Non
Opera	Oui	Oui	Oui

Utilisation de l'élément HTML audio et de ses attributs

- Pour illustrer l'utilisation de l'audio en HTML, soit à disposer de deux fichiers représentant la même piste audio, tels que: « magic.mp3 » et « magic.ogg » stocké dans le même dossier que le fichier HTML.
- L'élément audio possède un attribut obligatoire : l'attribut **controls** qui ne possède qu'une valeur (que nous ne sommes donc pas strictement obligé de préciser car celle-ci est implicite).
- Celui-ci a simplement pour objet d'afficher les contrôles de base (lecture, pause, retour au début et volume) de la piste audio.
- Afin de laisser le choix de fichier au navigateur , on utilise en outre de l'élément audio un élément **source** (représenté sous forme de balise orpheline) avec un attribut **src**.
- A tout cela, on ajoute un texte dans notre élément audio qui ne sera affiché que si aucun fichier audio n'a pu être lu.

Utilisation de l'élément HTML audio et de ses attributs

- Ce qui donne :



Utilisation de l'élément HTML audio et de ses attributs

- Les attributs facultatifs que l'on peut ajouter au fichier audio afin d'en améliorer les fonctionnalités sont les suivants :
- L'attribut **preload** permet de charger le fichier en même temps que la page web. Cet attribut possède trois valeurs : **auto**, **metadata** et **none**.
 - **auto** va charger tout le fichier audio en même temps que la page web tandis que **metadata** ne va charger que les données méta du fichier audio en même temps que la page web ;
- L'attribut **autoplay** va automatiquement lancer le fichier audio dès que celui-ci est prêt ;
- L'attribut **loop** va faire se répéter le fichier en boucle ;
- L'attribut **muted** va mettre le son du fichier sur 0.

Utilisation de l'élément HTML audio et de ses attributs

- Noter que seul l'attribut **preload** a besoin d'une valeur, car tous les autres ne possèdent qu'une valeur qui est donc utilisée par défaut.
- Cependant, c'est une bonne pratique de toujours préciser une valeur pour un attribut même lorsque celle-ci est évidente, et il est donc recommandé de le faire. (règle XML)

```
<!DOCTYPE html>
<html>
  <head>
    <title>L'audio</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>L'audio</h1>

    <audio controls preload="metadata" loop>
      <source src="magic.mp3">
      <source src="magic.ogg">
      Il serait temps de mettre à jour votre navigateur...
    </audio>
  </body>
</html>
```

HTML Tutorial

INSERER DU CONTENU VIDEO EN HTML

Les formats vidéo

- Pour la vidéo, on fait face au même problème que pour l'audio, étant donné qu'une vidéo est composée d'un codec audio, d'un codec vidéo et d'un conteneur rassemblant ces deux éléments.
- Une nouvelle fois, il sera donc conseillé d'utiliser différents formats vidéo afin de s'assurer une compatibilité avec le maximum de versions de navigateurs.
- On trouvera ci-après le tableau des compatibilités entre les formats vidéo les plus utilisés et les dernières versions des navigateurs les plus courants.

Les formats vidéo

- (Suite)

Navigateur	MP4	Ogg	WebM
Chrome	Oui	Oui	Oui
Safari	Oui	Non	Non
Firefox	Oui (seulement depuis la version 21)	Oui	Oui
Internet Explorer	Oui	Non	Non
Opera	Oui	Oui	Oui

L'élément video et ses attributs

- L'élément video s'utilise exactement comme l'élément audio. De plus, il possède les mêmes attributs que ce dernier à savoir :
- Un attribut **controls** pour ajouter les contrôles de base de la vidéo (lecture, arrêt, etc.) ;
- Un attribut **preload** pour précharger la vidéo, avec les trois valeurs: **auto** (chargement total), **metadata** (chargement partiel de la vidéo) et **none** (pas de chargement) ;
- Un attribut **autoplay** pour activer la lecture automatique de la vidéo ;
- Un attribut **loop** pour lire une vidéo en boucle ;
- Un attribut **muted** pour désactiver le son de la vidéo.

L'élément video et ses attributs

- Outre cela, l'élément video possède quelques attributs qui lui sont propres, notamment l'attribut **poster** qui permet d'afficher une image de fond avant que la vidéo ne soit lancée.
- Pour un exemple d'insertion de contenu vidéo dans une page web, on se dote de deux versions d'un même fichier vidéo : « lever-soleil.mp4 » et « lever-soleil.ogg ».
- On fait appel également à une image « lever-soleil.png », placée dans le même dossier, comme image de fond. (A recadrer préalablement à la taille de la vidéo. Noter qu'on peut changer la taille de la vidéo en CSS en appliquant les propriétés height et width sur l'élément video.)
- Enfin , il faut savoir que l'on pourrait aller beaucoup plus loin dans la personnalisation des options du lecteur vidéo, mais il nous faudrait pour cela utiliser le langage JavaScript.

L'élément video et ses attributs

- (Suite)

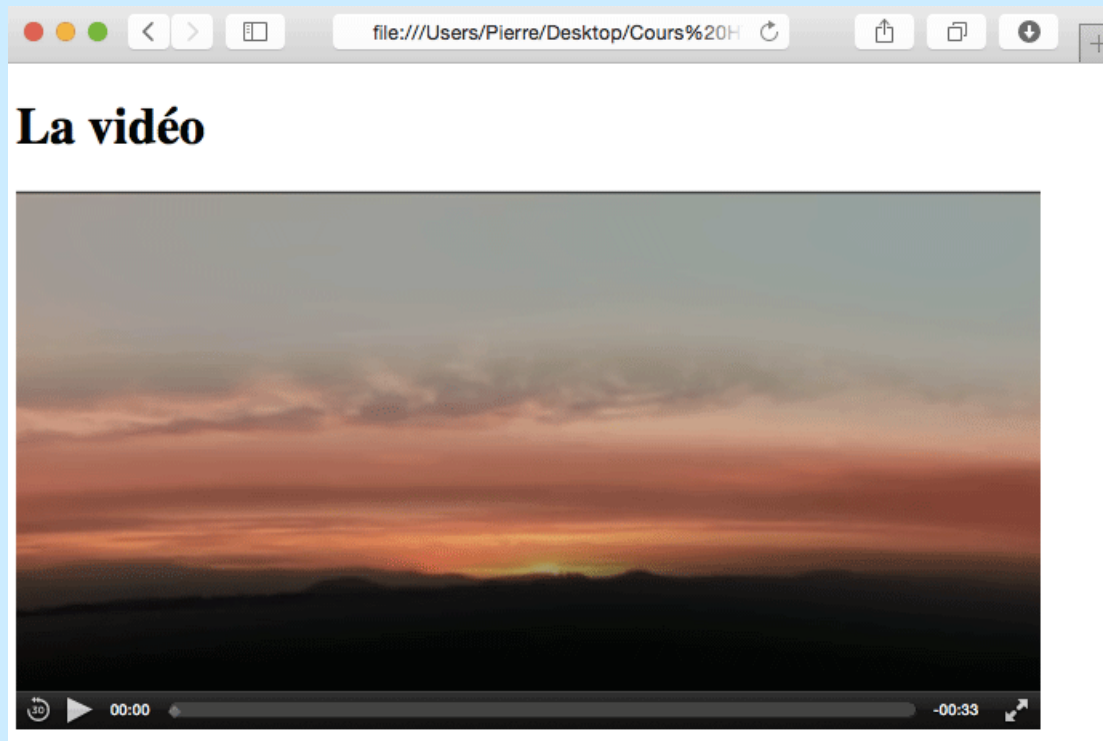
```
<!DOCTYPE html>
<html>
  <head>
    <title>Contenu vidéo</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>La vidéo</h1>

    <video controls preload="metadata" loop poster="lever-soleil.png">
      <source src="lever-soleil.mp4">
      <source src="lever-soleil.ogg">
      Il serait temps de mettre à jour votre navigateur...
    </video>
  </body>
</html>
```

L'élément video et ses attributs

- (Suite)



HTML Tutorial

INTEGRATION ET SEMANTIQUE :
YOUTUBE, FIGURE ET FIGCAPTION

L'alternative YouTube

- Si vous ne souhaitez pas vous embêter à créer différents formats de pistes audio ou de vidéo et à gérer le tout en HTML, vous disposez d'une autre alternative : télécharger votre audio ou votre vidéo sur un site comme YouTube, Dailymotion ou Vimeo puis intégrer le contenu sur votre site.
- En procédant ainsi, vous êtes certain d'assurer une compatibilité maximale avec n'importe quelle version de chaque navigateur puisque le site hébergeur va lui-même s'occuper de créer les différentes pistes adéquates et gérer la prise en charge.
- Pour intégrer une vidéo préalablement postée sur YouTube, par exemple, il vous suffit de vous rendre dans l'onglet « partager » de la vidéo puis de choisir « intégrer ». Vous n'aurez alors plus qu'à copier-coller le code d'intégration dans votre page HTML, à l'endroit voulu.

Les éléments figure et figcaption

- Les éléments **figure** et **figcaption** aident à marquer sémantiquement du contenu média comme des images, de l'audio ou des vidéos.
- En effet, il est difficile pour un navigateur de savoir « de quoi parle » ou « ce que représente » une image, un fichier audio ou vidéo sans plus de détail.
- On utilise l'élément HTML **figure** pour indiquer qu'une image, une piste audio ou vidéo n'est pas strictement décorative, mais sert à la compréhension générale de la page web. On n'utilisera donc pas cet élément si les contenus ne sont là que pour habiller la page.
- On utilise l'élément **figcaption** à l'intérieur de figure pour accoler une légende au contenu.

Les éléments figure et figcaption

- (Suite)

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="styles.css">
  </head>

  <body>
    <h1>HTML et CSS</h1>

    <p>
      Le HTML et le CSS sont à la base de tout projet web. Aujourd'hui,
      nous en sommes aux versions HTML5 et CSS3. Vous pouvez trouver
      les logos officiels de ces langages ci-dessous.
    </p>
    <figure>
      
      <figcaption>Logos officiels HTML5 et CSS3</figcaption>
    </figure>
  </body>
</html>
```

Les éléments figure et figcaption

- (Suite)



-
- (Suite)