

Adaptive markup

Lesson 13



Lesson Plan

1

Introduction

2

Adaptive layout

3

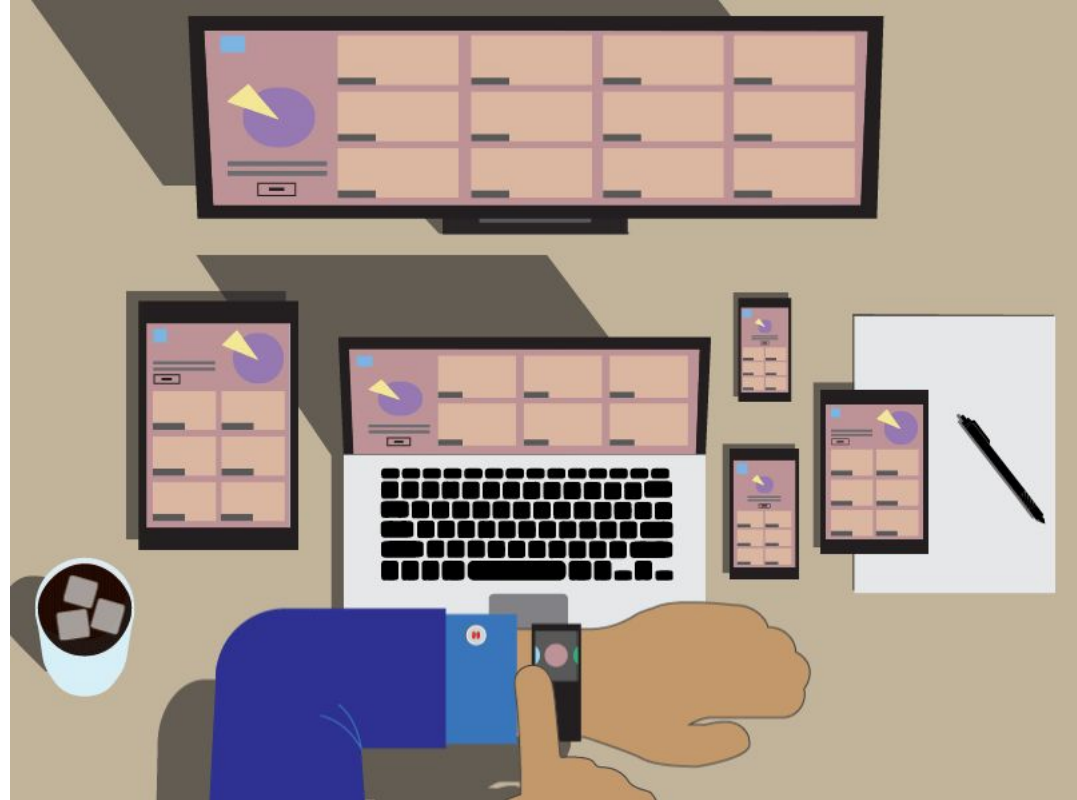
Responsive layout

4

Tools and best practices

Adaptive markup

- ◆ **Adaptive markup** is a set of techniques that we use to make our site adapt to user's device
- ◆ Additionally, it makes our web site look nice regardless of the content



Why is it important?

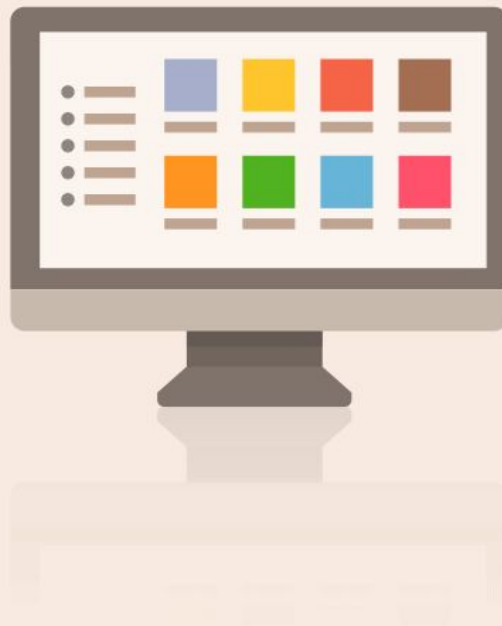
📌 Over 61.5% of global internet traffic comes from mobile devices

📌 Adaptive markup enhances UX

📌 Adaptiveness influences SEO

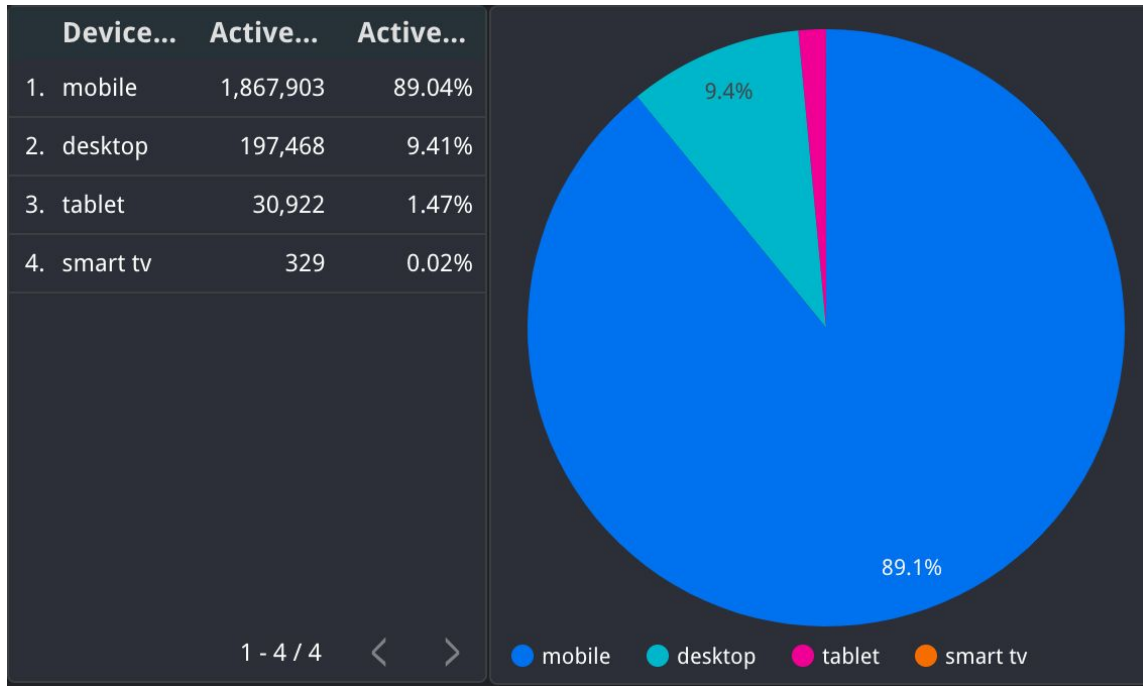
📌 Faster initial loading

- LCP (Largest Contentful Paint)
- FCP (First Contentful Paint)
- FID (First Input Delay)
- CLS (Cumulative Layout Shift)



Mobile VS desktop

90% mobile
10% desktop



Android



~56%

our company

~33%

~45%

worldwide

~18%

Responsive VS Adaptive

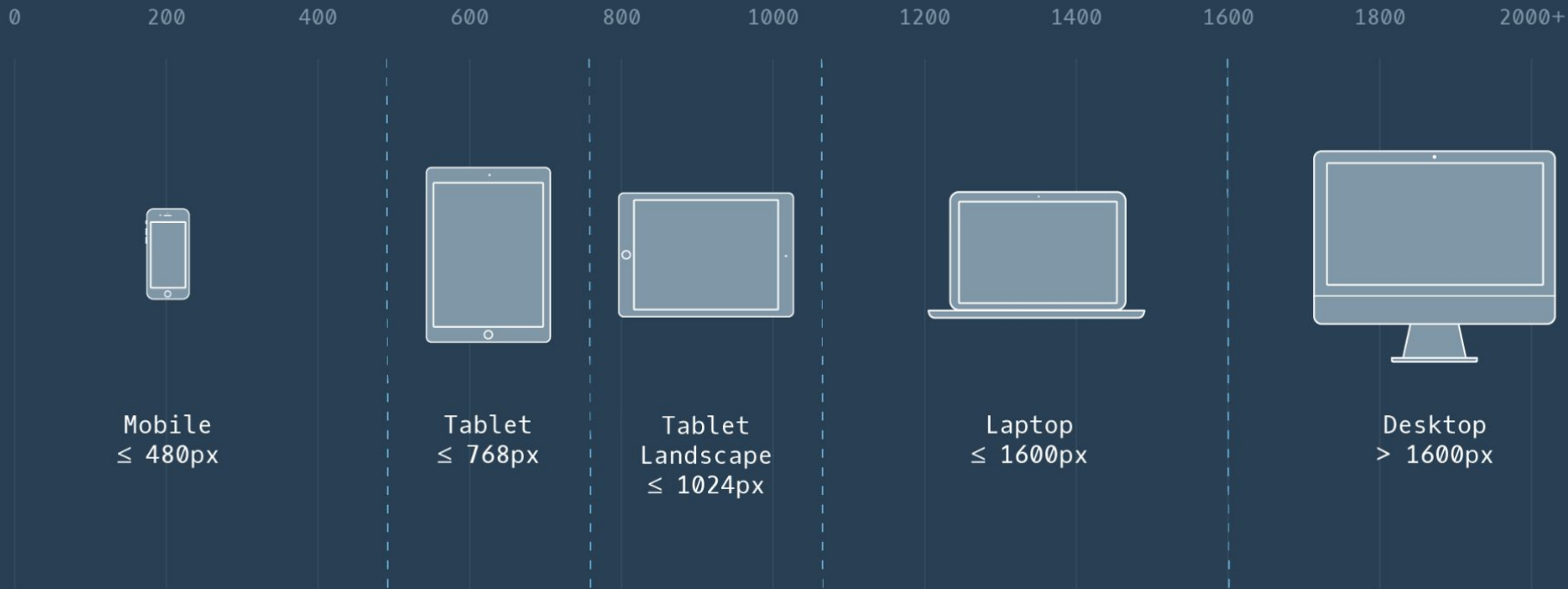
Responsive layout uses flexible grids and CSS to dynamically adjust content across all screen sizes.



Adaptive layout uses fixed layouts designed for specific screen sizes



Screen sizes



Screen Resolution

< 375px

~20%

> 375px

~80%

90% all devices are 360px-414px

Viewport meta tag

example →

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

- 💡 **width** - min viewport width
- 💡 **initial-scale** - zoom level on load
- 💡 **minimum-scale** - min zoom, default is 0.1
- 💡 **maximum-scale** - max zoom, default is 10
- 💡 **user-scalable** - defines if a site can be scalable; 0 - no, 1- yes (default)



This page has a viewport width of 980 pixels. This is how it looks at 100% zoom on an HTC Desire.



The viewport is still 980px wide, but since the page is zoomed out it is squeezed into 480px, the resolution of the phone's screen.

Media queries

@media (condition) { /* CSS styles */ }

device's media type (print, screen, speech)	@media print @media screen	<u>example</u>
orientation (landscape, portrait)	@media (orientation: portrait)	<u>example</u>
aspect ratio	@media (max-aspect-ratio: 3/2)	
viewport size	@media (max-width: 768px) @media (min-width: 320px) and (max-width: 480px)	<u>example</u>
user preferences	@media (prefers-reduced-motion) @media (prefers-reduced-transparency)	<u>example</u>
screen resolution	@media (min-resolution: 192dpi)	
device pointer type	@media (pointer: coarse) @media (pointer: fine)	

Responsive layout

example →

✓ Using relative units

(%, em, rem, vw, vh)

Relative Units



Static Units



✓ Flex container

```
.square { flex-grow: 1; }  
.square#three { flex-grow: 1; }
```



✓ Grid container



Adaptive images

example →

Object-fit & Background-size

- ✓ fill
- ✓ contain
- ✓ cover
- ✓ scale-down
- ✓ none

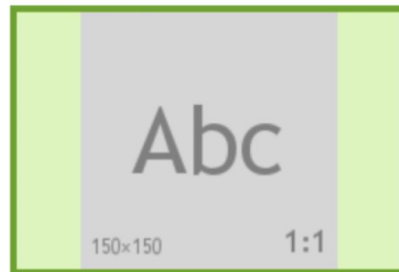
**Original
image**



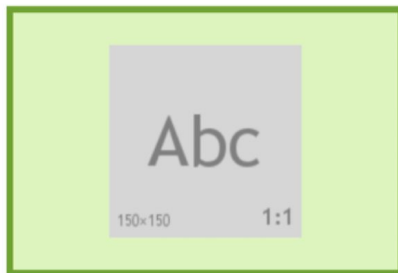
object-fit:fill



object-fit:contain



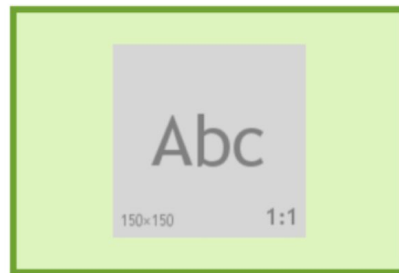
object-fit:scale-down



object-fit:cover



object-fit:none



Adaptive images

example →

Srcset, Sizes & Picture tag

	<code></code>	<code><picture></code>
supports several image formats	✗	✓ via <code><source></code>
adapts to viewport width	✓ via <code>srcset</code> & <code>sizes</code>	✓ via <code>media</code> in <code><source></code>
supports retina/HDPI	✓ via <code>srcset</code>	✓ via <code>srcset</code> in <code><source></code>
optimization type	«initial load» optimized	layout-driven media selection

Overflow

example →

visible

Lorem, ipsum
optiolaffsdfsdffssdiufhisdf
sit amet consectetur
optiolaffsdfsdffssdiufhisdf
elit. Nostrum dicta
quis nihil
optiolaffsdfsdffssdiufhisdf
molestias
laboriosam
commodi

temporibus esse,
eveniet,
optiolaffssdiufhisdfush
sapiente odit. Cum
provident ea
impedit nisi earum
vitae voluptatem.

hidden

Lorem, ipsum
optiolaffsdfsdffssdiufhisdf
sit amet consectetur
optiolaffsdfsdffssdiufhisdf
elit. Nostrum dicta
quis nihil
optiolaffsdfsdffssdiufhisdf
molestias
laboriosam
commodi

scroll

Lorem, ipsum
optiolaffsdfsdffssdiufhisdf
sit amet
consectetur
optiolaffsdfsdffssdiufhisdf
elit. Nostrum dicta
quis nihil
optiolaffsdfsdffssdiufhisdf
molestias
laboriosam

auto

Lorem, ipsum
optiolaffsdfsdffssdiufhisdf
sit amet
consectetur
optiolaffsdfsdffssdiufhisdf
elit. Nostrum dicta
quis nihil
optiolaffsdfsdffssdiufhisdf
molestias
laboriosam

clip

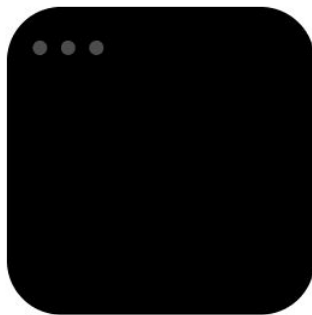
Lorem, ipsum
optiolaffsdfsdffssdiufhisdfush
sit amet consectetur
optiolaffsdfsdffssdiufhisdfush
elit. Nostrum dicta
quis nihil
optiolaffsdfsdffssdiufhisdfush
molestias
laboriosam
commodi

Mobile-first vs. Desktop-first

Design starts with mobile screens, then scales up.
Focus on essentials → faster, cleaner, smarter UI.

- 📱 Prioritize core content
- 📱 Add features progressively
- 📱 Improve performance on mobile
- 📱 Boost SEO with mobile indexing
- 📱 Clean UX across all devices

Desktop first



Mobile first



Common mistakes

- ✗ **Website scales instead of adapting** Always use `<meta name="viewport" ...>`
- ✗ **Disabled zoom** Avoid `user-scalable=no` to allow user zoom if needed
- ✗ **Duplicated HTML for layouts** Use one semantic structure, style via CSS/JS only
- ✗ **Too small elements on mobile** Use at least 14–16px for body text and sufficient element size - 44×44px (Apple) 48dp (Google)
- ✗ **Hover-only interactions** Menus and tooltips should work on tap/click instead of hover on mobile
- ✗ **Unoptimized images** Don't load full-HD on mobile. Use `srcset`, `picture`, WebP
- ✗ **Testing only on desktop** Test on real mobile devices, or use emulators (e.g., BrowserStack)
- ✗ **Improper use of vw/vh** `100vh` may cut off content
`100vw` + scrollbar = horizontal scroll

How to test and fix overflows

example →

Tool	What it does
💡 DevTools (Chrome, Firefox, Safari)	Emulate screen sizes, pixel density, orientation; test breakpoints
💡 Mobile OS Emulators	iOS Simulator, Android Emulator for OS-specific bugs
💡 BrowserStack / LambdaTest	Cloud-based cross-device/browser testing, localhost support
💡 Remote Debugging	Inspect real devices via Chrome or Safari
💡 Lighthouse & Validators	Audit font sizes, tap targets, performance & accessibility
💡 Screenshot Tools	Services like Screenfly or Am I Responsive for quick visual checks
💡 IDE Integrations	VSCode + BrowserSync = live preview on multiple screens
💡 Console Tools	Log viewport width, DPR, breakpoints <code>console.log(window.innerWidth)</code>

Quality Criteria for HTML Course

❤️ Mandatory for passing the course

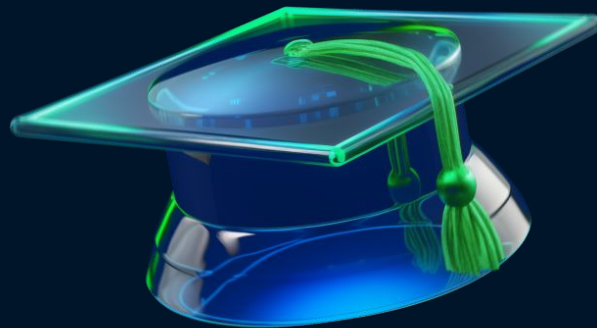
🟡 Required for the highest grade

🟢 Optional

- ❤️ Responsive adaptive layout is implemented.
- ❤️ No horizontal scroll on the full page.
- ❤️ Burger menu works on mobile.
- ❤️ Filter panel opens correctly on mobile.
- ❤️ Horizontal scroll is enabled for specific blocks.
- ❤️ Breakpoints are well chosen.
- ❤️ Consistent spacing and alignment.
- ❤️ Interactive elements are accessible on mobile.

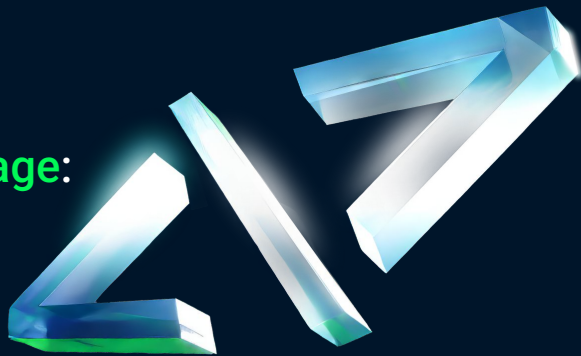
Summary

1. What is adaptive and responsive layout
2. What is media queries and their types
3. Adaptive images
4. Overflow
5. Mobile first approach
6. How to test and fix mistakes



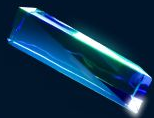
Homework

1. Add **adaptive layouts** for all pages
2. Enable horizontal scroll for blocks on **mobile homepage**:
 - catalog items
 - blog posts
 - team members
3. Ensure the **burger menu(all pages)** and **filter panel(catalog page)** open and work correctly on mobile devices.



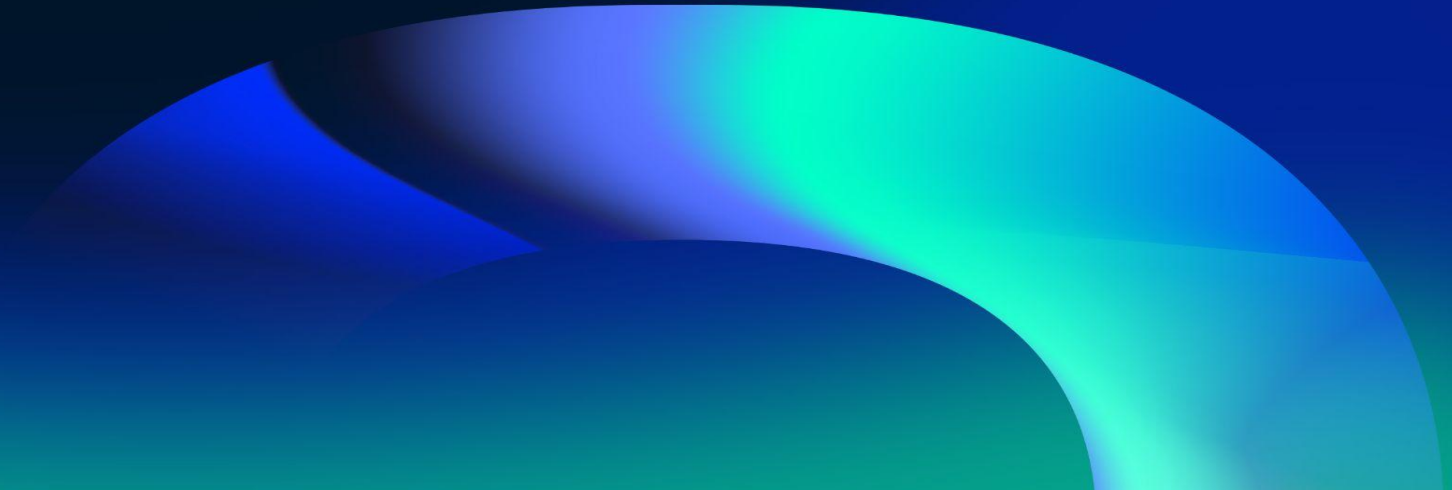
Test your layout on different screen sizes and pay close attention to spacing, alignment, and interactions. Avoid horizontal scroll on the entire page — make sure the layout fits the viewport properly.

B Academy
RO



QUESTIONS?

Please fill out the feedback form
It's very important for us





THANK YOU!

Have a good evening!