Web Programming

CSS Responsive Web Design

Responsive Web Design

- Design web page to be responsive
- Responsive means working on all devices with wide range of size varieties
- Responsive design could be achieved by using only HTML and CSS
- Result could be seen by shrinking, resizing, and changing the layout of devices

Principles of Responsive Design

- Cover most devices
- Keep the design aesthetic
- Must not leave out information, but adapt the content

Viewport

- User’s visible area of a web page
- Varies with the device dimension
- HTML5 lets developer to manipulate the viewport using `<meta>` tag
- It is a good practice to include the viewport in the meta tag in all the webpages to make sure it has the responsive capabilities

Using viewport in the <head> tag

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

Parts:

1. `width=device-width`

   Set the width of page to follow the screen width of the device

1. `initial-scale=1.0`

   Set the zoom level to 1 when page is first loaded by browser

Example: without viewport

```
<!DOCTYPE html>
<html lang="en">
<head>
    <style>
        img {
            width: 50%;
            height: 100%;
            object-fit: contain;
        }
    </style>
</head>
<body>
    <div>
        <img src="picture.jpg" alt="Garden"/>
    </div>
    <div>
        This is a garden with a beautiful view. In the middle of the garden we have a cottage and we could take a rest inside. This could be a fun experience to begin with!
    </div>
</body>
</html>
```

This is a garden with a beautiful view. In the middle of the garden we have a cottage and we could take a rest inside. This could be a fun experience to begin with!

Example: with viewport

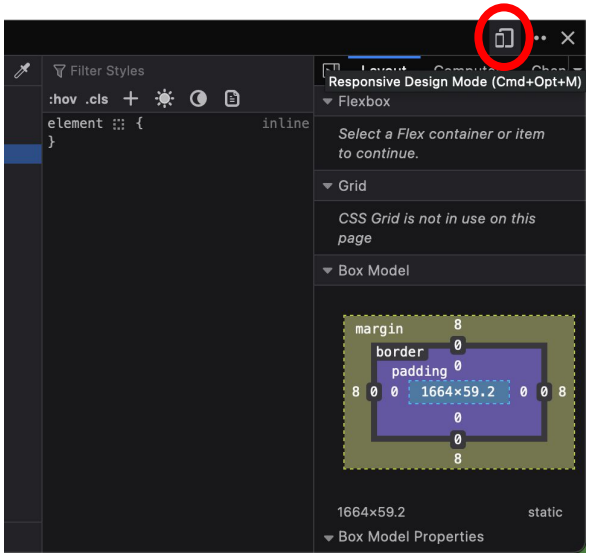
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" >
    <style>
        img {
            width: 50%;
            height: 100%;
            object-fit: contain;
        }
    </style>
</head>
<body>
    <div>
        <img src="picture.jpg" alt="Garden"/>
    </div>
    <div>
        This is a garden with a beautiful view. In the middle of the garden we have a cottage and we could take a rest inside. This could be a fun experience to begin with!
    </div>
</body>
</html>
```

This is a garden with a beautiful view. In the middle of the garden we have a cottage and we could take a rest inside. This could be a fun experience to begin with!

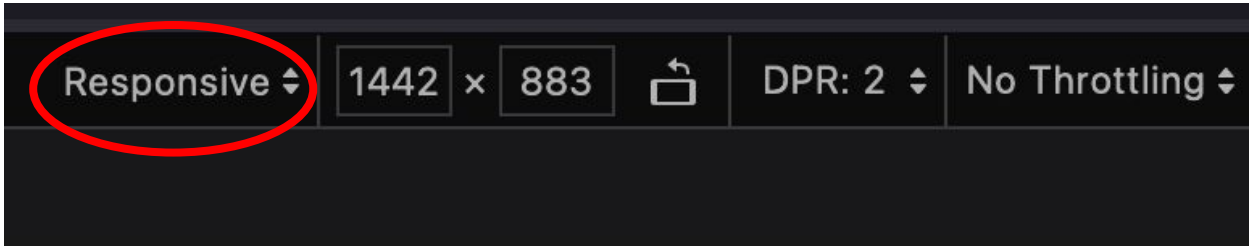
Trick

To resize the screen size, we could change the browser window size or use this steps:

1. Right click on the page
2. Select Inspect or Inspect Element depending on the browsers
3. Click the Responsive Design Mode icon



NOTE: select the size here



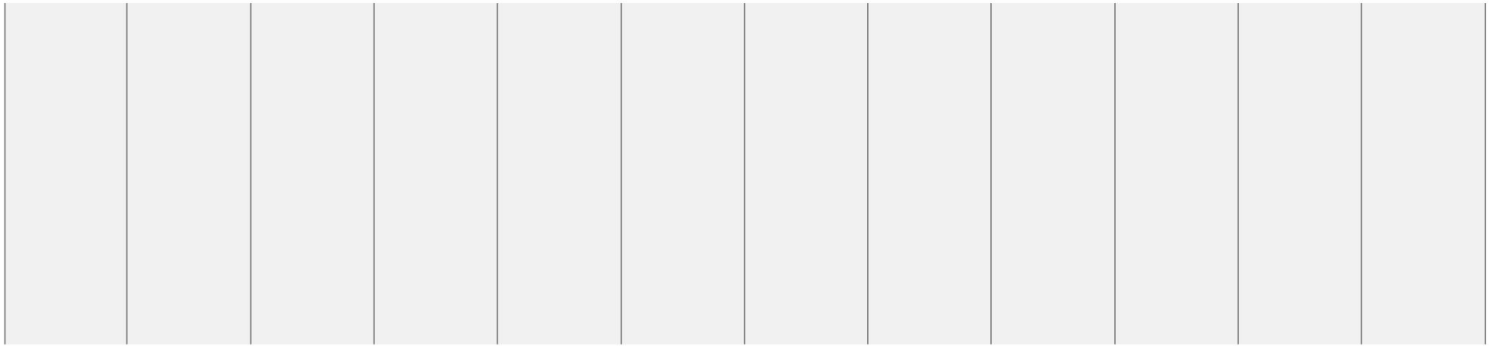
Rules to consider

1. Never use large fixed width elements
2. Never rely on a certain viewport size
3. Use CSS media queries to apply different style for small and large screens

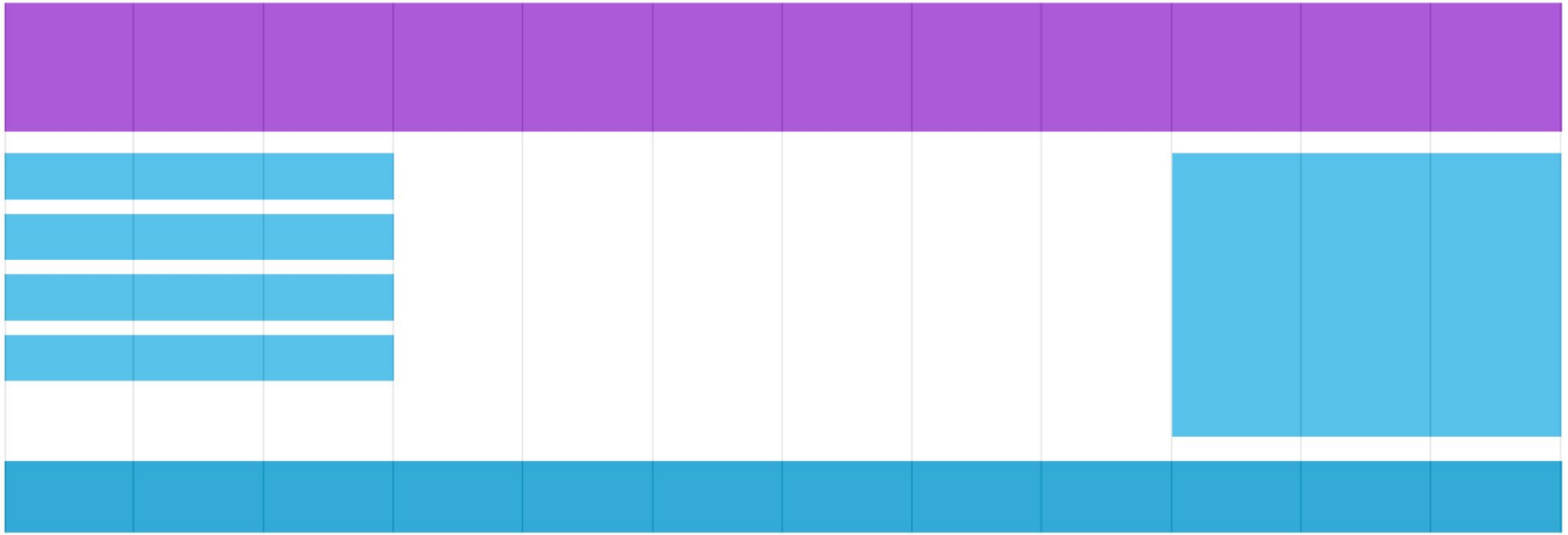
CSS Grid View

- Pages are usually designed as grid view
- The sections of the pages are separated by columns
- Makes it easier to place components on a page
- Responsive grid design has 12 columns in total with the total 100% width*

*The reason number 12 is used because it provides flexible number of possibilities such as 1, 2, 3, 4, 6, 12 allowing multitude of design options



CSS Grid View



Non-Responsive Design and how to countermeasure

When we add width, height and padding, the actual calculation is like this:

1. Actual width = width + padding + border
2. Actual height = height + padding + border

To countermeasure this, we could use `box-sizing` property and set the value to `border-box`

Example of not using border-box

```
<head>
  <style>
  .div1 {
   width: 300px;
   height: 100px;
   border: 1px solid blue;
  }

  .div2 {
   width: 300px;
   height: 100px;
   padding: 50px;
   border: 1px solid red;
  }
  </style>
</head>
<body>
  <h1>Without box-sizing</h1>
  <div class="div1">This div is smaller (width is 300px and height is 100px).</div>
  <br/>
  <div class="div2">This div is bigger (width is also 300px and height is 100px).</div>
</body>
```

Same width and height but why different?

Without box-sizing

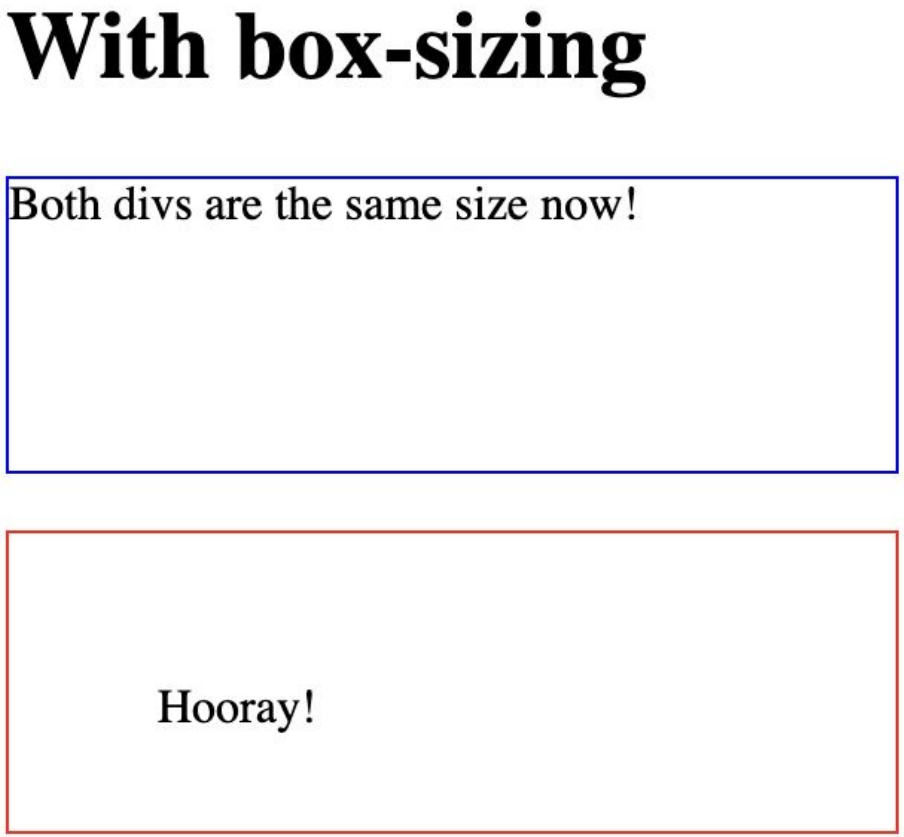
This div is smaller (width is 300px and height is 100px).

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

Note:
Actual width = padding + width + border
= 50px * 2 + 300px + 1px * 2
= 100px + 300px + 2px
= 402 px
Actual height = padding + height + border
= 50px * 2 + 100px + 1px * 2
= 100px + 100px + 2px
= 202 px

With box-sizing

```
<head>
 <style>
 .div1 {
   width: 300px;
   height: 100px;
   border: 1px solid blue;
   box-sizing: border-box;
 }
 .div2 {
   width: 300px;
   height: 100px;
   padding: 50px;
   border: 1px solid red;
   box-sizing: border-box;
 }
 </style>
</head>
<body>
 <h1>With box-sizing</h1>
 <div class="div1">Both divs are the same size now!</div>
 <br/>
 <div class="div2">Hooray!</div>
</body>
```


With box-sizing
Both divs are the same size now!
Hooray!

Media Query

- Technique in CSS to design responsive web pages
- Use this to include a block of CSS based on a certain condition
- Always remember to design from the smallest device screen first, so start from

Mobile Phones ➜ Tablets ➜ Desktop

- Could also manipulate the orientation of the screens: portrait and landscape

Typical Device Breakpoints

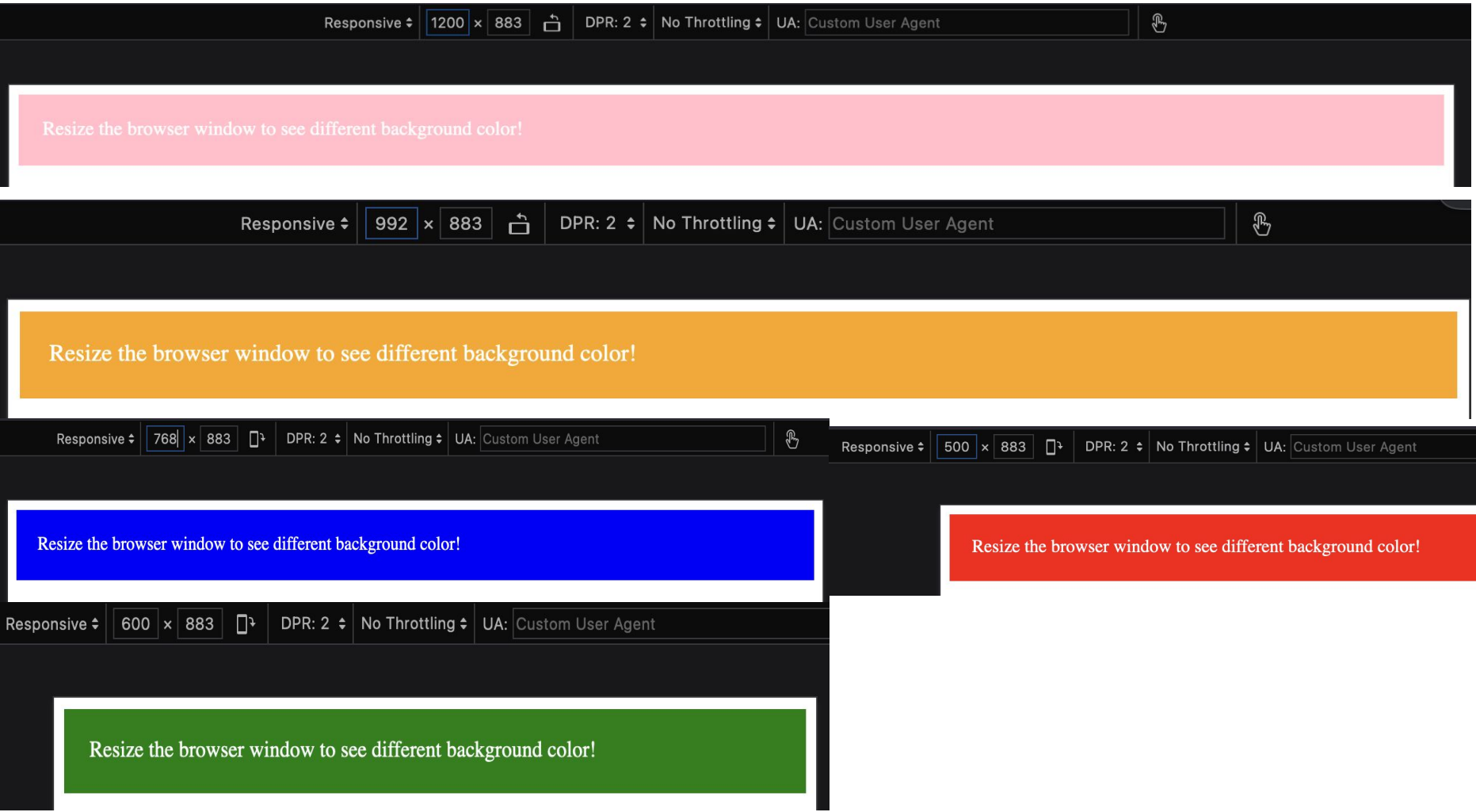
```
/* Extra small devices (phones, 600px and down) */
@media only screen and (max-width: 600px) {
    /* css here */
}

/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (min-width: 600px) {
    /* css here */
}

/* Medium devices (landscape tablets, 768px and up) */
@media only screen and (min-width: 768px) {
    /* css here */
}

/* Large devices (laptops/desktops, 992px and up) */
@media only screen and (min-width: 992px) {
    /* css here */
}

/* Extra large devices (large laptops and desktops, 1200px and up) */
@media only screen and (min-width: 1200px) {
    /* css here */
}
```

Example of responsiveness: CSS Part (style.css)

```css
div {
    padding: 20px;
    color: white;
}
/* Extra small devices (phones, 600px and down) */
@media only screen and (max-width: 600px) {
    div {background: red;}
}
/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (min-width: 600px) {
    div {background: green;}
}
/* Medium devices (landscape tablets, 768px and up) */
@media only screen and (min-width: 768px) {
    div {background: blue;}
}
/* Large devices (laptops/desktops, 992px and up) */
@media only screen and (min-width: 992px) {
    div {background: orange;}
}
/* Extra large devices (large laptops and desktops, 1200px and up) */
@media only screen and (min-width: 1200px) {
    div {background: pink;}
}
```

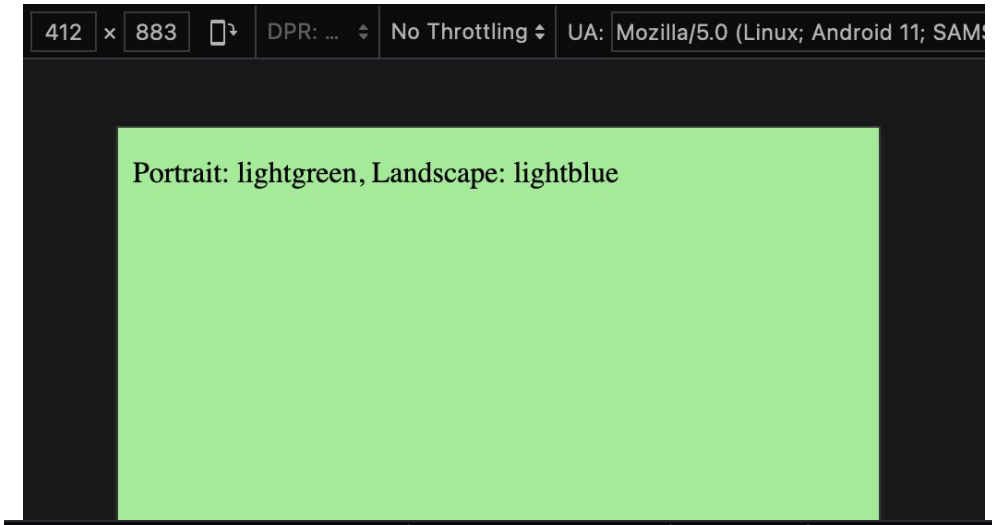
Example of responsiveness: HTML part

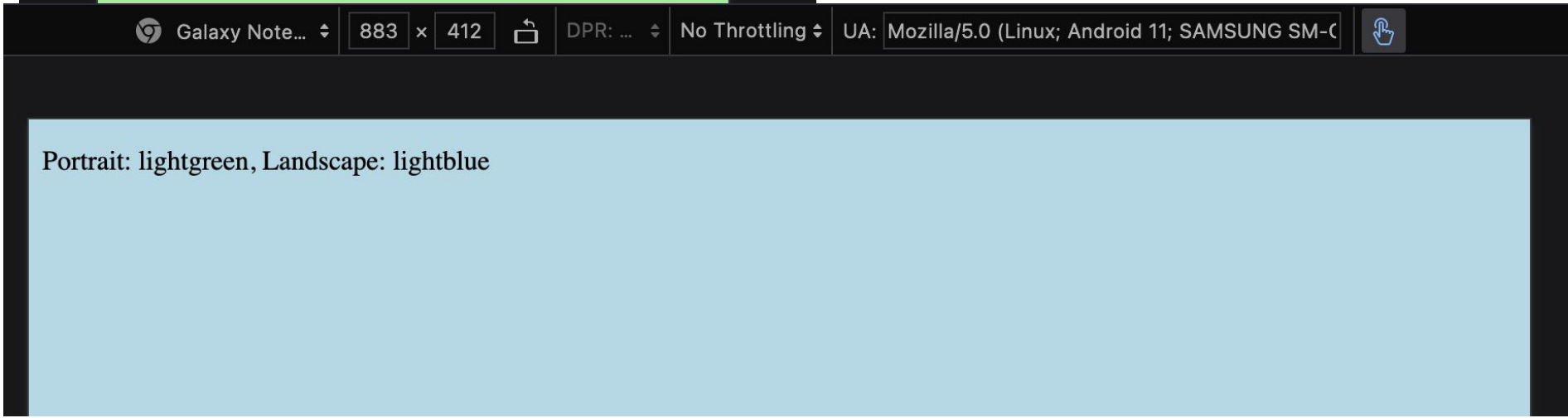
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Responsive Web Design</title>
    <link href="style.css" rel="stylesheet" />
</head>
<body>
    <div>Resize the browser window to see different background color!</div>
</body>
</html>
```


Responsive 1200 x 883 DPR: 2 No Throttling UA: Custom User Agent
Resize the browser window to see different background color!
Responsive 992 x 883 DPR: 2 No Throttling UA: Custom User Agent
Resize the browser window to see different background color!
Responsive 768 x 883 DPR: 2 No Throttling UA: Custom User Agent
Resize the browser window to see different background color!
Responsive 500 x 883 DPR: 2 No Throttling UA: Custom User Agent
Resize the browser window to see different background color!
Responsive 600 x 883 DPR: 2 No Throttling UA: Custom User Agent
Resize the browser window to see different background color!

Orientation: Portrait and Landscape

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0" >
<style>
body {
 background-color: lightgreen;
}
@media only screen and (orientation: landscape) {
 body {
   background-color: lightblue;
 }
}
</style>
</head>
<body>
<p>Portrait: lightgreen, Landscape: lightblue</ p>
</body>
</html>
```


412 × 883
DPR: ...
No Throttling
UA: Mozilla/5.0 (Linux; Android 11; SAM
Portrait: lightgreen, Landscape: lightblue


Galaxy Note...
883 × 412
DPR: ...
No Throttling
UA: Mozilla/5.0 (Linux; Android 11; SAMSUNG SM-(
Portrait: lightgreen, Landscape: lightblue

Steps of creating a responsive web

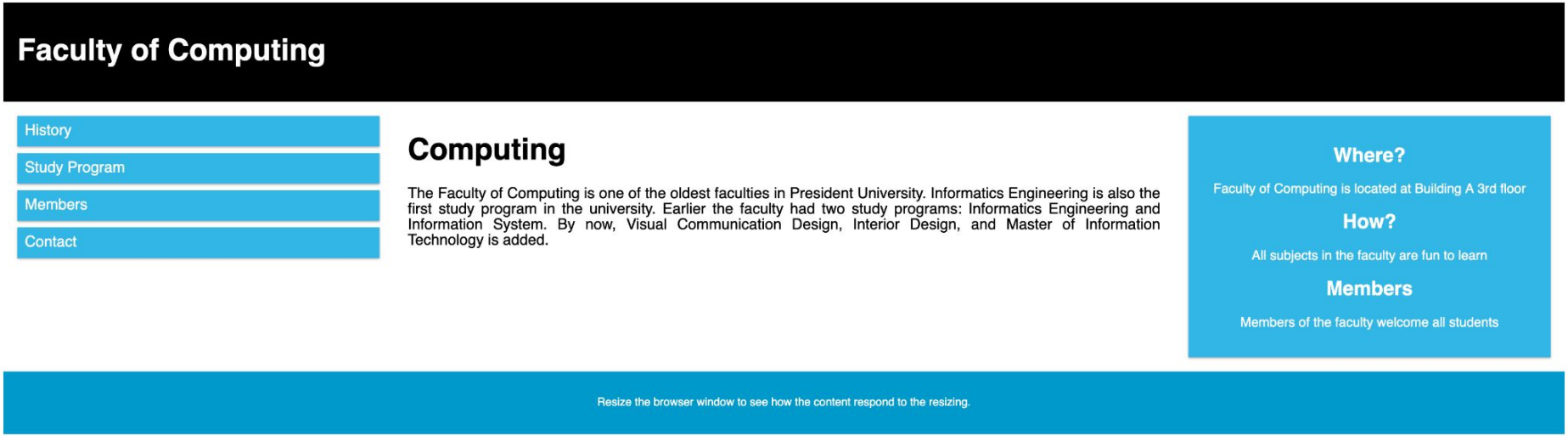
1. Ensure all HTML elements have `box-sizing` property set to `border-box`
2. Think about how to design the page contents, how many columns or rows and the percentage
3. Calculate responsive columns percentage
4. Use media query to split device breakpoints

Columns percentage assuming 12 columns

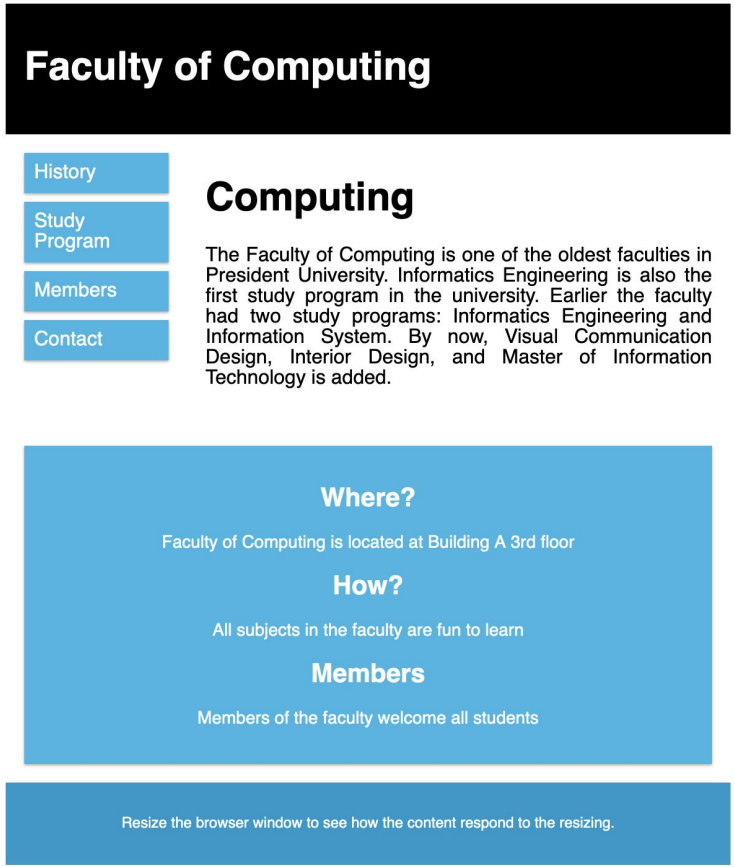
Width of a screen is considered as 100%. There are 12 columns, so

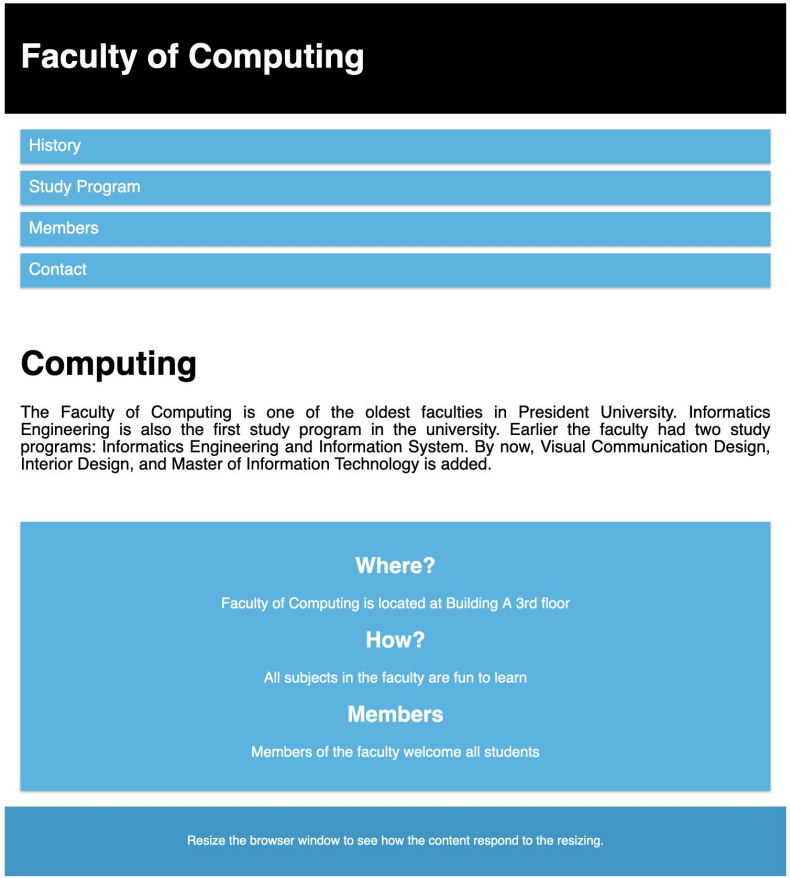
1 column = 100 / 12	= 8.33%
2 columns	= 16.66%
3 columns	= 25%
4 columns	= 33.33%
5 columns	= 41.66%
6 columns	= 50%
7 columns	= 58.33%
8 columns	= 61.66%
9 columns	= 75%
10 columns	= 83.33%
11 columns	= 91.66%
12 columns	= 100%

Target Desktop



Target Tablet and Mobile





File: style.css

```css
* {
    box-sizing: border-box;
}

.row::after {
    content: "";
    clear: both;
    display: table;
}

/* for all classes with col- */
[class*="col-"] {
    float: left;
    padding: 15px;
}

html {
    font-family: "Lucida Sans", sans-serif;
}
```

```css
.main {
    text-align: justify;
}

.header {
    background-color: black;
    color: white;
    padding: 15px;
}

.menu ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
}

.menu li {
    padding: 8px;
    margin-bottom: 7px;
    background-color: #33B5E5;
    color: white;
    box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
}
```

```css
.menu li:hover {
    background-color: #0099CC;
}

.aside {
    background-color: #33B5E5;
    padding: 15px;
    color: white;
    text-align: center;
    font-size: 14px;
    box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
}

.footer {
    background-color: #0099CC;
    color: white;
    text-align: center;
    font-size: 12px;
    padding: 15px;
}
```

```css
/* mobile first */
[class*="col-"] {
    width: 100%;
}

/* for tablets */
@media only screen and (min-width: 600px) {
    .col-s-1 {width: 8.33%;}
    .col-s-2 {width: 16.66%;}
    .col-s-3 {width: 25%;}
    .col-s-4 {width: 33.33%;}
    .col-s-5 {width: 41.66%;}
    .col-s-6 {width: 50%;}
    .col-s-7 {width: 58.33%;}
    .col-s-8 {width: 66.66%;}
    .col-s-9 {width: 75%;}
    .col-s-10 {width: 83.33%;}
    .col-s-11 {width: 91.66%;}
    .col-s-12 {width: 100%;}
}
```

```css
/* for desktops */
@media only screen and (min-width: 768px) {
    .col-1 {width: 8.33%;}
    .col-2 {width: 16.66%;}
    .col-3 {width: 25%;}
    .col-4 {width: 33.33%;}
    .col-5 {width: 41.66%;}
    .col-6 {width: 50%;}
    .col-7 {width: 58.33%;}
    .col-8 {width: 66.66%;}
    .col-9 {width: 75%;}
    .col-10 {width: 83.33%;}
    .col-11 {width: 91.66%;}
    .col-12 {width: 100%;}
}
```

File: index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Responsive Web Design</title>
    <link href="style.css" rel="stylesheet"/>
</head>
<body>
    <div class="header">
        <h1>Faculty of Computing</h1>
    </div>
```

```
    <div class="row">
        <div class="col-3 col-s-3 menu">
            <ul>
                <li>History</li>
                <li>Study Program</li>
                <li>Members</li>
                <li>Contact</li>
            </ul>
        </div>

        <div class="col-6 col-s-9 main">
            <h1>Computing</h1>
            <p>The Faculty of Computing is one of the oldest faculties in President
University. Informatics Engineering is also the first study program in the university.
Earlier the faculty had two study programs: Informatics Engineering and Information System
By now, Visual Communication Design, Interior Design, and Master of Information Technology
is added.</p>
        </div>
```

```
        <div class="col-3 col-s-12">
            <div class="aside">
                <h2>Where?</h2>
                <p>Faculty of Computing is located at Building A 3rd floor</p>
                <h2>How?</h2>
                <p>All subjects in the faculty are fun to learn</p>
                <h2>Members</h2>
                <p>Members of the faculty welcome all students</p>
            </div>
        </div>
    </div>

    <div class="footer">
        <p>Resize the browser window to see how the content respond to the resizing.</p>
    </div>
</body>
</html>
```

Now try resizing the page

- The result will vary depending on the screen pixels
- This is not limited to only three devices, design on your needs

Real-World CSS

1. Bootstrap https://getbootstrap.com/
2. Uno CSS https://unocss.dev/
3. Tailwind CSS https://tailwindcss.com/
4. Pico CSS https://picocss.com/
5. SASS https://sass-lang.com/
6. LESS https://lesscss.org/
7. W3CSS https://www.w3schools.com/w3css/w3css_intro.asp
8. etc

References

https://www.w3schools.com/css/css_rwd_intro.asp