WEBDESIGN & -PROGRAMMIERUNG
FH HAGENBERG // WS 2018/19 // BLOCK 1 // HTML & CSS

RESPONSIVE DESIGN

WHAT?

- „Responsive web design“ (RWD) describes an approach to make websites look good on a variety of devices, independent of their screen sizes.
- Ethan Marcotte coined the term in his „A List Apart“ article in 2010.

WHY?

- … because using a browser on a desktop computer with a certain minimum screen size is not the only possible way to consume websites anymore.

HOW?

- Media queries
  - CSS module to target specific media types (screen, print, projection, …) and features (min-width, max-width, resolution, device-aspect-ratio, …)
- Responsive images
  - Make images resize on smaller screens and only load the size you need using the `srcset` attribute or the `<picture>` tag.
- Fluid typography
  - Make the font size adapt to the viewport width.

STEP BY STEP

- Make a plan/design how the website should look on different screen sizes.
- Find a HTML structure that fits the designs:
  - Think about how to group and nest HTML elements.
  - Think about semantics and source order.
- Implement the CSS:
  - Use e.g. flexbox and grid for the layout and media queries to target different screen sizes.

START MOBILE FIRST

- You are forced to reduce.
- It’s easier to add things later than to remove them.
- You have to focus on the most important content first.
- Sometimes it’s easier, because the possibilities and the space are limited.
- You automatically have a fallback if media queries are not supported.

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

CSS MEDIA QUERIES

- Consist of a **media type** (screen, print, tv, …) and one or more **features** that can be either true or false.
  - Features can be combined by using logical operators.

- If the media type matches and all features evaluate to true the corresponding styles are applied.

CSS MEDIA QUERY WITHIN A STYLESHEET

```
@media screen and (min-width: 480px) and
(orientation: landscape) {

  .sidebar {
    background-color: #efefef;
  }
}
```

CSS MEDIA QUERY ON A LINK ELEMENT

```
<link rel="stylesheet"
      media="(max-width: 480px)"
      href="mobile.css">

<link rel="stylesheet"
      media="print"
      href="print.css">
```

CSS MEDIA QUERIES & MOBILE FIRST

```
.sidebar {
  width: 100%;
}

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

  .sidebar {
    max-width: 400px;
  }
}
```

RESPONSIVE IMAGES

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

- We don’t want to display a large image on small screens (and load more data than necessary) or have to scale up a small image on large screens.
- We always want to see a perfectly cropped image if there is not enough space to show the full picture.

SRCSET & SIZES ATTRIBUTES

- To provide images with different sizes and help the browser to choose the right one for the current screen size.

```
<img src="appletree.jpg" alt="An apple tree in autumn."
     srcset="appletree-400.jpg 400w,
             appletree-800.jpg 800w,
             appletree-1200.jpg 1200w"
     sizes="(max-width: 800px) 100vw,
            (max-width: 1000px) 80vw,
            60vw">
```

PICTURE ELEMENT

- To provide differently cropped images for different screen sizes.

```
<picture>
  <source media="(max-width: 600px)" srcset="appletree-portrait.jpg">
  <source media="(min-width: 601px)" srcset="appletree.jpg">
  <img src="appletree.jpg.jpg" alt="An apple tree in autumn.">
</picture>
```

FLUID FONT SIZES

- Set the font size depending on the viewport width.

```
h1 {
  font-size: 20vw;
}
```

- Set the font size for selected viewport width values.
  - 32px (min) on a 360px viewport; 44px on a 1920px viewport

```
h1 {
  font-size: calc(32px + (44 - 32) * (100vw - 360px) / (1920 - 360));
}
```

RESPONSIVE DESIGN – SUMMARY

- Websites nowadays should look good on a variety of screen sizes.
- Thinking mobile first has some advantages.
- First plan – then implement.
- Don’t forget to add the viewport meta information.
- Use CSS media queries to target different media types and screen sizes.
- Think about the content and use e.g. the srcset attribute or the picture element.
- Use fluid font sizes.