

# CSS3 : Mobile

- Introduction
- Dimensions des écrans
- Viewport et unités
- Media Queries
- Media Features
- Techniques de design

# Introduction

- Aujourd'hui, un site doit être accessible sur différents supports (smartphone, tablette, PC, auto, montre,...). Il faut adapter le design et l'interaction à tous ces dispositifs.
- Plutôt que créer plusieurs sites pour chaque support, on en crée un seul qui s'adapte
- Selon le support, le contenu est affiché en taille réelle ou la largeur du contenu est calculée pour s'ajuster sur celle de l'écran

# Largeur d'écran

- Contrairement au PC, sur mobile, le navigateur ne dispose pas de "fenêtre" réelle, ni de barres de défilement car tout est prévu pour naviguer de manière tactile
- La surface physique n'est pas toujours égale à la surface utilisable en pixels (exemple : les écrans Retina ou haute définition)
  - Largeur physique (iPhone 5) : device-width = 320px
  - Largeur utilisable (iPhone 5) : screen-width = 640px
  - Et "viewport" de l'iPhone 5 sur Safari = 980px !

# Viewport

- Le **viewport** est la zone fictive utilisée par les **navigateurs** pour mobiles pour l'affichage d'une page web. Il peut donc varier !

<https://www.mydevice.io/#compare-devices>

<http://screensiz.es/>

- On va le paramétrer via une balise `<meta>`  
`<meta name="viewport" content="width=320" />`  
Si on fixe la largeur à 320px, pas idéal si on pivote !

# Adapter le viewport

- Pour s'adapter à l'appareil

```
<meta name="viewport"  
      content="width=device-width" />
```

- Si la page Web est plus large que le viewport ?

```
<meta name="viewport"  
      content="width=device-width,initial-scale=1" />
```

- On fixe le facteur d'agrandissement pour que le contenu ne subisse aucune réduction

# Sans ou avec viewport

## Exemple SANS viewport défini



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.

## Exemple AVEC viewport défini



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril

# Sans viewport (code)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>SANS viewport</title>
    <style>
      img {
        max-width: 100%;
        height: auto;
      }
    </style>
  </head>
  <body>
    <h1>Exemple SANS viewport défini</h1>
    
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod
    tincidunt ut laoreet dolore magna aliquam erat volutpat.
    Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea
    commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie
    consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui
    blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi.</p>
  </body>
</html>
```

# Avec viewport (code)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>SANS viewport</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <style>
      img {
        max-width: 100%;
        height: auto;
      }
    </style>
  </head>
  <body>
    <h1>Exemple SANS viewport défini</h1>
    
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod
    tincidunt ut laoreet dolore magna aliquam erat volutpat.
    Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea
    commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie
    consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui
    blandit praesent luptatum zzril delenit augue duiis dolore te feugait nulla facilisi.</p>
  </body>
</html>
```



# Unités liées au viewport

- Des nouvelles unités relatives au viewport ont été ajoutées pour s'adapter à la partie visible de la page
  - vw :  $1/100^{\text{ème}}$  de la largeur du viewport
  - vh :  $1/100^{\text{ème}}$  de la hauteur du viewport
  - vmin :  $1/100^{\text{ème}}$  de la plus petite dimension du viewport

# Media Queries

- Pour afficher au mieux, il faut **adapter le design** de la page à la dimension du média
- On définit des styles spécifiques en fonction des périphériques, avec la règle **@media** qu'on appelle un **media query**
- Les valeurs possibles les plus utilisées sont : **screen**, print, projection, handheld, tv, all
- Il y a aussi : speech, braille, embossed, tty

# Media features

- Associés aux medias, il y a les **media-features** qui sont des propriétés liées au navigateur
- On les place entre parenthèses combinées avec les media pour définir les critères d'affichage
- Les valeurs les plus courantes sont : width, height, orientation, color, resolution,...
- On définit des plages de valeurs avec les préfixes **min-** et **max-** et on utilise des booléens pour tester (**and** pour le ET) et (, pour le OU)

# Exemples de media-queries

- Pour les écrans plus large que 1024px

```
@media screen and  
(min-width:1024px)  
{  
  body {  
    background-color:white;  
  }  
  
  p {  
    color:black;  
  }  
}
```

- Pour les écrans en mode paysage

```
@media screen and  
(orientation:landscape)  
{  
  body {  
    background-color:#555;  
  }  
  
  #menu {  
    float:left;  
  }  
}
```

# Exemples de media-queries

- Pour écran ou imprimante ou TV

```
@media screen, print, tv
{
  body {
    background-color:white;
  }

  #pub {
    display:none;
  }
}
```

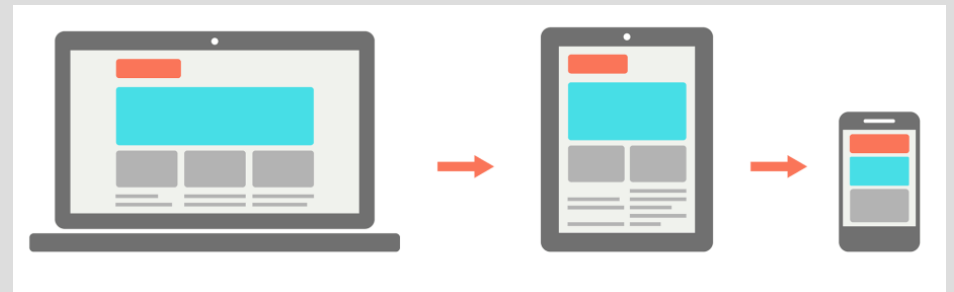
- Pour les écrans monochromes

```
@media only screen and
(not color)
{
  body {
    background-color:white;
  }

  h1, h2 {
    color:black;
  }
}
```

# Techniques de design

- Responsive design
  - Conçu pour le PC avec un contenu détaillé et on élimine au fur et à mesure
- Mobile-first design
  - Conçu pour le mobile avec le contenu principal et on ajoute les détails



# Texte adaptatif

- On peut faire varier la typographie en utilisant les unités liées au viewport (**vw** :viewport width)

```
<h1 style="font-size:4vw;">Hello World</h1>
```

- On peut utiliser les media-queries et modifier la taille de police selon l'écran

```
@media screen and (min-width: 601px) {  
  p { font-size: 32px; }  
}
```

```
@media screen and (max-width: 600px) {  
  p { font-size: 20px; }  
}
```

# Image adaptative

- **Image** : largeur à 100% et hauteur automatique  
`img { max-width: 100%; height: auto; }`
- **Image de fond** avec **background-size** avec les valeurs **contain** ou **cover** pour couvrir au mieux la zone en gardant le ratio largeur/hauteur
- Différentes images pour différents écrans
  - Avec les **media-queries** en CSS3
  - Avec les balises HTML5 **<picture>** **<source>**