

# Responsive Web design

CSS3 and HTML5

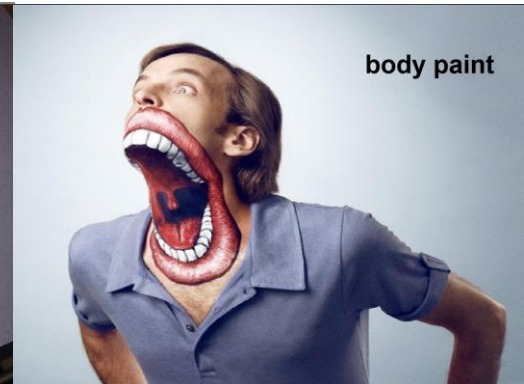
# Traditional desing



print



card



body paint

packaging



car wrap

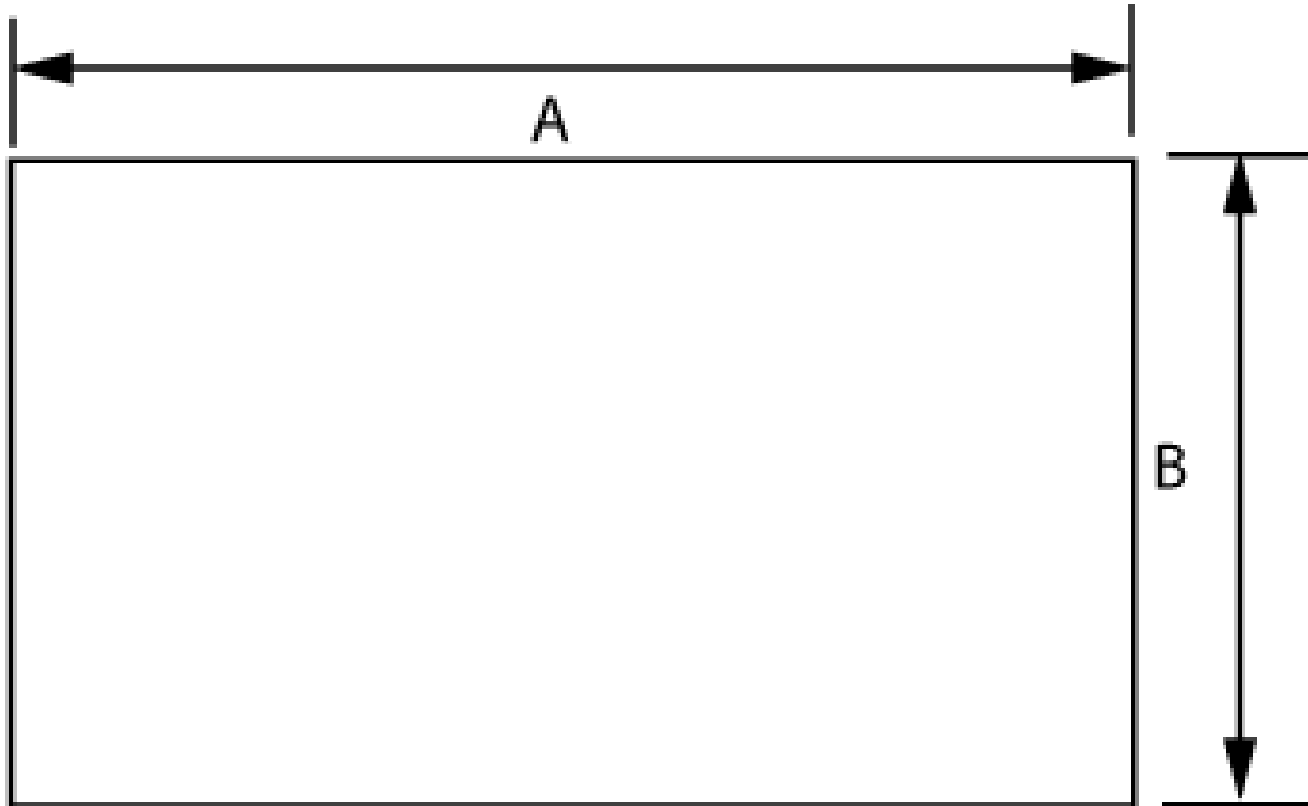
fashion



Dezigual®

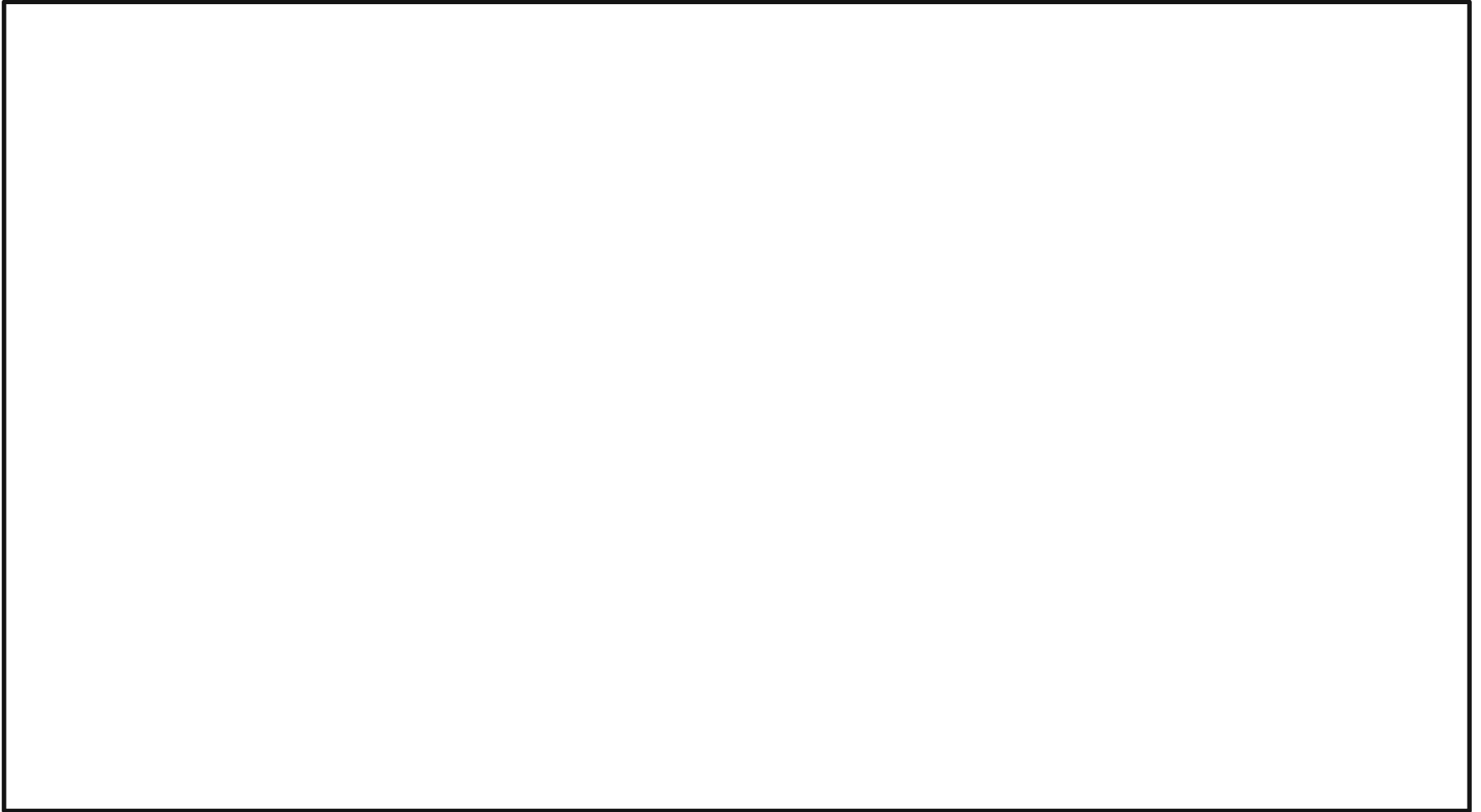
# The common thing

---




# The web

---



# Does it make sense?



Twitter  
@960GRIDSYSTEM


# 960

GRID SYSTEM

[Download](#) (260 KB) → Templates for Fireworks, Inkscape, Illustrator, OmniGraffle, Photoshop & Visio. Printable sketch paper. CSS framework. The 960.css file is 3.6 KB. Repository at [GitHub](#).

DESIGNED BY FUSION

Bring your photos to canvas with one-week worldwide delivery, 10% off with coupon "FUSION"





Custom CSS generator



Grid overlay bookmark

## Essence

The 960 Grid System is an effort to streamline web development workflow by providing commonly used dimensions, based on a width of 960 pixels. There are two variants: 12 and 16 columns, which can be used separately or in tandem. [Read more](#).

## Dimensions

The 12 column grid is divided into portions that are 60 pixels wide. The 16 column grid consists of 40 pixel increments. Each column has 10 pixels of margin on the left and right, which create 20 pixel wide gutters between columns. [View demo](#).

## Purpose

The premise of the system is ideally suited to rapid prototyping, but it would work equally well when integrated into a production environment. There are printable sketch sheets, design layouts, and a CSS file that have identical measurements.

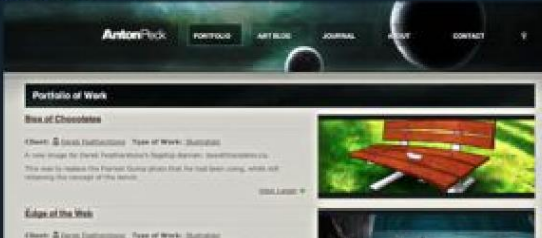
[Four Kitchens](#) — 12 col [SHOW GRID](#)



Four Kitchens is a handy, one-stop solution for building big websites. Our work is handled almost entirely in-house in Austin, Texas, by experts in design, development, usability, and user experience. But we won't just build websites. We also provide consulting for everything from project strategy to individual components and everything in between. If your Drupal website is slow or crashing your server, our optimization experts can enhance its performance by implementing long-term solutions. No shortcuts!

Our free-culture philosophy

[Anton Pack](#) — 16 col [SHOW GRID](#)



Anton Pack

PORTFOLIO ART BLOG JOURNAL ABOUT CONTACT

Portfolio of Work

Box of Chocolates

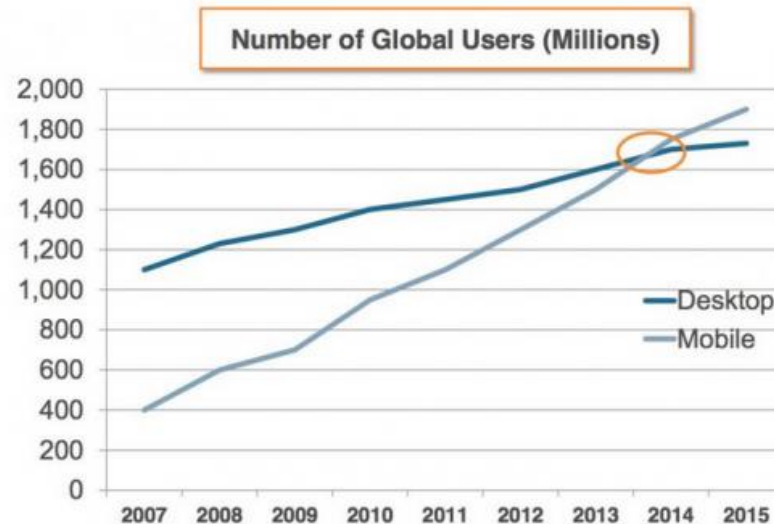
Client: Local Businesses Type of Work: Illustration

A new design for David Friedman's Specialty Sweets. David Friedman's Specialty Sweets is a new design for David Friedman's Specialty Sweets. This was to replace the previous design that had been using, while not retaining the message of the brand.

Edge of the Web

Client: Local Businesses Type of Work: Illustration

# Why is important?



# Solutions for mobile media consumption

---

- native app
  - Tightly attached to an ecosystem.
  - Duplicated content with the web.
  - Only interesting if hardware access is required (e.g. graphics card, accelerometer) ... and think of html5
  - No indexing
- web app
  - Old embedded web browser with limited functionality
- mobile website
  - m site vs www site
  - Duplicated content.
  - Wrong links to desktop webpages
- responsive web design

# Web design nowadays...

---



## What is Responsive Web Design?

- Responsive Web Design makes your web page look good on all devices (desktops, tablets, and phones).
- Responsive Web Design is about using CSS and HTML to resize, hide, shrink, enlarge, or move the content to make it look good on any screen

source:w3school



# Benefits

---

- A Single website
- Same content distributed across different devices
- A single url. Good for the SEO
- A single codebase
- Different functionality
- Different layout

# Example



# Example





STOP  
DREAMING  
AND GET  
BACK TO  
REALITY

# Responsive design principles

---

- Stop thinking in pixels
  - Think proportional
  - Think percentages
  - Think modular
- Mobile first
  - Basic formatting to the page (type, colors, margins)
  - Media queries for bigger screens
    - Less capable browsers will always served a readable site

# HTML5 - First things first

---

HTML v5 introduces mechanisms to dynamically know the visible part of a webpage.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

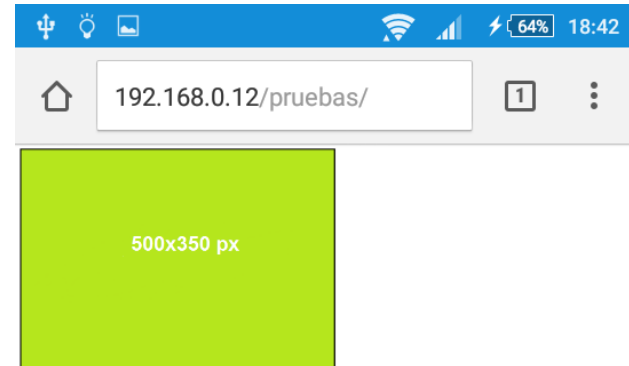
- Name is the name we assign to the element, viewport in this case.
- With the content we state that the width of the page should be the same as the device-width. Will vary between a Tablet and a Smartphone.
- With initial-scale=1.0 starts the visualization of the webpage with no zoom.

Do not forget to use “,” as separation to have full browser compatibility.

# Example



On this the White Rabbit blew three blasts on the trumpet, and then unrolled the parchment scroll, and read as follows: 'The Queen of Hearts, she made some tarts, All on a summer day: The Knave of Hearts, he stole those tarts, And took them quite away!' 'Consider your verdict,' the King said to the jury. 'Not yet, not yet!' the Rabbit hastily interrupted. 'There's a great deal to come before that!' 'Call the first witness,' said the King; and the White Rabbit blew three blasts



On this the White Rabbit blew three blasts on the trumpet, and then unrolled the parchment scroll, and read as follows: 'The Queen of Hearts, she made some tarts, All on a summer day: The Knave of Hearts, he stole those tarts, And took them quite away!' 'Consider your verdict,' the King said to the jury. 'Not yet, not yet!' the Rabbit hastily interrupted. 'There's a great deal to come before that!' 'Call the first witness,' said the King; and the White Rabbit blew three blasts on the trumpet, and called out, 'First witness!' The first witness was the Hatter. He came in with a teacup in one hand and a piece of bread-and-butter in the other. 'I beg pardon, your Majesty,' he began, 'for bringing these in: but I hadn't quite finished my tea when I was sent for.'

# Fluid grids



What we need for this:

- Box-sizing
- A grid system
- Media-queries



# Box-Sizing: CSS3

```
<!DOCTYPE html>
<html>
<head>
<style>

```

## Without box-sizing

This div is smaller (width is 300px and height is 100px).

This div is bigger (width is also 300px and height is 100px).

## With box-sizing

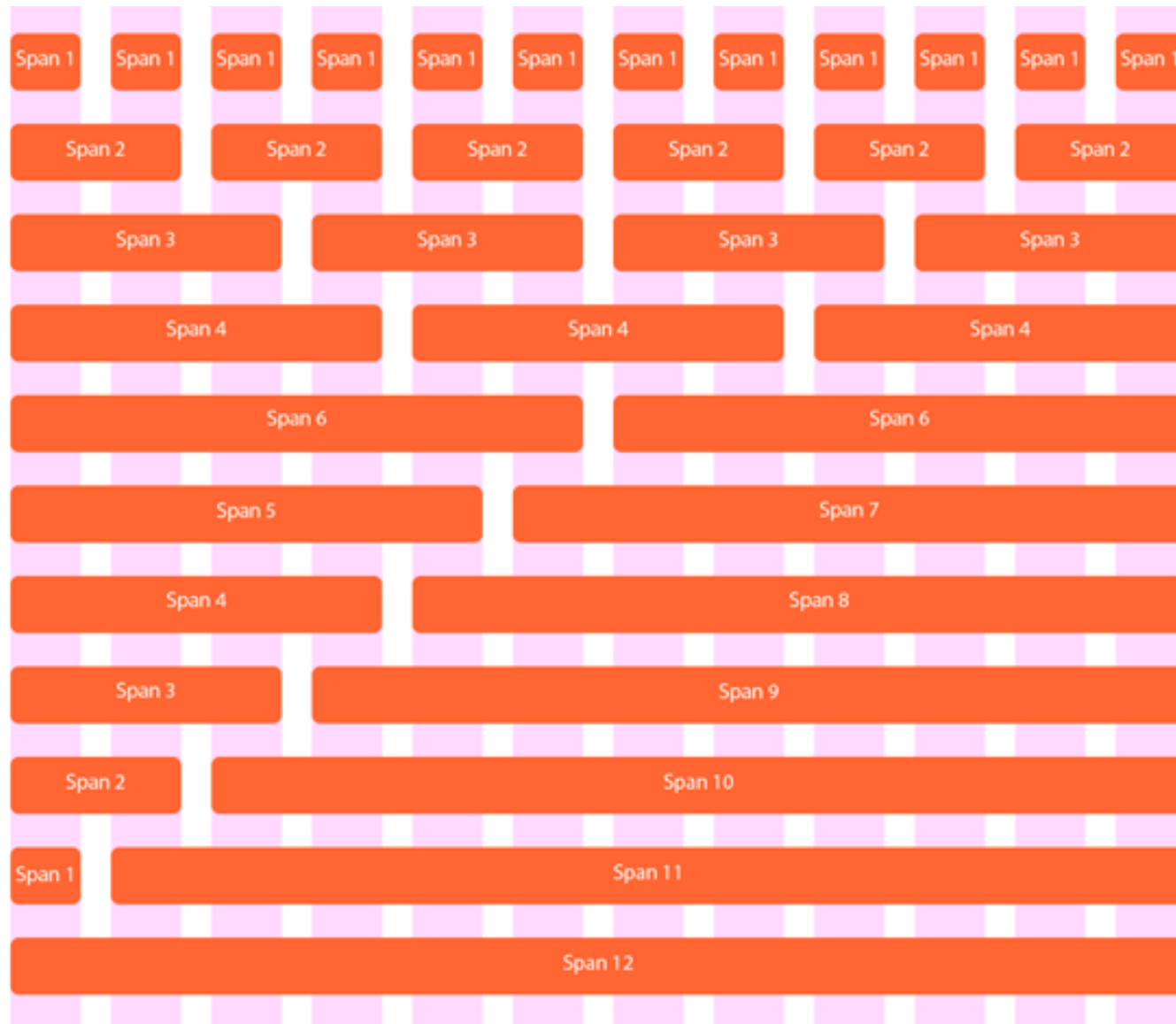
Both divs are the same size now!

Hooray!

[http://www.w3schools.com/CSSref/css3\\_pr\\_box-sizing.asp](http://www.w3schools.com/CSSref/css3_pr_box-sizing.asp)

Por defecto el tamaño de un elemento es ancho+padding+borde para el ancho y alto+padding+borde para el alto, Box-sizing es una propiedad de CSS3 que permite incluir padding y bordes en el cálculo del tamaño total de un elemento.

# How to build up a grid system?



# How to build up a grid system?

---

Divide the view-port in n **equal** columns

- No more pixel-based divisions.
- Traditionally, we divide in 10 or 12 columns

- They should have the form

```
[class*="col-"] {  
    float: left;  
    padding: 15px;  
}
```

```
.col-1 {width: 10%;}  
.col-2 {width: 20%;}  
.col-3 {width: 30%;}  
.col-4 {width: 40%;}  
.col-5 {width: 50%;}  
.col-6 {width: 60%;}  
.col-7 {width: 70%;}  
.col-8 {width: 80%;}  
.col-9 {width: 90%;}  
.col-10 {width: 100%;}
```

- We need a container for all columns

```
<div class="columnas">  
    <div class="col-2">...</div>  
    <div class="col-9">...</div>  
</div>
```

```
.columnas::after {  
    content: "";  
    clear: both;  
    display: block;  
}
```

# Example

## RESPONSIVE WEB DESIGN

A

B

C

D

### Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

## RESPONSIVE WEB DESIGN

A

B

C

D

### Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.



# Media Queries

---

- The spirit of if-else
  - if media query test is met > css will be applied
- common breakpoints: 320 480 600 768 960 1200
  - [responsivexp.com](https://responsivexp.com) ← for testing different viewport sizes

# Media Queries

## Mobile

```
[class*="col-"] {  
float:left;  
width: 100%;  
}
```

## Tablets

```
@media only screen  
and (min-width:  
600px) {
```

```
.col-tab-1 {width: 10%;}  
.col-tab-2 {width: 20%;}  
.col-tab-3 {width: 30%;}  
.col-tab-4 {width: 40%;}  
.col-tab-5 {width: 50%;}  
.col-tab-6 {width: 60%;}  
.col-tab-7 {width: 70%;}  
.col-tab-8 {width: 80%;}  
.col-tab-9 {width: 90%;}  
.col-tab-10 {width: 100%;}  
}
```

## Laptops

```
@media only screen  
and (min-width: 768px) {
```

```
.col-1 {width: 10%;}  
.col-2 {width: 20%;}  
.col-3 {width: 30%;}  
.col-4 {width: 40%;}  
.col-5 {width: 50%;}  
.col-6 {width: 60%;}  
.col-7 {width: 70%;}  
.col-8 {width: 80%;}  
.col-9 {width: 90%;}  
.col-10 {width: 100%;}  
}
```

# Media Queries

## RESPONSIVE WEB DESIGN

A

B

C

D

### Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido según las características del dispositivo, como la resolución de pantalla.

## Tablets

@media only screen  
and (min-width:  
600px) {

```
.col-tab-1 {width: 10%;}  
.col-tab-2 {width: 20%;}  
.col-tab-3 {width: 30%;}  
.col-tab-4 {width: 40%;}  
.col-tab-5 {width: 50%;}  
.col-tab-6 {width: 60%;}  
.col-tab-7 {width: 70%;}  
.col-tab-8 {width: 80%;}  
.col-tab-9 {width: 90%;}  
.col-tab-10 {width: 100%;}
```

}

## Laptops

@media only screen  
and (min-width: 768px) {

```
.col-1 {width: 10%;}  
.col-2 {width: 20%;}  
.col-3 {width: 30%;}  
.col-4 {width: 40%;}  
.col-5 {width: 50%;}  
.col-6 {width: 60%;}  
.col-7 {width: 70%;}  
.col-8 {width: 80%;}  
.col-9 {width: 90%;}  
.col-10 {width: 100%;}
```

}

# Media Queries

## RESPONSIVE WEB DESIGN

A

B

C

D

## Tablets

@media only screen  
and (min-width:  
600px) {

.col-tab-1 {width: 10%;}

.col-tab-2 {width: 20%;}

.col-tab-3 {width: 30%;}

## Laptops

@media only screen  
and (min-width: 768px) {

.col-1 {width: 10%;}

.col-2 {width: 20%;}

.col-3 {width: 30%;}

.col-4 {width: 40%;}

.col-5 {width: 50%;}

.col-6 {width: 60%;}

.col-7 {width: 70%;}

.col-8 {width: 80%;}

.col-9 {width: 90%;}

.col-10 {width: 100%;}

}

## RESPONSIVE WEB DESIGN

Grid-View  
nuest  
tar

A

B

Acerca

C

Media C  
adaptar  
caracter

D

## Grid View

Grid-View es una forma de gestionar el espacio de  
nuestra página web mediante columnas de un tamaño  
porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de  
tamaño.

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido  
según las características del dispositivo, como la resolución de pantalla.



# RESPONSIVE WEB DESIGN

## Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido según las características del dispositivo, como la resolución de pantalla.

## RESPONSIVE WEB DESIGN

## Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

```
.col-1 {width: 10%;}  
.col-2 {width: 20%;}  
.col-3 {width: 30%;}  
.col-4 {width: 40%;}  
.col-5 {width: 50%;}  
.col-6 {width: 60%;}  
.col-7 {width: 70%;}  
.col-8 {width: 80%;}  
.col-9 {width: 90%;}  
.col-10 {width: 100%;}  
}
```

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido según las características del dispositivo, como la resolución de pantalla.

# RESPONSIVE WEB DESIGN

## Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido según las características del dispositivo, como la resolución de pantalla.

## RESPONSIVE WEB DESIGN

## Grid View

Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de un tamaño porcentual al total de la pantalla.

Acerca o aleja el zoom para ver como cambia de tamaño.

```
.col-1 {width: 10%;}  
.col-2 {width: 20%;}  
.col-3 {width: 30%;}  
.col-4 {width: 40%;}  
.col-5 {width: 50%;}  
.col-6 {width: 60%;}  
.col-7 {width: 70%;}  
.col-8 {width: 80%;}  
.col-9 {width: 90%;}  
.col-10 {width: 100%;}  
}
```

Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido según las características del dispositivo, como la resolución de pantalla.

# Media Queries

---

```
<div class="columnas">
  <div class="col-2 col-tab-4 menu">
    <ul>
      <li>A </li>
      <li>B</li>
      <li>C</li>
      <li>D</li>
    </ul>
  </div>
  <div class="col-6 col-tab-6">
    <h1>Grid View</h1>
    <p>
      Grid-View es una forma de gestionar el espacio de nuestra página web mediante columnas de
      un tamaño porcentual al total de la pantalla.
    </p>
    <p>
      Acerca o aleja el zoom para ver como cambia de tamaño.
    </p>
  </div>
  <div class="col-2 col-tab-10">
    <p>
      Media Queries es un módulo de CSS3 que permite adaptar la representación del contenido
      según las características del dispositivo, como la resolución de pantalla.
    </p>
  </div>
</div>
```

# Images and video

---

```
img {  
    max-width: 100%;  
    height: auto;  
}
```

```
video {  
    max-width: 100%;  
    height: auto;  
}
```

```
<picture>  
  <source media="(min-width: 768px)" srcset="mas768.png" >  
  <source media="(min-width: 600px)" srcset="mas600.png">  
    
</picture>
```

# Frameworks

---

- Bootstrap 3
- Foundation
- Skeleton
- ...

# Guidelines and conclusions

---

- Los usuarios no deberían de tener que desplazarse de forma horizontal por la página o alejar el zoom si hay contenido demasiado ancho para mostrar por lo que deberemos añadir una serie de reglas adicionales a la hora de diseñar nuestra web adaptable.
- Nunca usar elementos con un gran ancho de tipo fijo, en su lugar utilizar atributos como `maxwidth:100%` o `height:auto`.
- El contenido no deberá estar optimizado para un cierto tipo de Viewport.
- Usar CSS Media Queries para aplicar un estilo distinto en pantallas grandes o pequeñas.
- De esta forma conseguiremos que el contenido se muestre correctamente sin importar desde donde estemos visualizando nuestra web.

# Let's do some exercise!

---

<https://jagalindo.github.io/issii-responsive-design/>