# Responzivní design

Co to je – intuitivně – design stránek, který se přizpůsobí podle toho, na kterém zařízení se dívám.

Mobil, laptop, desktop

Layout stránky kde jsou sloupce s hlavím textem, pak napravo a nalevo - na mobilu to nefunguje

Důvod – většinu uživatelů z mobilu, jak počet uživatelů, tak celkově čas strávený online

I když my to kódujeme na desktopu, je potřeba na to nezapomínat a podcenit to

Historie –

Desktop

Dvě verze (desktop a mobil)

Současnost   
  
Celá škála zařízení – hodinky, mobily, tablety, laptopy, desktopy, ultraširoké monitory, televize   
cíl – zajistit optimální uživatelský zážitek

informace o týmu etc, historie firmy, recenze od zákazníků, FAQ,   
  
Optimální -   
pohodlné kliknutí   
přizpůsobený obsah – mobil chce rychlou informaci

Desktop – informace o týmu etc, historie firmy, recenze od zákazníků, FAQ,

Méně animací, efektů   
přizpůsobená navigace

V responzivním designu – nekódujeme web pro konkrétní zařízení!

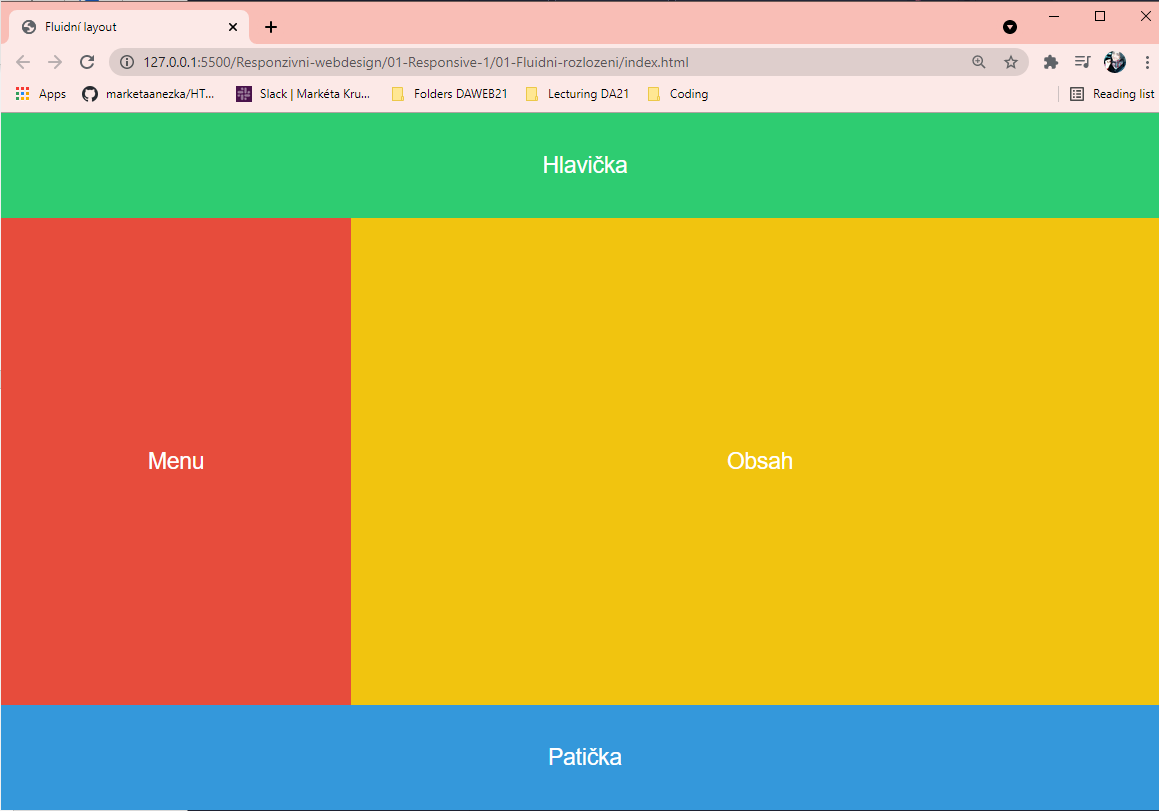
Jednotky

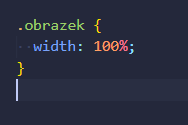
Font size – REM velikost relativne to root element

# Techniky RWD

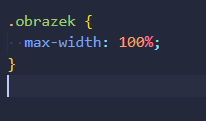
3 pilíře

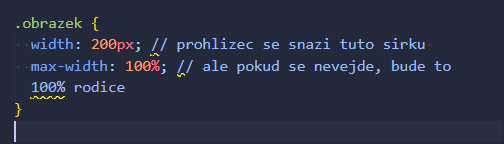
## Flexibilní grid

* Mřížka – vertikální linie, kam design klademe, aby byly prvky dobře zarovnané
* Prvky zabírají jednotky mřížky
* Nevíme, jak bude velká obrazovka
* Musíme psát design, aby se přizpůsobil všem šířkám   
  UKÁZKA – 01 – NENÍ TO PLNĚ RESPONZIVNÍ, ALE REAGUJE TO NA ŠÍŘKY
* 

Flexibilní obrázky  
obrázky se automaticky nepřizpůsobují – je potřeba nastavit šířku   
je to snadné – width 100% je to z rodiče, i pokud to je body. Někdy max-width.   
Ideální – mít upravený obrázek pro malý displej – jiný výřez – pokročilejší, ale trvá to   
UKÁZKA -   
ODKOMENTOVAT PRVNÍ – VE STYLU NIC – přidat:   
  
 ikonky třeba chceme malé, ale většinou toto stačí

Druhá možnost – druhý obrázek odkomentovat – někdy chceme, aby se obrázek přizpůsobil, ale nechceme, aby zvětšoval NAD SVOU ORIGINÁLNÍ VELIKOST



Jde to kombinovat , pokud je rodič užší než 200px, tak se to zmenší   


IMG vs Background img v css

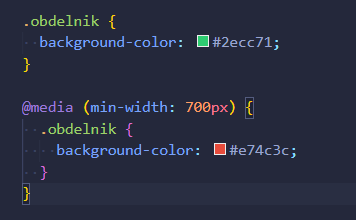
Img – je to součást obsahu (logo, diagram, etc.)   
– vezmou to čtečky pro lidi s horším zrakem, při zvětšení písma se zvětší i obrázek  
 google search (díky altu)   
Při tisku se vytiskne i obrázek

CSS background image – není to součást obsahu, pouze pro design, náhrada textu, opakování, roztahování

## Media queries

Reagujeme na šířku prohlížeče nejčastěji a na to se i naučíte   


Styly se aplikují, pokud je „podmínka“ splněna, lze dát min width i max width   
sem píšeme věci, které chceme, aby se ZMĚNILY, nemusíme kódovat celé CSS   
lze dodat i typ, pokud je to screen nebo print nebo speech  
min-width – to je moment, odkud se už obsah roztahuje nesmyslně a chceme, aby se ten layout nějak změnil. – BREAKPOINT   
**slide 46 – nejdůležitější** V media queries píšeme celý styl, ne jenom vlastnost!! Tu, co měníme na dané šířce   
UKÁZKA – OBDÉLNÍK **POZOR NA ZOOM!!!!!!!!**

****  
pokud podmínka není splněna – jako kdyby tam nic nebylo!   
pokud ano, přidá se to do CSS – a jelikož umíme specificitu, tak se to přebije, jelikož je to níž   
proto píšeme mobile first, nahoru   
vlasnost **display: none** – něco lze skrýt většinou to je na mobilu – potom to musíme vrátit na správnou hodnotu zpátky

Pozor ADBLOCKER  
  
cviceni

01-sloupce

@media (min-width: 800px) {

  .kontejner {

    display: flex;

  }

  .sloupec {

    flex-basis: calc(100% / 3);

  }

}

02-skryvani obsahu

.banner {

  display: none;

}

@media (min-width: 800px) {

  .banner {

    display: block;

  }

}

03 -galerie

img {

  width: 100%;

}

@media (min-width: 500px) {

  .galerie {

    v

  }

  .foto {

    flex-basis: 50%;

  }

}

@media (min-width: 650px) {

  .foto {

    flex-basis: calc(100% / 3);

  }

}

@media (min-width: 800px) {

  .foto {

    flex-basis: 25%;

  }

}

@media (min-width: 1000px) {

  .foto {

    flex-basis: calc(100% / 6);

  }

}

Layout 01

@media (min-width: 640px) {

  .kontejner {

    display: flex;

    flex-wrap: wrap;

    justify-content: center;

  }

  .horni,

  .treti {

    flex: 0 0 100%;

  }

  .prvni,

  .druhy {

    flex: 0 0 50%;

  }

}

@media (min-width: 1000px) {

  .prvni,

  .druhy,

  .treti {

    flex: 0 0 calc(100% / 3);

  }

}

@media (min-width: 640px) {

  .obsah {

    display: flex;

    flex-wrap: wrap;

  }

  .menu {

    flex-basis: 30%;

  }

  .clanek {

    flex-basis: 70%;

  }

}

@media (min-width: 640px) {

  .menu,

  .reklama {

    flex-basis: 20%;

  }

  .clanek {

    flex-basis: 60%;

  }

}

Layout 03

.reklama1,

.reklama2 {

  margin: 20px 0;

}

@media (min-width: 640px) {

  .obsah {

    display: flex;

  }

  .clanek {

    flex: 0 0 60%;

  }

  .reklama {

    flex: 0 0 40%;

  }

}

@media (min-width: 1000px) {

  .kontejner {

    display: flex;

  }

  .menu {

    flex: 0 0 25%;

  }

  .obsah {

    flex: 0 0 75%;

    display: block;

  }

  .reklama {

    display: flex;

  }

  .reklama1,

  .reklama2 {

    margin: 0 10px;

    flex: 0 0 calc(50% - 20px);

  }

}