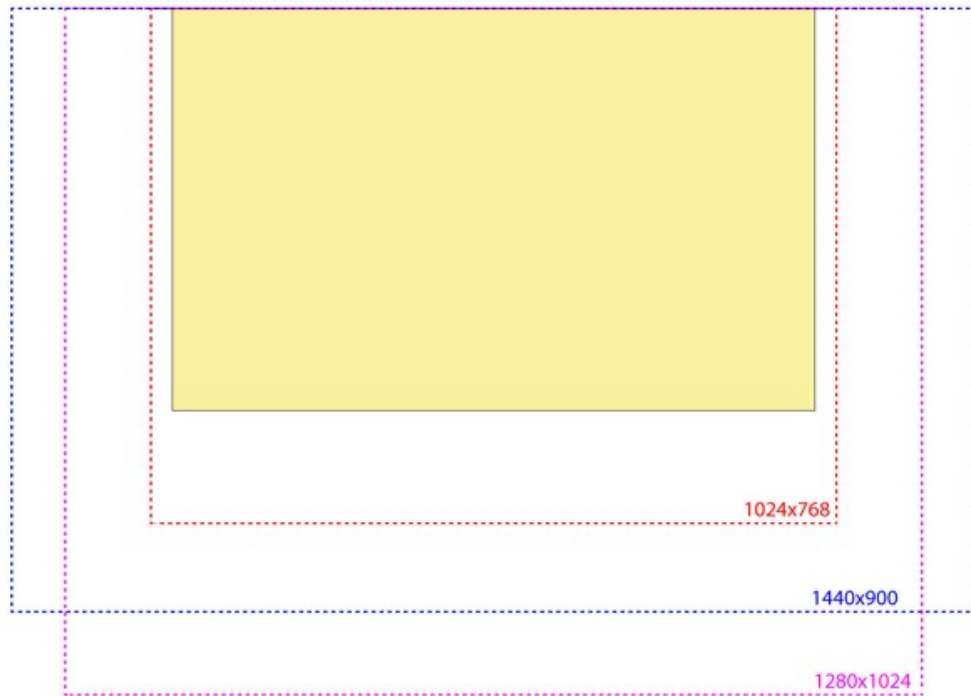


Responsive dizajn

- Do pre nekoliko godina, svet web dizajna u mnogim aspektima bio je jednostavniji.
- Mogli smo (*prošlo vreme*) govoriti o uslovnoj unificiranosti prikaza na webu. Pošto je većina korisnika koristila rezoluciju 1024x768 i slične, a nije bilo mobilnih i sličnih uređaja koji pristupaju netu, posao web dizajnera je bio prilično jednostavan, barem u pogledu određivanja osnovnih dimenzija sajtova.
- Danas su sasvim normalne i zastupljene full HD ili veće rezolucije čak i na mobilnim telefonima, dok računari i televizori dobijaju do skoro nezamislive 4k, pa i veće.

Kako je bilo ranije?

- Većina autora sajtova je prihvatila fiksnu širinu u pikselima kao nešto osnovno.
- U tom periodu ustalio se je i standard od (najčešće) 960px širine za osnovu sajta



Promene

- U poslednjih desetak godina, stvari su počele da se menjaju.
- Bili smo svedoci istovremenog povećanja i smanjivanja ekrana. Sa jedne strane, sve je više i više monitora i televizora (sada se i gubi polako razlika) koji imaju veće rezolucije, a sa druge strane je mnoštvo telefona, tablet i sličnih uređaja sa nesagledivim rasponom rezolucija.

Promene

- U takvom, izmenjenom stanju stvari, postalo je jasno da fiksna širina (od oko 960px) više nije adekvatno rešenje.
- Jasno je da isti sajt širine 960px neće istovremeno biti adekvatan i na iOS/Android telefonu i na fullHD televizoru.
- Pojavila se je ideja posebnog interneta za mobilne i slične uređaje koja je ubrzo propala. Takav pristup je uključivao kreiranje posebnog sajta za računare, i posebnog za telefone. Možda se i sećate tih sajtova, polovinom dvehiljaditih. Osnovni problem je bio razlika u sadržaju. Takvi sajtovi jednostavno nisu mogli da pruže isti doživljaj kao pune verzije sajtova i nikad nisu prihvaćeni u većoj meri. Ubrzo su nestali.

Promene

- Iz toga se rodila (i ostala) dominantna ideja **jednog weba (engl. One web)** u kome ne delimo korisnike na različite varijante istog sajta.
- Ali to znači da isti sajt treba da se prikaže i na telefonu sa rezolucijom 320x280 i na računaru sa full HD rezolucijom od 1920x1080.
- Ranije smo bili ograničeni tehnologijom, koja je u međuvremenu dostigla potreban nivo i taj problem je nestao. I zbog toga svakako da je bilo potrebno novo rešenje.

Responsive dizajn

- Rešenje je došlo u formi **prilagodljivog dizajna** (engl. ***Responsive design***). Termin je prvi upotrebio Ethan Marcotte u svome članku na sajtu *A list apart*:
<http://alistapart.com/article/responsive-web-design>
- Taj tekst je objavljen 2010. godine i pokrenuo je lavinu koja je promenila web. I danas smo svedoci neprestanih promena u sferi web dizajna.

Responsive dizajn

U pomenutom tekstu, Ethan (između ostalog) kaže:

U proteklim godinama sretao sam se sa više kompanija koje zahtevaju “iPhone sajt” kao deo njihovog projekta. Možemo izdvojiti prilagođene verzije na posebnim poddomenima, lokacijama distanciranim i odvojenim od “ne-iPhone verzije sajta”.

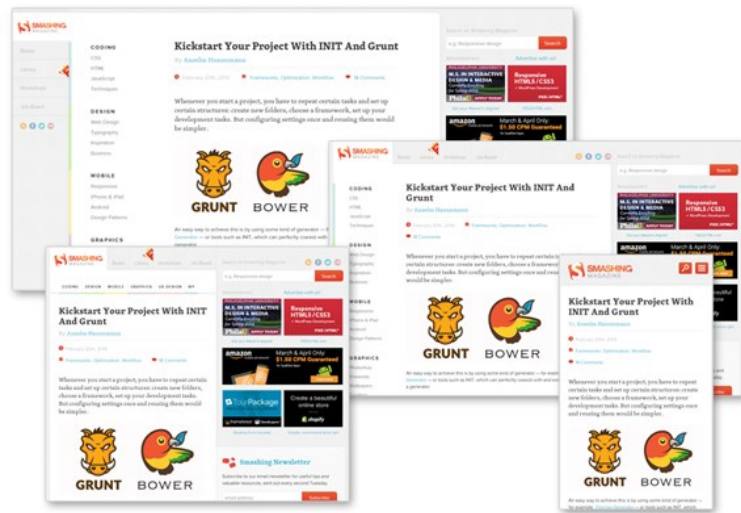
Ali, šta je sledeće? iPad sajt? N90 sajt? Da li smo spremni da nastavimo podršku za svaki novi uređaj i da kreiramo uvek dizajn po meri? U nekom trenutku, ovo počinje da podseća na igru bez dobitnika. Ali kako se mi – i naši dizajn projekti – možemo prilagoditi?

Responsive dizajn

- Svakako je bila potrebna promena paradigme, što se i desilo. Više nismo razmišljali o posebnim sajtovima za računare, mobilne uređaje, televizore i sl, već **o istim sajtovima za sve uređaje**, ali sajtovima koji će se **automatski prilagoditi** i reagovati (engl. *respond – responsive*) na okruženje u kome su prikazani.
- Ukoliko korisnik pređe sa računara na tablet, sajt bi trebao da se prilagodi i da prerasporedi elemente, kao i da promeni potrebne funkcionalnosti. Ovakav pristup odbacuje potrebu za kreiranjem više posebnih sajtova za svaki novi uređaj koji se pojavi.

Responsive dizajn

- Po nekim procenama, danas polovina korisnika pristupa webu preko mobilnog telefona, tableta ili sličnog uređaja. Taj udeo uopšte nije mali.
- Zbog toga moramo prilagoditi naše sajtove da pruže adekvatno iskustvo na svim uređajima.



Responsive dizajn

- Responsive dizajn, iako je promenio način kreiranja i dizajniranja sajtova, **nije revolucija, već samo evolutivni korak**. Ono što je omogućilo sve to, je svakako napredak i razvoj CSS3 i HTML5 jezika.



Responsive dizajn

- Osnova (mada ne i kompletan aspekt) su **Media upiti (engl. *Media queries*)** koji uz ostale CSS mogućnosti omogućavaju promenu rezolucije i rasporeda prikaza u trenutku.
- Ali responsive dizajn nije samo promena rezolucije i automatsko skaliranje slika, već mnogo toga novog u sferi web dizajna (o čemu ćemo uskoro pričati.)

Media tipovi

Media tipovi nisu novina u CSS verziji 3. Uvedeni su još u verziji 2.1, ali su imali drugačiju namenu. Do pojave CSS3 koristili smo ih za eventualno odvajanje CSS-a koji treba biti primenjen prilikom štampe. Na primer:

```
<link href="base.css" rel="stylesheet"  
type="text/css" media="screen" />
```

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

Osim *print* i *screen* varijanti, bile su dostupne još neke vrednosti, ali se praktično nisu često upotrebljavale. Za kompletan spisak pogledajte sledeći link (starija specifikacija): <http://www.w3.org/TR/CSS21/media.html#media-types>

Media upiti

Zbog toga, media atributi su bili retko kad u upotrebi do pojave **media upita (engl. *media queries*)** u CSS verziji 3. Media query nam dozvoljava ne samo da targetiramo određeni tip uređaja ili namenu (kako je bilo ranije), već i da proveravamo karakteristike trenutnog prikaza u browseru i karakteristike samih uređaja.

Na primer, možemo postaviti:

```
<link href="test.css" rel="stylesheet"  
type="text/css" media="screen and (max-width:  
480px) and (orientation: portrait)" />
```

Media upiti

Na ovaj način se mogu pozvati različiti CSS fajlovi, ali ono što se mnogo češće koristi jesu media upiti direktno **unutar jednog CSS fajla**. U tom slučaju ne moramo pozivati više fajlova, već unutar samog (jednog) CSS eksternog fajla odvajamo celine i definisati media upite.

Pogledajmo jedan primer, pa ćemo se kasnije vratiti da pojasnimo samu strukturu i pisanje koda. **Ovaj primer samo prikazuje funkcionalnosti media upita. Ne znači da uvek koristimo baš ove media upite i dimenzije. Oni zavise od samog projekta na kome radimo.**

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Media queries</title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>

<body>
<div id="wrapper">
  <p>.....</p>
</div>
</body>
</html>
```

```
body {  
  margin:0;  
  padding:0;  
}
```

```
#wrapper {  
  margin:0 auto;  
  padding:10px;  
  background:green;  
  width:1170px;  
}
```

```
@media screen and (min-width:1024px) and (max-width:1260px)  
{  
  #wrapper {  
    width:960px;  
  }  
}
```

```
@media screen and (min-width:768px) and (max-width:1023px)  
{  
  #wrapper {  
    width:700px;  
  }  
}
```

```
@media screen and (max-width:767px) {  
  #wrapper {  
    width:90%;  
  }  
}
```


Sada se po prvi put srećemo sa media upitima, i obratite pažnju na zagrade u zagradaama, odnosno:

Svi opisi koji su obuhvaćeni zagradaama koje pripadaju media upitu biće primenjeni samo ukoliko su uslovi ispunjeni.

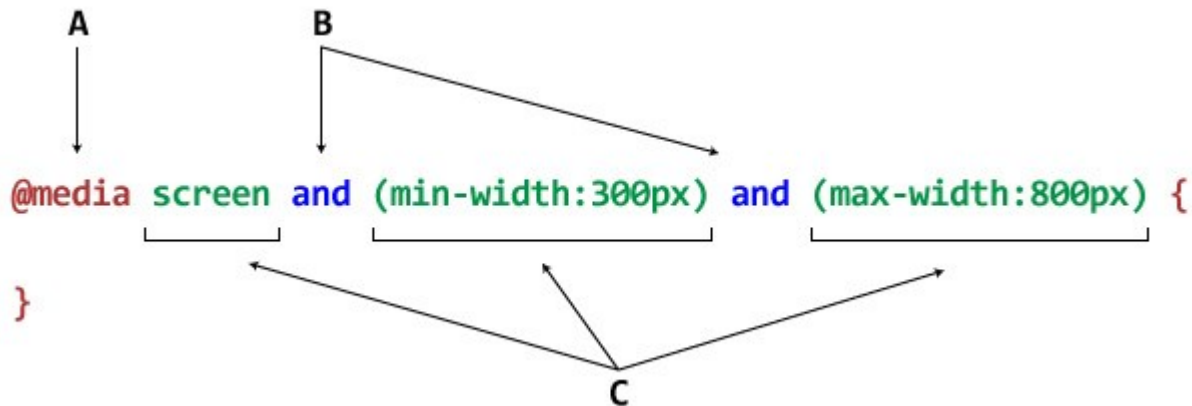
@media uslov {

```
selektor {  
    svojstvo:vrednost;  
    svojstvo:vrednost;  
}
```

```
selektor {  
    svojstvo:vrednost;  
    svojstvo:vrednost;
```

```
}  
ITAcademy
```

Hajde sada da razjasnimo od čega se sastoji media upit unutar CSS fajla.

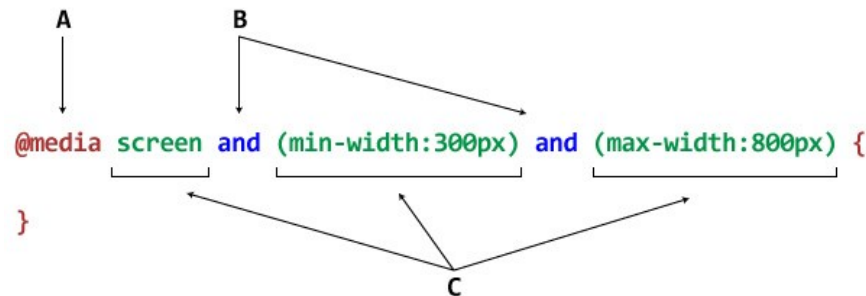


A - media upit oznaka. Uvek počinjemo sa **@media**

B - logički operatori

C - media uslovi (engl. media features)

Media upiti



- Kada postavljamo media upit, govorimo browseru da CSS opise koji slede unutar te grupe (između vitičastih zagrada) primenjuje ukoliko su ispunjeni određeni uslovi.
- Ti uslovi su na skici obeleženi sa C – media features. Možemo više uslova postavljati i vezivati logičkim operatorima (na slici B).

Media uslovi (*media features*)

Media uslovi:	Min/max prefiksi:	Na šta se odnosi:
Width	Da	Trenutna širina prikaza
Height	Da	Trenutna visina prikaza
device-width	Da	Širina ekrana uređaja
device-height	Da	Visina ekrana uređaja
orientation	Ne	Trenutna orijentacija ekrana – landscape (horizontalno) ili portrait (vertikalno)
aspect-ratio	Da	Trenutni odnos širine-visine
device aspect-ratio	Da	Odnos širine-visine uređaja
Color	Da	Broj dostupnih bitova za svaku komponentu boje
color-index	Da	Broj dostupnih boja na uređaju
monochrome	Da	Broj dostupnih bitova na monohromatskim ekranima
Resolution	Da	Rezolucija (gustina piksela) uređaja
Scan	Ne	Progresivno ili prepleteno skeniranje
Grid	Ne	Provera da li je prikaz na grid tipu ekrana (terminali ili telefoni sa jednim fontom i sl.)

Media uslovi (*media features*)

- Većina media uslova prihvata *min- i max-* prefikse kako bi mogli da izrazimo uslov u obliku „*jednako ili veće od*“, odnosno „*jednako ili manje od*“.
- Ukoliko bi koristili operatore iz matematike (konkretno $<$ i $>$) to bi izazvalo konflikt sa HTML sintaksom. Ukoliko nema ovih prefiksa, upit se odnosi na tačnu vrednost, odnosno na jeste/nije opcije u zavisnosti od tipa uslova.

Width

Verovatno najčešći media uslov koji se koristi u upitima. Pomoću width proveravamo trenutnu širinu prikaza. Kao vrednost unosimo brojčanu vrednost, i to najčešće u pikselima (px).

Možemo pisati npr:

```
@media screen and (width: 800px) { ... }
```

Ali ovakav media upit, realno nema svrhu, jer bi funkcionisao samo u jednoj jedinoj širini određenoj sa 800 piksela širine. Dovoljno je da se promeni prikaz za 1px i uslov više ne važi.

Zbog toga se koristi width sa prefiksima. Na primer:

```
@media screen and (max-width: 800px) { ... }
```

Width

Možemo umesto max pisati min vrednost i postaviti jednako ili veće. A možemo kombinovati min i max i dobiti nešto ovako:

@media screen and (min-width: 400px) and (max-width: 800px) { ... }

U ovom sada upitu, primenjuje se CSS samo ako je prikaz između 400 i 800 piksela, uključujući i granične vrednosti.

Height

Height je visina prikaza i sva pravila koja važe za *width*, primenjuju se i ovde. Zbog same prirode sajtova, gde se visina uglavnom dinamički određuje u zavisnosti od dužine sadržaja, ovaj uslov se retko kad koristi.

device-width i device-height

- Za razliku od *width* i *height* uslova koji određuju trenutnu širinu, odnosno visinu prikaza, *device-width* i *device-height* se ne menjaju u zavisnosti od konteksta, i uvek su isti za određeni uređaj, tačnije njegov ekran.
- I kod ovih uslova moguće je primenjivati min i max prefikse i pomoću njih postavljati od-do varijante.
- Ono što malo komplikuje upotrebu je gustina piksela na modernim ekranima, odnosno, drugim rečima, realne dimenzije ekrana i CSS dimenzije ekrana.

Orientation

Određuje da li je uređaj u uspravnoj, vertikalnoj orijentaciji (širina ekrana je manja od visine) ili je u horizontalnoj (širina ekrana je veća od visine). Uglavnom se koristi ukoliko želimo da izdvojimo stilizaciju za određene mobile telefone. Možemo pisati na primer:

@media screen and (orientation: portrait) and (max-width:768px) { ... }

aspect-ratio i device-aspect-ratio

Ova dva uslova postavljamo kad želimo da proverimo odnos stranica trenutnog prikaza (veličina browsera na računaru), odnosno odnos stranica samog ekrana. I ovde su dostupni min i max prefiksi.

Odnos između *aspect-ratio* i *device-aspect-ratio* je kao kod *width* i *device-width*.

U oba slučaja postavljamo kao vrednost dva cela broja odvojena kosom crtom. Na primer:

**@media screen and (device-aspect-ratio: 16/9),
screen and (device-aspect-ratio: 16/10) { ... }**

aspect-ratio i device-aspect-ratio

Sa druge strane mogli smo pisati:

@media screen and (min-aspect-ratio: 1/1) { ... }

Primetite da je sad upotrebljen aspect-ratio (a ne device-aspect-ratio) i to sa prefiksom min. U ovom slučaju, targetiramo samo kvadratne i landscape prikaze, jer mora biti ispunjen uslov da je odnos stranica prikaza 1:1 ili veći od toga (zbog min-).

Color

Uslov koji određuje broj bitova za komponente boja. Možemo koristiti da proverimo da li je u pitanju color ekran. Ovaj uslov danas i nema neku realnu upotrebu. Mogući su prefiksi.

Na primer:

@media screen and (color) { ... }

Ovim upitom ćemo taretirati sve ekrane u boji, što je danas praktično svaki uređaj. Takođe, možemo pisati nešto ovako:

@media screen and (min-color: 4) { ... }

U ovom slučaju targetiramo sve ekrane koji imaju minimum 4 bita boje po kanalu.

color-index

Proverava da li je u pitanju prikaz na ekranu sa indeksiranim (ograničenim bojama). Na primer:

@media screen and (color-index) { ... }

I ovde su dostupni prefiksi, jer možemo pisati i broj boja kao vrednost.

Monochrome

Slično kao ranije pomenuti color samo što funkcioniše sa monohromatskim, crno belim ekranima. Na primer:

@media screen and (monochrome) { ... }

Targetira samo crno bele ekrane. Ili možemo targetirati crno bele ekrane sa minimum 8 bita po pikselu:

@media screen and (min-monochrome: 8) { ... }

Resolution

Određuje rezoluciju ekrana, odnosno gustinu piksela. Unosimo vrednosti u tačkama po inču (dots per inch – dpi) ili u tačkama po centimetru (dots per centimeter – dpcm). Takođe, i ovde možemo koristiti prefikse min i max.

Na primer, želimo da targetiramo novije ekrane sa gustinom piksela većom od 240dpi. Možemo tad pisati:

@media screen and (min-resolution: 240dpi) { ... }

Scan

Koristimo kod prikaza na tv ekranima. Možemo izdvojiti interlaced ili progressive skeniranje. Ovde ne možemo koristiti prefikse.

Na primer:

@media tv and (scan: progressive) { ... }

Ovako bi targetirali samo tv ekrane sa progresivnim skeniranjem.

Logički operatori media upita

- U prethodnim slajdovima smo se upoznali sa media uslovima, a sada ćemo detaljnije obraditi logičke operatore pomoću kojih možemo da kombinujemo više uslova ukoliko je to potrebno i tako kreirati složene upite.

and

Sabiraju se uslovi (media features) i moraju svi biti tačni kako bi upit bio primenjen. Na primer:

@media (max-width: 480px) and (orientation: portrait)
{ ... }

U ovom primeru bi se media upit (media query) primenjivao samo ako je širina prikaza ispod **481** piksela i ukoliko je uređaj u vertikalnom prikazu. Drugim rečima, svi povezani uslovi (media features) moraju biti ispunjeni. Ovaj logički operater se uglavnom najčešće koristi.

not

Proverava negaciju nad upitom, odnosno media upit će biti primenjen ukoliko zadati uslov **nije** ispunjen. Pritom, not se odnosi na sve vezane media upite, ne samo na prvi. Pogledajmo primere:

@media not (monochrome) { ... }

Ovakav primer kreira media upit i primenjuje zadate CSS opise u njemu na svim uređajima koji nemaju monohromatski (crno-beli) ekran.

not

Ali, ukoliko postavimo sledeće:

@media not all and (monochrome) { ... }

Praktično bi rezultat bio isti. Jer ovakav media upit proverava: Da li nije u pitanju bilo koji monohromatski uređaj.

Ovaj logički operator (kao i naredni) se uglavnom ne koristi.

only

Ovaj logički operater je uveden kod media upita samo zbog starijih browsera. Zamisao je da se spreče stariji browseri da ne učitavaju CSS opise u media upitu.

```
@media only screen and (max-width: 480px)  
{ ... }
```

Zarez (engl. comma)

Ukoliko želimo da postavimo dva media upita koji će imati iste CSS opise, možemo ih odvojiti zarezom. Princip je sličan kao kod postavljanja selektora: dva različita selektora možemo napisati u jednom opisu, samo ćemo ih odvojiti zarezom.

Tako je i kod media upita. Na primer:

**@media (min-width: 600px) and (orientation: portrait),
(min-width: 800px) and (orientation: landscape) { ... }**

Zarez (engl. comma)

Ovakav media upit, odnosno CSS opisi u ovakvom media upitu bi bili primenjeni kada je

- širina prikaza veća od 600 piksela i pri tom je uređaj u vertikalnom načinu rada

ili

- (potpuno nevezano za prethodno) kada je širina prikaza veća od 800 piksela i pri tom je uređaj u horizontalnom načinu rada. Odnosno, kao da smo pisali:

@media (min-width: 600px) and (orientation: portrait) { ... }

@media (min-width: 800px) and (orientation: landscape) { ... }

Sve ove logičke operatore i media features, pomenute iznad, možemo kombinovati i dobiti (uglavnom nepotrebno) komplikovanije varijante.

Najčešća postavka media upita je kod širina. I uglavnom srećemo ovako nešto:

```
/* ovde idu opisi za svaki prikaz */
```

```
@media screen and (min-width:1024px) and (max-width:1260px) {
```

```
/* ovde idu opisi za prikaz između 1024 i 1260 piksela */
```

```
}
```

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

```
/* ovde idu opisi za prikaz između 768 i 1023 piksela */
```

```
}
```

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

```
/* ovde idu opisi za prikaz manji od 768 piksela */
```

```
}
```