

Media queries

U prethodnoj lekciji zaustavili smo se prilikom realizacije grid sistema, odmah nakon uvođenja relativnih jedinica za definisanje širine redova i kolona. Za dodatnu optimizaciju grid sistema, u priču o responsive designu neophodno je uvesti još neke pojmove. Prvi takav pojam, kojim će se baviti lekcija pred vama, biće medija upiti (*media queries*).

Šta su medija upiti?

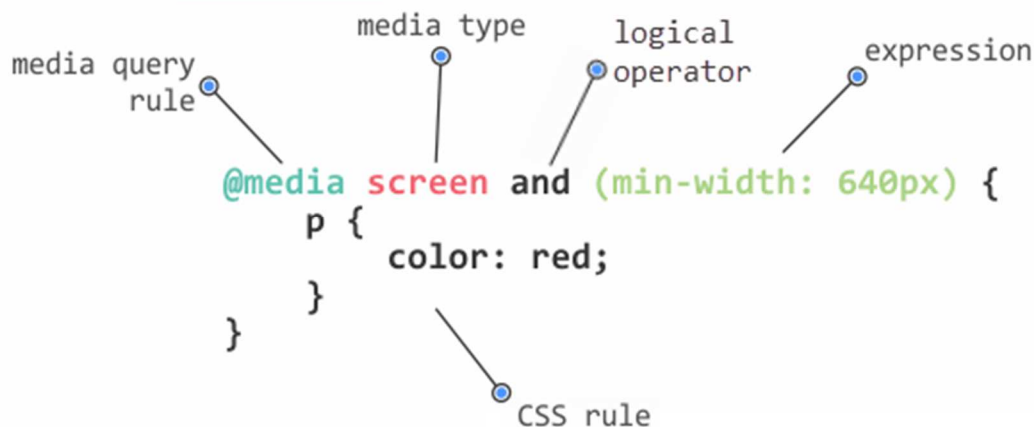
Medija upiti (*media queries*) posebna su funkcionalnost CSS jezika, koja omogućava da se određeni CSS opisi primene samo u slučaju zadovoljenja nekog uslova.

Primer jednog medija upita je sledeći:

```
@media screen and (min-width: 640px) {  
  p {  
    color: red;  
  }  
}
```

Ovo je medija upit koji unutar sebe poseduje jedan CSS opis. CSS opisom definiše se boja teksta paragraf elemenata. Korišćenjem ovakvog medija upita rečeno je da će svi paragraf elementi imati crvenu boju teksta, ali samo kada je širina HTML dokumenta veća od 640 piksela ili jednaka toj vrednosti i kada se HTML dokument prikazuje na displejima uređaja.

Kako biste sve ovo lakše razumeli, različiti elementi upravo prikazanog medija upita predstavljeni su slikom 13.1.



Slika 13.1. Različiti segmenti medija upita

Sa slike 13.1. se može videti da medija upit započinje `@media` at-rule izjavom, koja je na slici obojena plavom bojom. Nakon ove ključne reči, medija upiti mogu sadržati različite delove:

- tipove medija;
- izraze za testiranje osobina uređaja;
- logičke operatore.

Tip medija (media type)

Reč `screen` unutar prikazanog medija upita određuje tip medija na koje će se upit odnositi. Medij `screen` nije jedini tip koji se može naći unutar medija upita. Kompletan spisak medija koji se mogu iskoristiti prilikom formiranja medija upita prikazan je u tabeli 13.1.

Tip medija	Opis
<code>all</code>	stilovi medija upita primenjuju se u svim situacijama
<code>print</code>	stilovi medija upita primenjuju se samo prilikom štampe
<code>speech</code>	stilovi medija upita koji se primenjuju samo prilikom glasovnog čitanja koje izvršavaju takozvani <u>sintetizatori govora</u>
<code>screen</code>	stilovi medija upita primenjuju se samo prilikom prikaza web stranice na ekranu; to znači da se stilovi medija upita neće primenjivati u štampi ili prilikom korišćenja sintetizatora govora; ovo je podrazumevana vrednost

Tabela 13.1. Tip medija unutar medija upita

Sve ovo praktično znači da će se medija upit sa slike 13.1. primenjivati samo kada se HTML dokument prikazuje na displeju korisničkih uređaja. Ovo je inače i najčešća situacija, te se zbog toga i vrednost `screen` uglavnom izostavlja, s obzirom na to da je podrazumevana.

Pored vrednosti `screen`, veoma korisna može biti i vrednost `print`. Korišćenjem medija upita sa takvim tipom medija moguće je definisati potpuno proizvoljnu stilizaciju koja bi se primenjivala samo kada se iz web pregledača pokrene štampa web stranice. Takva funkcionalnost može biti i više nego korisna prilikom razvoja poslovnih web aplikacija.

Tip medija nije obavezan deo medija upita, tako da se može izostaviti.

Pitanje

Ukoliko je potrebno naglasiti da se određeni medija upit koristi samo prilikom štampe, dovoljno je navesti tip:

- `speech`
- `screen`
- **`print`**
- `paper`

Objašnjenje:

Tip medija upita `print` definiše primenu medija upita samo prilikom štampe.

Izrazi (expressions)

Pored upravo opisanog tipa medija, definisanje uslova za primenu CSS opisa mnogo preciznije je moguće obaviti korišćenjem izraza. Svaki medija upit može imati jedan ili više izraza, koji definišu u kojim situacijama će se stilizacija unutar medija upita aktivirati.

U prikazanom primeru, izraz medija upita je sledeći:

```
(min-width: 640px)
```

Unutar izraza medija upita vrši se proveravanje neke od funkcionalnosti uređaja na kome se HTML stranica prikazuje. U konkretnom primeru, proverava se da li je vidno polje (*viewport*) uređaja na kome se HTML dokument čita jednako ili veće od 640px. Tako se jedan izraz unutar medija upita sastoji iz testirane funkcionalnosti i vrednosti, kao na slici 13.2.



Slika 13.2. Različiti segmenti izraza medija upita

CSS omogućava testiranje različitih funkcionalnosti uređaja koji parsiraju HTML dokumente. Najznačajnije osobine uređaja koje se mogu testirati prikazane su tabelom 13.2.

Osobina	Opis
height	visina vidnog polja
width	širina vidnog polja
orientation	orijentacija vidnog polja, odnosno da li je uređaj u <u>portret</u> ili <u>landscape</u> modu
resolution	rezolucija izlaznog uređaja, odnosno gustina tačaka, koja se izražava korišćenjem jedinica <code>dpi</code> ili <code>dpcm</code>
aspect-ratio	odnos širine i visine vidnog polja
color	broj bita po jednoj boji izlaznog uređaja za prikaz slike

Tabela 13.2. Različite osobine uređaja koje se mogu testirati unutar medija upita

Jedan medija upit može imati proizvoljan broj izraza za testiranje osobina uređaja.

Logički operatori

Različiti uslovi unutar medija upita međusobno se razdvajaju specijalnim ključnim rečima koje se nazivaju logički operatori. Takva je situacija i unutar medija upita prikazanog slikom 13.1. Unutar njega, ključna reč `and` predstavlja primer logičkog operatora.

CSS poznaje nekoliko logičkih operatora koje je moguće koristiti prilikom formiranja složenih medija upita (tabela 13.3).

Logički operator	Opis
and	grupa više izraza i pri tome zahteva da svaki od njih bude ispunjen kako bi se primenila stilizacija definisana medija upitom
not	operator koji se koristi za postizanje negacije; kada se postavi ispred nekog izraza medija upita, vrši invertovanje uslova; na primer, ukoliko izraz definiše maksimalnu širinu od 1024 piksela, negacijom će takav uslov biti zadovoljen za sve širine veće od 1024px
only	logički operator kojim se definiše obaveza ispunjenja svih uslova unutar složenih medija upita; iako je tako nešto potpuno očekivano i bez korišćenja ovog logičkog operatora, ovaj operator je namenjen starijim web browserima; oni imaju tendenciju da ignorišu izraze koji dolaze nakon prvog izraza, tako da se korišćenjem ovog logičkog operatora njima stavlja do znanja da je neophodno da se zadovolje svi uslovi složenih medija upita
, (comma)	karakter zapeta se može koristiti za kombinovanje većeg broja medija upita; karakter zapeta se ponaša kao logički operator <i>ili</i> , zato što zadovoljenje samo jednog medija upita iz niza navedenih znači primenu definisane stilizacije

Tabela 13.3. CSS logički operatori koji se mogu koristiti u medija upitima

Već je rečeno da se prilikom formiranja medija upita tip medija (`screen`, `print...`) može izostaviti, dok je broj izraza proizvoljan. Jedan takav primer može da izgleda ovako:

```
@media (max-width: 1024px) and (orientation: landscape) {  
    ...  
}
```

Prikazani primer ilustruje složeni medija upit koji se sastoji iz dva izraza kojima se testiraju dve različite funkcionalnosti uređaja. Prvim izrazom rečeno je da će se stilizacija u vitičastim zgradama primenjivati samo na uređaje maksimalne širine vidnog polja 1024px. Drugim izrazom definisano je da će se stilizacija primenjivati samo kada je uređaj u položenom (*landscape*) modu. Za razdvajanje dva upravo opisana izraza iskorišćen je logički operator `and`, kojim je definisano da će uslovi oba definisana izraza morati da se zadovolje kako bi se primenila stilizacija medija upita.

Korišćenje medija upita za kreiranje fleksibilnog grida

U prethodnoj lekciji smo se zaustavili prilikom kreiranja jednog grid sistema za raspoređivanje HTML elemenata po redovima i kolonama. Rečeno je da bi sledeći korak podrazumevao kreiranje ponašanja po kome bi naš grid imao različit broj kolona na različitim uređajima. Tako nešto nismo mogli uraditi bez medija upita, tako da je sada došao trenutak da ono što je teoretski opisano u dosadašnjem toku ove lekcije iskoristimo za kreiranje fleksibilnog grida.

Medija upiti koji će naš grid sa šest kolona učiniti fleksibilnim mogu da izgledaju ovako:

Radno okruženje

HTML fajl:

```
<div id="row">
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
  <div class="column">
    <div class="box"></div>
  </div>
</div>
```

CSS fajl:

```
.row {
  width: 90%;
  max-width: 960px;
  margin: 0 auto;
}

.row::after {
  content: "";
  display: block;
  clear: left;
}

.column {
  float: left;
  width: 16.66%;
  box-sizing: border-box;
  padding: 8px;
}

.box {
  height: 144px;
  background-color: #914E67;
}

@media screen and (max-width: 1024px) {
  .column {
    width: 33.33%;
  }
}

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

```

        .column {
            width: 50%;
        }
    }

    @media screen and (max-width: 414px) {
        .column {
            width: 100%;
        }
    }

```

Upravo su prikazana tri različita medija upita pomoću kojih naš grid postaje fleksibilan. Prvim medija upitom definisano je šta će se dogoditi kada je širina vidnog polja browsera jednaka ili manja od 1024px. U takvoj situaciji, širina kolona je postavljena na 33.33%, što je zapravo jedna trećina kompletnog reda. Na ovaj način će ispod 1025px naš grid imati tri kolone umesto šest kolona.

Sledećim medija upitom definisano je šta će se dogoditi kada vidno polje postane manje od 769px. Tada kolone dobijaju širinu od 50%, pa to praktično znači da će ih biti dve.

Na kraju, kada je širina vidnog polja manja od 415px, grid će imati samo jednu kolonu, zato što je širina kolona postavljena na 100%.

Pokušajte da unutar radnog okruženja izmenite media upite unutar CSS koda kako bi omogućili prikaz 4 kolona u jednom redu ukoliko je širina vidnog polja browsera jednaka ili manja od 1024px.

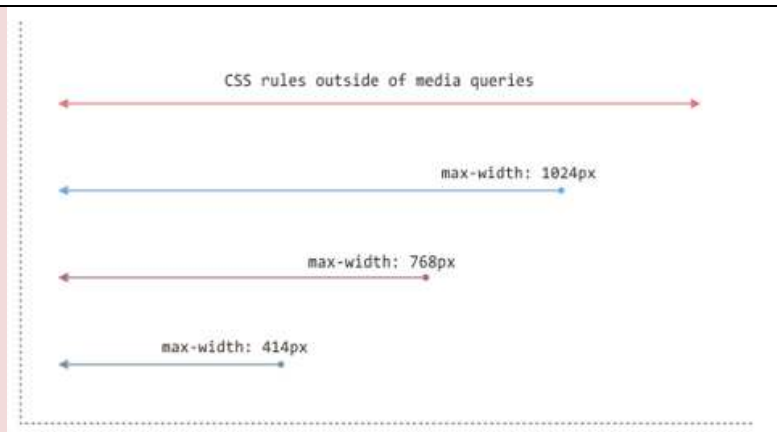
Desktop first i mobile first

U svetu web razvoja veoma često se mogu čuti pojmovi *mobile first* i *desktop first*. Reč je o dve tehnike za postizanje responsive dizajna. Na primer, prilikom realizacije upravo prikazanog primera korišćen je desktop first pristup. To znači da je prvo kreiran raspored komponentata za desktop uređaje, a da je tek nakon toga obavljeno njegovo prilagođavanje za uređaje sa displejima manjih dimenzija (tableti, telefoni...).

Mobile first pristup podrazumeva da se layout web sajta prvo kreira za pametne telefone, a zatim se takav raspored komponentata prilagođava uređajima sa većim displejima.

S obzirom na to da je broj mobilnih uređaja već odavno nadmašio broj konvencionalnih, desktop uređaja na webu, korišćenje mobile first pristupa ima puno smisla. Ipak, to ne znači da se korišćenjem desktop first pristupa ne može kreirati dobro optimizovan web sajt, čije korisničko okruženje bi bilo prilagođeno svim uređajima.

Korišćenje desktop first i mobile first pristupa diktira i osobine medija upita. Na primer, kada se koristi desktop first pristup, mnogo je zgodnije koristiti medija upite unutar kojih se ispituje `max-width` osobina, baš kao što je to učinjeno u prikazanom primeru. Na taj način je omogućeno stepenasto definisanje osobina koje će se aktivirati na uređajima sa različitim displejima. Takođe, izbegava se ponavljanje CSS opisa unutar više medija upita (slika 13.3).

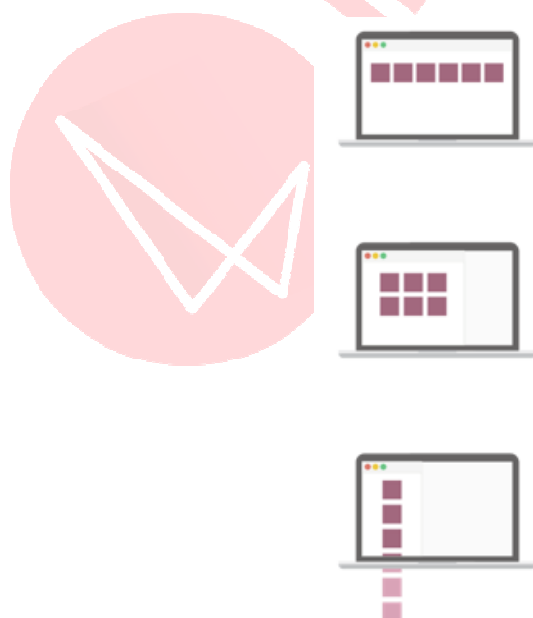


Slika 13.3. Efekat medija upita koji ispituju max-width osobinu

Sa slike 13.3. se može videti da se CSS opisi definisani izvan medija upita primenjuju u svim situacijama. Zatim, mediji upit sa `max-width: 1024px` uslovom primenjuje se za sve širine ispod 1025px. Ovakva osobina omogućava da se unutar konkretnih medija upita definišu samo one osobine koje su karakteristične za uslov koji se ispituje. Sve ostale osobine preuzimaju se iz CSS opisa definisanih iznad.

Na osnovama upravo ilustrovanog pristupa funkcionišu medija upiti i kada se koristi mobile first pristup. Jedina razlika je u upotrebi osobine `min-width` umesto `max-width`.

Funkcionisanje upravo kreiranih medija upita može se ispitati smanjivanjem i povećavanjem širine web pregledača na nekom desktop uređaju. U takvoj situaciji, biće dobijen prikaz kao na slici 13.4.



Slika 13.4. Efekat medija upita pri kreiranju fleksibilnog grida

Korišćenjem web pregledača na desktop ili laptop uređajima, može se obaviti testiranje fleksibilnosti našeg grida. Pri tome se prikazi na tabletu i telefonu simuliraju smanjivanjem širine browsera. Ipak, ukoliko bismo HTML dokument sa ovakvim gridom otvorili pomoću web pregledača na nekom telefonu ili tabletu, dočekalo bi nas iznenađenje – prikaz na takvim uređajima ne bi bio adekvatan. Kako bismo mogli da razumemo zašto je to tako, u priču o responsive dizajnu neophodno je uvesti i pojam vidnog polja (*viewport*), kome je posvećena naredna lekcija.

Rezime

- Medija upit – media query.
- Medija upiti dozvoljavaju da se određeni CSS opisi primene samo u slučaju zadovoljenja nekog uslova.
- Medija upit započinje `@media` at-rule izjavom.
- Prilikom formulisanja medija upita može se navesti tip medija na koje se upit odnosi – `all`, `print`, `screen`, `speech`.
- Tip medija nije obavezan deo medija upita, tako da se može izostaviti.
- Svaki medija upit može imati jedan ili više izraza koji definišu u kojim situacijama će se stilizacija unutar medija upita aktivirati.
- Jedan izraz unutar medija upita sastoji iz testirane funkcionalnosti i vrednosti.
- Različiti uslovi unutar medija upita međusobno se razdvajaju specijalnim ključnim rečima koje se nazivaju logički operatori.
- Složeni medija upit je onaj koji poseduje veći broj pojedinačnih uslova.
- Medija upiti se mogu koristiti kao efikasno sredstvo za kreiranje fleksibilnih medija upita.
- Desktop first pristup podrazumeva kreiranje rasporeda komponenata za desktop uređaje, nakon čega se takav raspored prilagođava za uređaje sa displejima manjih dimenzija.
- Mobile first pristup podrazumeva da se layout web sajta prvo kreira za pametne telefone, a zatim se takav raspored komponenata prilagođava uređajima sa većim displejima.

