Web design – Computer lab 5

Ing. Pavol Dubovec

Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2, 612 00 Brno – Královo Pole

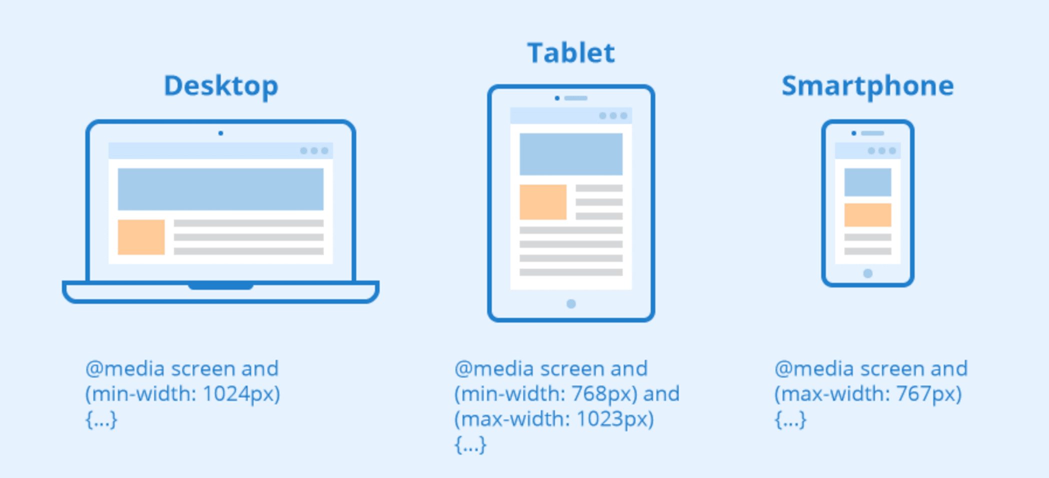
idubovec@fit.vut.cz

April 14, 2025

Sources for lab 5

- Lab 5 – section media queries
- Lecture 8 – media queries
- MDN – Media Queries
- W3Schools – Media Queries
- CSS-Tricks – Media Query Guide
- HTML/CSS – Google Style Guides
- MDN – Container Queries

Motivation



```
@media screen and
(min-width: 1024px)
{...}
```

```
@media screen and
(min-width: 768px) and
(max-width: 1023px)
{...}
```

```
@media screen and
(max-width: 767px)
{...}
```

Figure: Media queries examples. Source.

Task summary

1. Start from the document you have created in the previous lab (lab 4).
2. Find lab 5 on the labs website.
3. Adjust the document to display well on devices with different resolutions.
4. Consider the following device classes:
   - extra small (XS) $<$ 576 px
     - all in one column that takes up 100% of the document width
   - small (SM) $\geq$ 576 px – **mobile phone**
     - all in one column that takes up 100% of the document width
   - medium (MD) $\geq$ 768 px – **tablet**
     - two columns in ratio 7 (left): 5 (right)
   - large (LG) $\geq$ 992 px – **laptop**
     - three columns in ratio 4 (left): 4 (middle): 4 (right)
   - extra large (XL) $\geq$ 1200 px – **desktop**
     - three columns in ratio 4 (left): 4 (middle): 4 (right)
     - maximum content width: 1200px
   - extra extra large (XXL) $\geq$ 1400 px – **widescreen**
     - three columns in ratio 4 (left): 4 (middle): 4 (right)
     - maximum content width: 1320px

Document Design Specifications

- **CSS Variables:** Use CSS custom properties for consistent styling
  - Define colors, sizes, and spacing in: root
  - Example: –maxPageWidth: 1200px; –topLinkFontSize: 4rem;
- **Fonts:** Ubuntu from Google Fonts.
- **Colors:**
  - Text: #656565 (–colorText)
  - White text: white (–colEmphasizeMax)
  - Brown: #b15e23 (–colEmphasizeHigh)
  - Dark brown: #3c1f0a (–colEmphasizeMedium)
  - Light brown: #d5a17c (–colEmphasizeLight)
- **Background elements:**
  - Header/footer: `--bgImg: url('../img/footer_bg.gif');`
  - Navigation: Brown with dark brown hover effect
  - About/Exercises sections: Gradient from light gray to white

HTML Structure Guidelines

- **Viewport meta tag:** Essential for responsiveness
  - `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
- **Layout wrapper:** Create a container class for width control
  - Apply to each content section for consistent layout
  - Set max-width with auto margins: `max-width: var(--maxPageWidth); margin: auto;`
- **Semantic HTML structure:**
  - Use `<header>`, `<nav>`, `<section>`, `<footer>` elements
  - Group related content in block divs: `<div id="blockContent">`
  - Use class naming conventions for component types: `.divPrednaska`, `.divCviceni`
- **Image handling:**
  - Float images and apply appropriate margins
  - Plan for text wrapping around images

CSS Grid Implementation

- **Grid system approach:**
  - Create a simple 12-column grid system using percentage widths
  - Define column classes for all breakpoints: `[class*="col-"] { float: left; width: 100%; }`
  - Clear floats with ::after pseudo-element
- **Media query implementation:**
  - Start with mobile-first approach (default is single column)
  - Apply column widths at each breakpoint:
  - `@media (min-width: 576px) { .col-sm-6 { width: 50%; } ...}`
  - `@media (min-width: 768px) { .col-md-7 { ...} ...}`
  - `@media (min-width: 992px) { .col-lg-4 { ...} ...}`
  - `@media (min-width: 1200px) { .col-xl-4 { ...} ...}`
- **Layout adjustments:**
  - Use flex properties at tablet size to ensure equal heights
  - Modify gradient directions based on screen size (e.g., vertical on mobile, horizontal on tablet)
  - Consider adding media query for widescreen (XXL) devices ($\geq$ 1400 px)

Advanced Responsive Techniques

- **Box sizing reset:**
  - Apply `box-sizing: border-box;` to all elements
  - Ensures padding and borders are included in width calculations
- **Responsive images:**
  - Use floating techniques for smaller screens
  - Clear floats properly to avoid layout issues
- **Navigation considerations:**
  - Horizontal navigation for larger screens
  - Consider a hamburger menu pattern for smaller screens (next lab)
- **Testing methods:**
  - Use browser dev tools to test different screen sizes
  - Check layout at each breakpoint
  - Verify that content is accessible at all screen sizes
- **Further exploration:**
  - Compare your custom solution with frameworks like Bootstrap
  - Consider container queries for component-based responsiveness (newer CSS feature)