

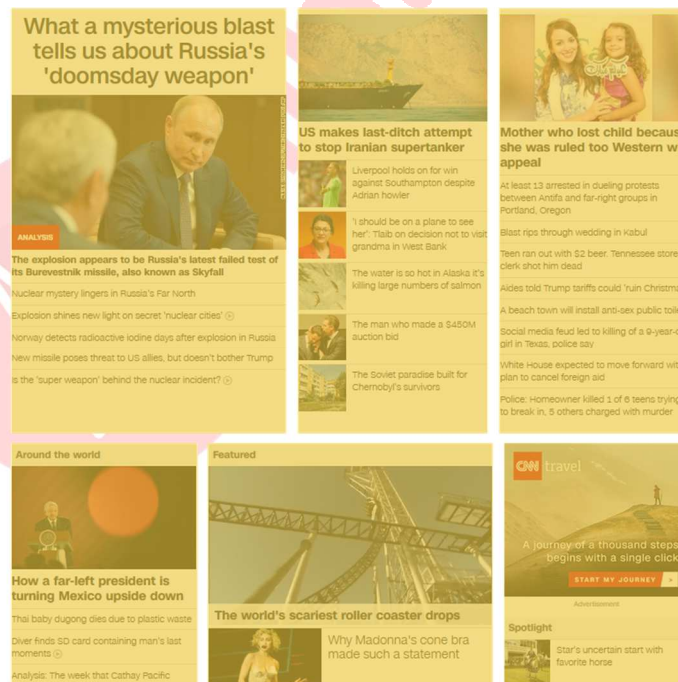
# Bootstrap grid

Dobar dizajn, bez obzira na to da li je reč o dizajnu za web ili print, podrazumeva vešt balans tipografije, boja i kompozicije. Prilikom kreiranja kompozicije, dizajnerima je na raspolaganju jedna veoma moćna tehnika – *grid*. Tako je *grid* pojam koji nije striktno vezan za web, ali je svoju primenu našao i prilikom dizajniranja i kreiranja korisničkih okruženja web sajtova.

Za većinu frontend programera, prva asocijacija na Bootstrap jeste upravo *grid* sistem koji ovaj softverski okvir poseduje. I zaista, Bootstrap je svoju slavu stekao upravo zahvaljujući odličnom grid sistemu za raspoređivanje elemenata. Lekcija pred vama pokušaće da vam približi najznačajnije osobine takvog sistema.

## Pojam grida u web dizajnu

Grid je zapravo osnovna struktura, odnosno kostur dizajna. Sastoji se iz nevidljivih linija, čijim presecanjem se gradi struktura, odnosno sistem koji se može koristiti za raspoređivanje elemenata. U slučaju web dizajna, takvi elementi su grafičke komponente i sadržaj različitog tipa od kojih je sastavljen jedan web sajt. Grid koji se koristi u web dizajnu po svojim osobinama je veoma sličan tabelama – sastoji se iz nevidljivih kolona i redova. I zaista, ukoliko analizirate većinu današnjih web sajtova, moći ćete da primetite načela ovakvog raspoređivanja (slika 2.1).



Slika 2.1. Grid na primeru jednog web sajta

Slika 2.1. ilustruje grid na primeru jednog popularnog web sajta. Čelije koje se grade presecima kolona i redova markirane su žutom bojom. Moguće je uočiti postojanje dva reda, sa po tri kolone.

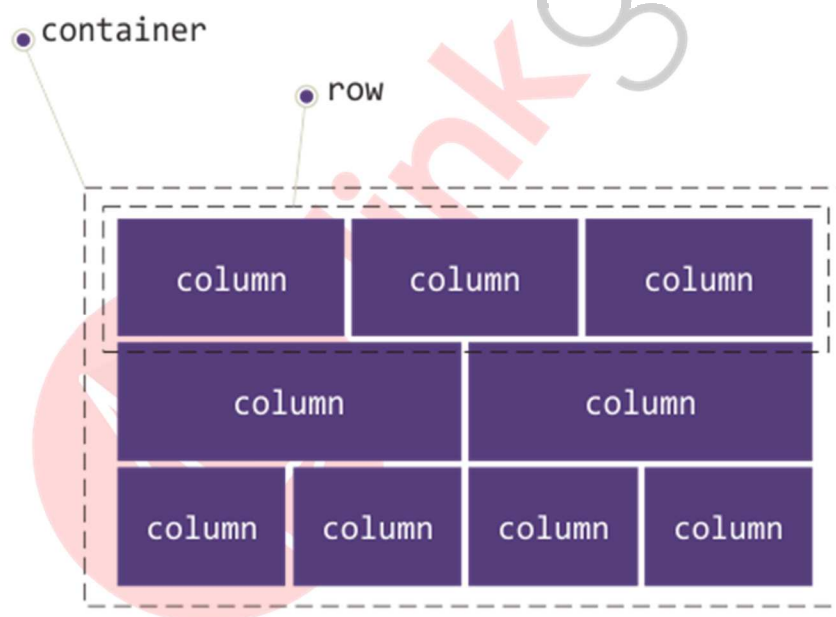
## Kako funkcioniše Bootstrap grid?

Bootstrap poseduje veoma moćan Grid System, koji za raspoređivanje elemenata koristi koncepte kao što su kontejneri, redovi i kolone. Bootstrap je svakako najpoznatiji upravo po svom grid sistemu, čije su presudne osobine jednostavnost korišćenja i velika moć konfiguracije. Takođe, osnovni postulati Bootstrap grid sistema veoma se lako i brzo uče.

Bootstrap grid može se sastojati iz tri grupe elemenata:

- kontejnera;
- redova i
- kolona.

Kontejneri su glavni okružujući elementi, unutar kojih se direktno smeštaju redovi. Redovi služe kao kontejneri za kolone, dok su kolone kontejneri za sadržaj. Tako se sadržaj isključivo smešta unutar kolona, kolone unutar redova, a redovi unutar kontejnera (slika 2.2).



*Slika 2.2. Osnovna struktura Bootstrap grida*

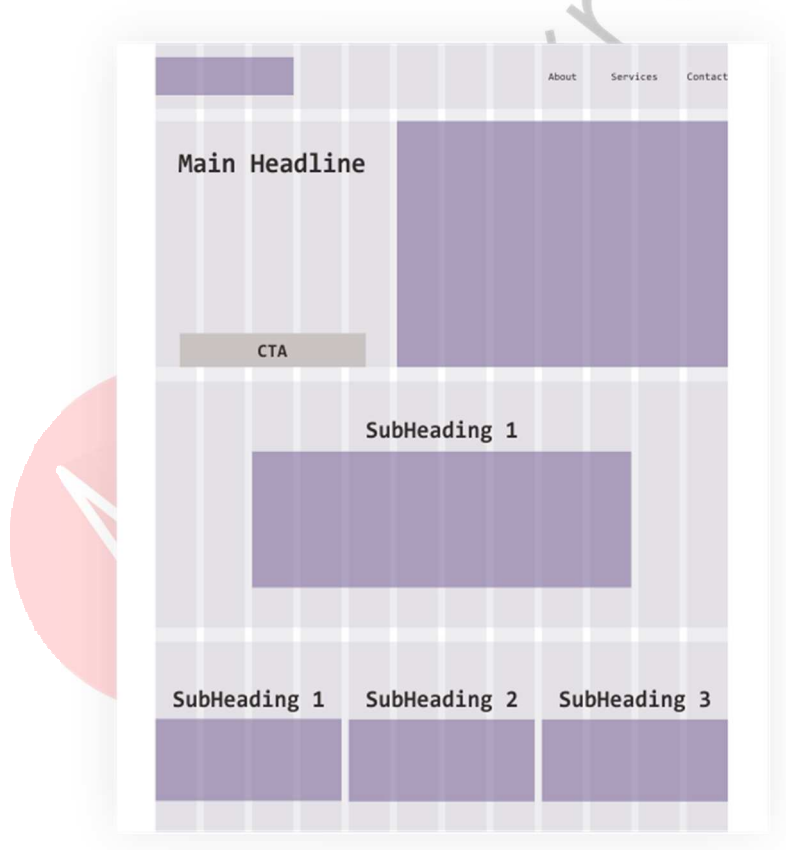
Na slici 2.2. može se videti osnovna struktura Bootstrap grida na primeru jednog kontejnera koji poseduje tri reda. Prvi red ima tri, drugi dve, a treći četiri kolone.

Maksimalan broj kolona Bootstrap grida je 12 (slika 2.3).



*Slika 2.3. Bootstrap grid može da ima maksimalno 12 kolona*

Kreiranjem Bootstrap kontejnera sa redovima i kolonama dobija se Bootstrap grid, pomoću koga je praktično bilo koji layout moguće pretočiti u funkcionalan kod.



*Slika 2.4. Wireframe realizovan korišćenjem grida sa 12 kolona*

Slika 2.4. ilustruje primer wireframea jednog sajta, za čiju realizaciju su korišćene osobine Bootstrap grid sistema.

### Pitanje

Odabrati tačne tvrdnje o osnovnim elementima Bootstrap grida:

- a) sadržaj se smešta direktno unutar kontejnera
- b) redovi se smeštaju unutar kolona
- c) **kolone se smeštaju unutar redova**

### Objašnjenje:

*Kontejneri su glavni okružujući Bootstrap elementi, unutar kojih se direktno smeštaju redovi. Redovi služe kao kontejneri za kolone, dok su kolone kontejneri za sadržaj.*

## Kontejnerski elementi

U prethodnim redovima je rečeno da su osnovni okružujući elementi Bootstrap grida kontejneri. Primarna namena Bootstrap kontejnera je da okruže sav sadržaj Bootstrap grida. Pored takve osnovne namene, kontejneri se mogu koristiti i za dodavanje unutrašnjeg razmaka (paddinga) i centriranje kompletnog sadržaja grida.

Kreiranje Bootstrap kontejnera se može obaviti korišćenjem dve osnovne klase (tabela 2.1).

Klasa	Opis
.container	definiše kontejner fiksne veličine, kojim se postiže kreiranje <u>boxed layouta</u>
.container-fluid	klasa kojom se kreira okružujući element koji zauzima kompletnu dostupnu širinu unutar vidnog polja; na ovaj način se postiže kreiranje takozvanog <u>full-width layouta</u>

Tabela 2.1. Klase za kreiranje Bootstrap kontejnera

Razlike između dve osnovne klase za kreiranje Bootstrap kontejnera ilustrovane su slikom 2.5.



Slika 2.5. Razlike između kontejnera koji se kreiraju klasama .container i .container-fluid

## Definisanje redova i kolona

Nakon definisanja Bootstrap kontejnera, moguće je preći na definisanje redova i kolona. Bootstrap redovi se kreiraju korišćenjem klase **.row**:

```
<div class="container">
  <div class="row">

    </div>
    <div class="row">

    </div>
    <div class="row">

    </div>
  </div>
</div>
```

Prikazani kod ilustruje Bootstrap grid sa tri reda. Kontejner i pojedinačni redovi su kreirani korišćenjem `div` elemenata na koje su postavljene odgovarajuće klase – `.container` za glavni kontejner i `.row` za svaki od redova. Bitno je razumeti da su umesto `div` elemenata mogli biti korišćeni i bilo koji drugi kontejnerski elementi sa specijalnim semantičkim značenjem (`section`, `article`, `heading`...).

Osnovna klasa za kreiranje Bootstrap kolona jeste klasa **.col**:

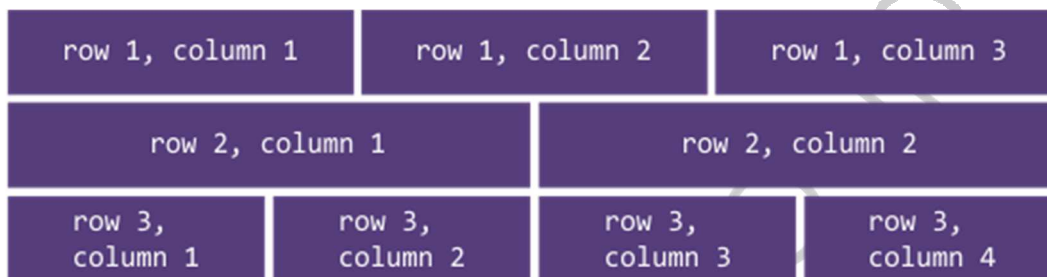
```
<div class="container">
  <div class="row">
    <div class="col">
      row 1, column 1
    </div>
    <div class="col">
      row 1, column 2
    </div>
    <div class="col">
      row 1, column 3
    </div>
  </div>
  <div class="row">
    <div class="col">
      row 2, column 1
    </div>
    <div class="col">
      row 2, column 2
    </div>
  </div>
  <div class="row">
    <div class="col">
      row 3, column 1
    </div>
    <div class="col">
      row 3, column 2
    </div>
    <div class="col">
      row 3, column 3
    </div>
  </div>
</div>
```

```

    </div>
    <div class="col">
        row 3, column 4
    </div>
</div>
</div>

```

Sada su redovima dodate i kolone sa tekstualnim sadržajem, tako da struktura grida odgovara onoj sa slike 2.2 – Bootstrap grid poseduje tri reda, od kojih prvi ima tri, drugi dve, a treći četiri kolone. Na ovaj način se dobija Bootstrap grid sa osobinama ilustrovanim slikom 2.6.



*Slika 2.6. Bootstrap grid kreiran upravo prikazanim kodom – poseduje tri reda, sa po tri, dve i četiri kolone, respektivno*

### Hijerarhija osnovnih Bootstrap elemenata za kreiranje grida

Bitno je razumeti da osnovni elementi za kreiranje Bootstrap grida (kontejneri, redovi i kolone) imaju jasno utvrđenu namenu, pa je tako i njihova hijerarhija utvrđena pravilima Bootstrapa. Kontejneri su glavni okružujući elementi. Tako kreiranje Bootstrap grida uvek treba da započne kreiranjem kontejnerskog elementa. Unutar kontejnera direktno se mogu naći samo redovi. Isto tako, unutar redova, kao direktni potomci mogu postojati samo kolone. Na kraju, sadržaj se mora smeštati isključivo unutar kolona.

Upravo navedeni koraci nisu samo puka pravila, već i potreba kako bi se dobio potpuno funkcionalan grid. Bootstrap se oslanja na upravo opisanu hijerarhiju (kontejner -> red -> kolona), kako bi na pravi način međusobno razdvojio susedne kolone i sadržaj unutar kolona od njihovih ivica.

### Podešavanje širine kolona

U prethodnom primeru, za definisanje kolona iskorišćena je Bootstrap klasa `.col` i to na svim kolonama. Na ovaj način su u svakom redu dobijene kolone identičnih širina, što se može videti na slici 2.6. Drugim rečima, korišćenjem klase `.col`, sve kolone unutar jednog reda podjednako dele dostupan prostor. Ukoliko, na primer, unutar reda postoje dve kolone, one će imati širinu od po 50%. Tri kolone unutar jednog reda imaće širinu od po 33,33%, a četiri kolone širinu od po 25%.

Kontrolu nad širinom Bootstrap kolona je moguće preuzeti korišćenjem specifičnih klasa ovog softverskog okvira. One započinju prefiksom `col-`, nakon koga se navodi jedan od brojeva u rasponu od 1 do 12.

# .col-\*

*Slika 2.7. Struktura Bootstrap klase za definisanje širine kolona; na mestu karaktera zvezdica navodi se jedan od brojeva između 1 i 12*

Pošto je iz prethodnih redova poznato da Bootstrap grid poseduje 12 kolona, nije teško zaključiti da se brojem definiše udeo u kompletnoj širini Bootstrap grida. Korišćenjem takvog pristupa, dobijaju se različite klase za kreiranje Bootstrap kolona, ilustrovane tabelom 2.2.

Klasa	Značenje
.col-1	Širina kolone će biti 1/12 ukupne širine grida
.col-2	Širina kolone će biti 2/12 ukupne širine grida
.col-3	Širina kolone će biti 3/12 ukupne širine grida
.col-4	Širina kolone će biti 4/12 ukupne širine grida
.col-5	Širina kolone će biti 5/12 ukupne širine grida
.col-6	Širina kolone će biti 6/12 ukupne širine grida
.col-7	Širina kolone će biti 7/12 ukupne širine grida
.col-8	Širina kolone će biti 8/12 ukupne širine grida
.col-9	Širina kolone će biti 9/12 ukupne širine grida
.col-10	Širina kolone će biti 10/12 ukupne širine grida
.col-11	Širina kolone će biti 11/12 ukupne širine grida
.col-12	Širina kolone će biti jednaka širini kompletnog grida

*Tabela 2.2. Klase za kreiranje Bootstrap kolona*

Jedan primer samostalnog definisanja širina Bootstrap kolona može da izgleda ovako:

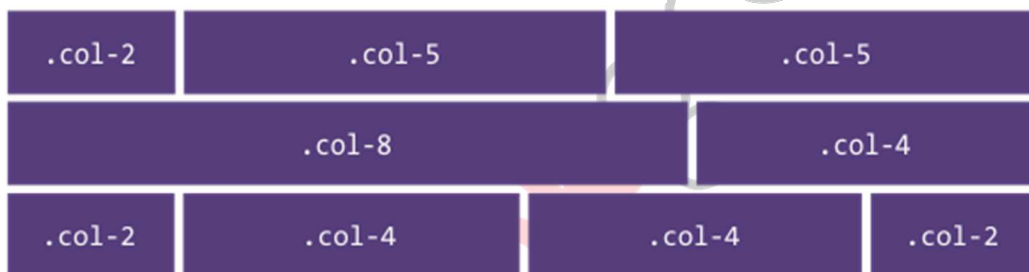
```
<div class="container">
  <div class="row">
    <div class="col-2">
      row 1, column 1
    </div>
    <div class="col-5">
      row 1, column 2
    </div>
    <div class="col-5">
      row 1, column 3
    </div>
  </div>
  <div class="row">
    <div class="col-8">
      row 2, column 1
    </div>
    <div class="col-4">
      row 2, column 2
    </div>
  </div>
</div>
```

```

<div class="row">
  <div class="col-2">
    row 3, column 1
  </div>
  <div class="col-4">
    row 3, column 2
  </div>
  <div class="col-4">
    row 3, column 3
  </div>
  <div class="col-2">
    row 3, column 4
  </div>
</div>
</div>

```

Na ovaj način je stvoren Bootstrap grid sa kolonama čija je širina eksplicitno definisana. Efekat različitih upotrebljenih klasa se ilustrativno može predstaviti kao na slici 2.8.



Slika 2.8. Struktura Bootstrap grida sa kolonama čija je veličina eksplicitno određena, specifičnim klasama

## Responsive grid

Osnovna načela na kojima se Bootstrap grid zasniva jesu postulati *responsive* i *mobile first* dizajna. Zbog toga Bootstrap grid automatski obezbeđuje osobinu prilagođavanja trenutnoj veličini vidnog polja. U narednim redovima biće iznete osnovne osobine Bootstrap *responsive* sistema, kao i pristupi za njihovo kontrolisanje.

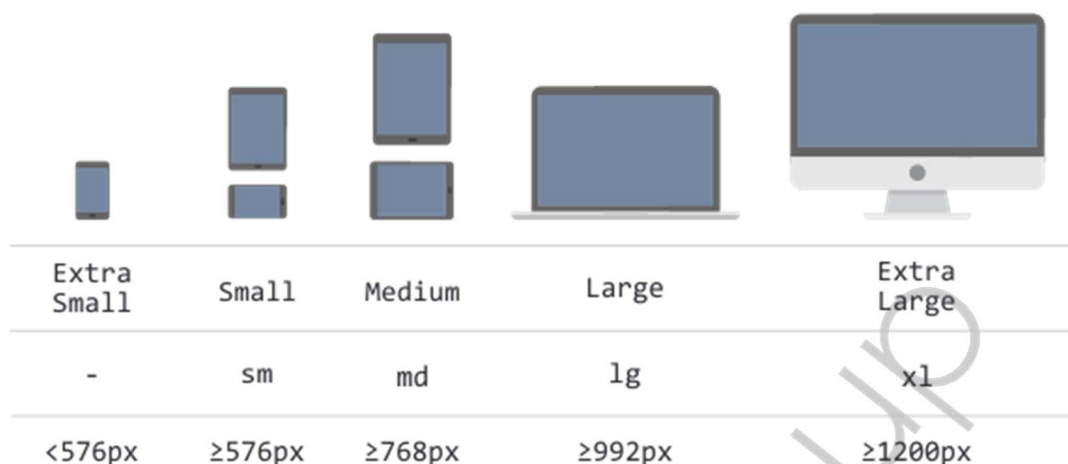
Bootstrap poseduje automatski sistem koji se koristi za detekciju osobina uređaja, pri čemu se prevashodno misli na osobine displeja. Tako su unutar Bootstrapa displeji, pa samim tim i uređaji, raspodeljeni u nekoliko grupa i imenovani specifičnim nazivima (tabela 2.3).

Displej uređaja	Skraćenica	Uslov
Extra Small	-	kada je vidno polje manje od 576px
Small	sm	kada je vidno polje jednako ili veće od 576px
Medium	md	kada je vidno polje jednako ili veće od 768px
Large	lg	kada je vidno polje jednako ili veće od 992px
Extra Large	xl	kada je vidno polje jednako ili veće od 1200px

Tabela 2.3. Bootstrap podela uređaja, odnosno njihovih displeja



Podaci iz tabele se ilustrativno mogu prikazati kao na slici 2.9.



Extra Small	Small	Medium	Large	Extra Large
-	sm	md	lg	xl
<576px	≥576px	≥768px	≥992px	≥1200px

Slika 2.9. Bootstrap responsive sistem

Displeji, odnosno uređaji podeljeni su u pet grupa i to u zavisnosti od širine vidnog polja. Takođe, Bootstrap koristi i specifično imenovanje različitih veličina displeja (*Small*, *Medium*, *Large*...), koje je praćeno i postojanjem odgovarajućih skraćenica – *sm*, *md*, *lg*... Ovakve skraćene odrednice koriste se prilikom formiranja brojnih klasa prilikom rada sa Bootstrapom, što će uskoro biti i ilustrovano.

Upravo prikazane *responsive* osobine unutar Bootstrapa su realizovane korišćenjem sledećih medija upita:

```
// Extra small devices (portrait phones, less than 576px)
// No media query for `xs` since this is the default in Bootstrap

// Small devices (landscape phones, 576px and up)
@media (min-width: 576px) { ... }

// Medium devices (tablets, 768px and up)
@media (min-width: 768px) { ... }

// Large devices (desktops, 992px and up)
@media (min-width: 992px) { ... }

// Extra large devices (large desktops, 1200px and up)
@media (min-width: 1200px) { ... }
```

Prikazani medija upiti ilustruju način na koji Bootstrap pravi razliku između različitih tipova uređaja. Prikazani kod je jedan od dokaza da se Bootstrap zasniva na mobile first pristupu – kod izvan medija upita odnosi se na uređaje sa najmanjim displejima, dok se medija upiti koriste za širine vidnog polja preko 575px.

## Responsive osobine kontejnera

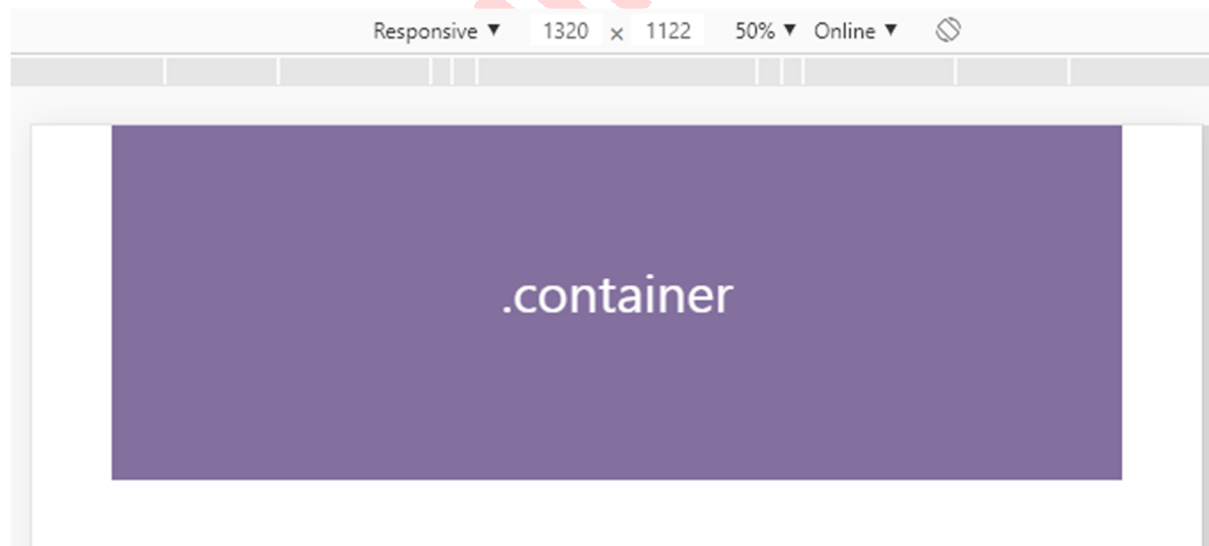
Bootstrap kontejneri samim kreiranjem dobijaju *responsive* osobine. Takve osobine su ilustrovane tabelom 2.4.

Bootstrap klasa (tip kontejnera)	Extra Small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra Large ≥1200px
.container	100%	540px	720px	960px	1140px
.container-fluid	100%	100%	100%	100%	100%

Tabela 2.4. Responsive osobine Bootstrap kontejnera

Iz tabele 2.4. se može razumeti ponašanje kontejnerskih Bootstrap elemenata na različitim uređajima, odnosno za različite širine vidnog polja. Kada se kontejner kreira korišćenjem klase `.container-fluid`, ponašanje je uvek identično. Ipak, kada se glavni okružujući Bootstrap element kreira korišćenjem klase `.container`, njegova veličina se menja u zavisnosti od širine vidnog polja. Na primer, kada je vidno polje jednako ili šire od 1200px (što po Bootstrap nomenklaturi odgovara odrednici *Extra Large* uređaja), `.container` element će imati širinu 1140px. Čim širina vidnog polja padne ispod 1200px, širina `.container` elementa se smanjuje na 960px.

U ovakvo ponašanje `.container` elementa samostalno se možete uveriti smanjivanjem širine web pregledača (animacija 2.1).



Animacija 2.1. Responsive osobine `.container` elementa

Iz tabele 2.4. i upravo prikazane animacije, može se videti da `.container` element prati promenu širine vidnog polja i tako prilikom smanjivanja dobija nekoliko unapred utvrđenih fiksnih vrednosti za svoju širinu. Kada širina vidnog polja padne ispod 576px, širina `.container` elementa postaje 100%, kako bi se što više prostora iskoristilo za prikaz sadržaja.

Ipak, programer ima mogućnost da prilikom korišćenja Bootstrap kontejnera samostalno odluči do kada će takav kontejner zauzimati 100% širine vidnog polja. Za takvo kontrolisanje kontejnera koriste se klase definisane tabelom 2.5.

<b>Bootstrap klasa</b> (tip kontejnera)	<b>Extra Small</b> <576px	<b>Small</b> ≥576px	<b>Medium</b> ≥768px	<b>Large</b> ≥992px	<b>Extra Large</b> ≥1200px
<code>.container</code>	100%	540px	720px	960px	1140px
<code>.container-sm</code>	100%	540px	720px	960px	1140px
<code>.container-md</code>	100%	100%	720px	960px	1140px
<code>.container-lg</code>	100%	100%	100%	960px	1140px
<code>.container-xl</code>	100%	100%	100%	100%	1140px

Tabela 2.5. Responsive klase za kreiranje Bootstrap kontejnera

U tabeli 2.5. se mogu videti različite klase koje nakon prefiksa `.container-` poseduju i odrednice `sm`, `md`, `lg` i `xl`. To su upravo odrednice koje odgovaraju nešto ranije prikazanom imenovanju koje Bootstrap koristi za označavanje displeja različitih veličina.

Klase iz tabele 2.5, koje se koriste za kontrolisanje *responsive* ponašanja Bootstrap kontejnera, mogu se razumeti na sledeći način: ukoliko se, na primer, iskoristi klasa `.container-lg`, Bootstrap kontejner će imati širinu od 100% vidnog polja, sve dok vidno polje ne postane širine 992px ili više.

## Responsive osobine kolona

Već je prikazano da su Bootstrap kolone osnovni elementi grid sistema unutar kojih se direktno smešta sadržaj web sajta. Što se tiče osobina Bootstrap kolona, do sada je bilo reči o njihovoj veličini. Pored pristupa koji su do sada prikazani, Bootstrap poseduje ugrađeni mehanizam koji omogućava kontrolisanje veličine kolona u zavisnosti od tipa uređaja, odnosno širine vidnog polja. Za obavljanje takvog posla koriste se već viđeni nastavci `sm`, `md`, `lg` i `xl`, koji se dodaju na klasu `.col` (tabela 2.6).

Oblik klase	Značenje
.col-*	klasa kojom se definišu osobine kolona kada je vidno polje uže od 576px
.col-sm-*	klasa kojom se definišu osobine kolona kada vidno polje ima širinu 576px ili više
.col-md-*	klasa kojom se definišu osobine kolona kada vidno polje ima širinu 768px ili više
.col-lg-*	klasa kojom se definišu osobine kolona kada vidno polje ima širinu 992px ili više
.col-xl-*	klasa kojom se definišu osobine kolona kada vidno polje ima širinu 1200px ili više

Tabela 2.6. Responsive klase za kreiranje Bootstrap kolona

U tabeli 2.6. je prikazan format klase koje se koriste za kontrolisanje Bootstrap kolona na različitim uređajima. Bitno je razumeti da u tabeli nisu prikazane klase koje je direktno moguće koristiti. Naime, karakter zvezdica je neophodno zameniti nekim brojem između 1 i 12, čime se definiše širina kolone na konkretnom uređaju. Tako opšti oblik za kreiranje klase kojima se kontroliše veličina kolona na različitim uređajima izgleda kao na slici 2.10.

**.col-\*-\***

Slika 2.10. Struktura Bootstrap klase za definisanje širine kolona na različitim uređajima; na mestu prve zvezdice navodi se odrednica koja određuje uređaj, a na mestu druge zvezdice jedan od brojeva između 1 i 12

Na mesto prve zvezdice postavlja se jedna od odrednica iz tabele: *sm*, *md*, *lg*, *xl*. Takođe, ovakvu odrednicu je moguće i izostaviti i tada se klasa odnosi na *Extra Small* uređaje. Na mesto druge zvezdice postavlja se broj u rasponu od 1 do 12, kojim se definiše veličina kolone za prethodno definisani tip displeja (uređaja). U nastavku će biti prikazano nekoliko primera klase koje je moguće postaviti na Bootstrap kolone i njihova značenja:

- .col-sm-3 – kolona koja će za širine vidnog polja 576px i više, zauzimati 3/12 ukupne širine grida;
- .col-xl-6 – kolona koja će za širine vidnog polja 1200px i više zauzimati polovinu širine grida;
- .col-md-1 – kolona koja će za širine vidnog polja 768px i više zauzimati 1/12 ukupne širine grida.

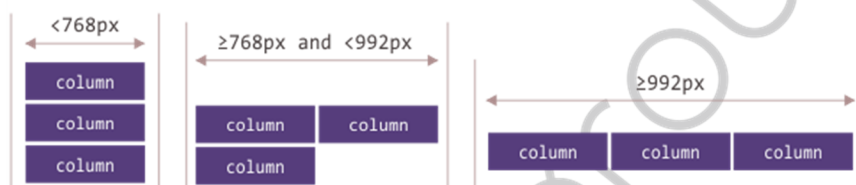
Prava moć ovakvih klasa za definisanje osobina kolona ogleda se u mogućnosti njihovog kombinovanja. Tako je moguće uraditi nešto ovako:

```
<div class="col-12 col-md-6 col-lg-4">
</div>
```

Sada su na jednom `div` elementu postavljene tri klase, kojima je definisano nekoliko osobina ovakve kolone:

- kolona će za širine vidnog polja do 767px zauzimati kompletnu širinu unutar grida (12/12);
- kolona će za širine 768px i više zauzimati jednu polovinu širine grida (6/12);
- kolona će za širine 992px i više zauzimati jednu trećinu širine grida (4/12).

Na ovaj način biće dobijen efekat kao na slici 2.11.



Slika 2.11. Ponašanje tri Bootstrap kolone sa klasama `.col-12` `.col-md-6` `.col-lg-4`

Bitno je razumeti da se zbog *mobile first* prirode Bootstrapa prikazane *responsive* klase uvek odnose na svoju kategoriju, ali i na sve kategorije iznad. Upravo zbog toga je moguće navesti `.col-12` i ovako definisana veličina kolone bi se odnosila na sve kategorije displeja – *xs*, *sm*, *md*, *lg*, *xl*. Naravno, pod uslovom da na koloni nije definisana još neka klasa sa nekom drugom kategorijom uređaja. Takva situacija postoji i u prikazanom primeru. Prvo je navedena klasa `.col-12`, a zatim i `.col-md-6` klasa. Može se primetiti da je preskočena klasa za male (*small, engl., sm*) displeje, ali zbog upravo navedene osobine klasa `.col-12` se u prikazanom primeru odnosi i na ekstra male (*xs*) i male (*sm*) displeje.

## Rezime

- Bootstrap je svoju slavu stekao zahvaljujući odličnom grid sistemu za raspoređivanje elemenata.
- Grid je osnovna struktura, odnosno kostur dizajna, a sastoji se iz nevidljivih linija čijim presecanjem se gradi struktura, odnosno sistem koji se može koristiti za raspoređivanje elemenata.
- Bootstrap grid može se sastojati iz tri grupe elemenata: kontejneri, redovi i kolone.
- Kontejneri su glavni okružujući elementi unutar kojih se direktno smeštaju redovi.
- Redovi služe kao kontejneri za kolone.
- Kolone su kontejneri za sadržaj.
- Maksimalan broj kolona Bootstrap grida je 12.
- Kreiranje Bootstrap kontejnera se može obaviti korišćenjem klasa `.container` i `.container-fluid`.
- Bootstrap redovi se kreiraju korišćenjem klase `.row`.
- Bootstrap kolone se kreiraju korišćenjem klasa koje započinju prefiksom `col`.