

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and circles that resemble a circuit board or a stylized tree structure, extending from the top to the bottom of the frame.

RESPONSIVE DESIGN CSS

DELJEHIER VIRGIL - ATELIER NUMÉRIQUE AGEVAL

COMPRENDRE LES DÉFIS APPORTÉS

- Avec la miniaturisation des composants et les capacités de connexion de plus en plus puissante, cependant, de nouveaux appareils connectés ont vu le jour et les sites web peuvent maintenant être consultés à partir de nombreux appareils différents : ordinateurs de bureau, ordinateurs portables, tablettes, smartphones, montres connectées, etc.
- Cela a multiplié les défis en termes d'ergonomie et de design pour les concepteurs de site : en effet, comment faire en sorte qu'un site Internet s'affiche correctement à la fois sur un écran de bureau et sur un smartphone ? Doit-on créer deux sites différents ? Deux versions d'un même site ? Dans ce cas-là, est-il préférable de retirer des informations pour rendre la version mobile plus légère ou peut-on se « contenter » de réarranger le contenu ?

RESPONSIVE DESIGN ?

- site web devrait s'afficher aussi bien sur un écran de PC que sur un écran de smartphone ou sur n'importe quel type d'appareil.
- Aujourd'hui, nous pouvons utiliser principalement trois méthodes pour répondre aux défis amenés par les différentes tailles d'écran. On peut :
 - Créer une application dédiée pour les mobiles ;
 - Créer une « copie mobile » de notre site en utilisant l'initiative AMP (« Accelerated Mobile Pages ») de Google ;
 - Utiliser les Media Queries ou requêtes media.

LES MEDIA QUERIES

- Les Media Queries vont nous permettre d'appliquer certaines règles CSS de manière conditionnelle. Par exemple, on va pouvoir définir une largeur pour un élément pour certaines tailles d'écrans et une largeur différente pour le même élément pour d'autres tailles d'écran.
- Cela va nous permettre d'afficher des pages avec des organisations différentes selon la taille de l'écran d'un visiteur. Attention ici à ne pas confondre les Media Queries et les valeurs de taille relatives et en % ou le flexbox → exemple : [responvie.html](#)

L'ÉLÉMENT HTML MÉTA

- L'élément HTML meta est utilisé pour définir des métadonnées pour un document HTML.
- Une métadonnée est une donnée qui ne va pas être affichée sur la page mais qui va pouvoir servir aux différents robots pour comprendre et afficher la page.
- Utilisation de meta avec les attributs name et content → author / description / viewport / keyword

LE META VIEWPORT : QUELLES VALEURS CHOISIR ?

- On va définir une balise meta name="viewport" avec propriétés et des valeurs adaptées afin que les différents navigateurs mobiles ne définissent pas eux-mêmes leur propre viewport et leur niveau de zoom de notre page.
- Généralement, nous définirons une largeur de viewport égale à la largeur de l'appareil dans le viewport ainsi qu'un niveau de zoom initial égal à 1 et interdirons les utilisateurs de zoomer ou de dézoomer. Nous allons donc utiliser la balise suivante

```
<head>
  <title>Cours HTML et CSS</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no">
  <link rel="stylesheet" href="cours.css">
</head>
```

SYNTAXE DES MEDIA QUERIES

- A l'heure actuelle, nous pouvons choisir parmi les types de media suivants :
 - all : Valeur par défaut. Nos règles vont s'appliquer à tous les appareils ;
 - Screen : Nos règles ne vont s'appliquer qu'aux appareils dotés d'un écran ;
 - Printer : Nos règles ne s'appliqueront que pour les imprimantes ;
 - Speech : Nos règles ne s'appliqueront qu'aux lecteurs d'écran qui sont capable de rendre le contenu d'une page de manière sonore.

```
/*Styles spécifiques pour les écrans de taille moyenne type tablette*/  
@media screen and (min-width: 780px) and (max-width: 979px){  
  .conteneur-flex{  
    flex-flow: row wrap;  
  }  
  .sous-conteneur-flex{  
    margin: 0px 10px;  
  }  
}
```


IMAGES RESPONSIVE

- En n'utilisant qu'une seule image, nous allons être obligés de choisir la version la plus grande possible de celle-ci afin qu'elle ne pixellise pas lorsqu'un utilisateur l'affiche sur un grand écran. Or, plus une image est grande et plus elle est lourde : nous allons donc imposer aux mobiles le téléchargement d'une image très lourde sans raison ce qui va ralentir la vitesse d'affichage de nos pages ;
- En n'utilisant qu'une seule image, il est fort probable que l'image ne rende pas bien sur mobile car comme celle-ci sera plus petite à cause de la taille de l'écran le sujet dans l'image va apparaître lointain ou comme dézoomé.

PIXELS PHYSIQUES, PIXELS CSS ET ÉCRANS RETINA

- Ici, retenez que les écrans retina ont généralement une résolution deux fois plus importante que les écrans standards. Cela signifie que chaque « pixel retina » est l'équivalent de 4 « pixels standards » (2 en largeur, 2 en hauteur).
- Ainsi, pour qu'une image s'affiche correctement sur un écran retina, il faudra qu'elle soit deux fois plus grande que l'écran sur lequel elle doit s'afficher (1960x 980px par exemple pour un affichage sur un écran retina de 980x 490px).

1ERE SOLUTION : IMAGES SVG

- La première solution, qui semble la plus évidente pour gérer les différentes tailles d'écran et les différentes résolutions est d'utiliser des images au format SVG pour Scalable Vector Graphic.
- Comme le nom l'indique, ces images sont vectorielles, ce qui signifie qu'on va pouvoir les agrandir ou les rapetisser à l'infini sans perte de qualité.
- Cependant, cela ne résout qu'une partie du problème puisqu'en n'utilisant qu'une seule image pour toutes les versions de notre site, cette image risque d'apparaître comme trop imposante pour la version bureau ou trop dézoomée pour la version mobile.

PROPOSER PLUSIEURS VERSIONS D'UNE IMAGE EN UTILISANT LES ATTRIBUTS HTML SRCSET ET SIZES

- La première solution, qui semble la plus évidente pour gérer les différentes tailles d'écran et les différentes résolutions est d'utiliser des images au format SVG pour Scalable Vector Graphic.
- Comme le nom l'indique, ces images sont vectorielles, ce qui signifie qu'on va pouvoir les agrandir ou les rapetisser à l'infini sans perte de qualité.
- Cependant, cela ne résout qu'une partie du problème puisqu'en n'utilisant qu'une seule image pour toutes les versions de notre site, cette image risque d'apparaître comme trop imposante pour la version bureau ou trop dézoomée pour la version mobile.



MERCI À VOUS !
AVEZ-VOUS DES QUESTIONS ?