

|                                      |   |
|--------------------------------------|---|
| <b>Nastavni predmet</b>              | <b>SKRIPTNI JEZICI I WEB PROGRAMIRANJE</b>  |
| <b>Naslov cjeline</b>                | CSS – uređivanje web sjedišta   |
| <b>Naslov jedinice</b>               | LV24: CSS media queries i flexbox layout  |
| <b>Očekivani ishodi učenja</b>       | <ul style="list-style-type: none"><li>• opisati i primijenjivati modul flexbox,</li><li>• opisati i koristiti flex spremnik,</li><li>• dati primjer za primjenu flexbox izgleda,</li><li>• prilagoditi veličine prikaza,</li><li>• definirati i upotrebljavati različite stilove za različite uređaje,</li><li>• demonstrirati skraćene zapise vrijednosti,</li><li>• objasniti i primjenjivati media upite</li></ul> |
| <b>Ime i prezime učenika / datum</b> |   |

## UPUTE

Ako se pri traženju odgovora koristite internetom ili internetskim servisima, obavezno je navesti izvor-poveznicu. Ako je korišten generativni AI treba navesti i pitanje (prompt).

Ukoliko tekst odgovora sadrži gramatičke ili stručne nepravilnosti, to može utjecati na ocjenu, tako da je nužno tekst koji se ubacuje pročitati, proučiti i razmisliti o njemu, te ga po potrebi ispraviti. Također, nerazumijevanje prezentiranog koda će utjecati na ocjenu, pa je nužno sav kod koji se ubacuje i prezentira kao odgovor pregledati, proučiti, razmisliti o njemu i shvatiti ga, te po potrebi ispraviti, dopuniti i/ili komentirati. Za nejasnoće i/ili dodatne upute obratiti se nastavnicima. Vježbu za ocjenjivanje može se predati najkasnije do 10 minuta prije kraja sata.

## PRIPREMA ZA VJEŽBU

**Odgovorite pisano u bilježnicu:**

- ▶ 1. Ukratko objasnite div element.

## IZVOĐENJE VJEŽBE

### CSS FlexBox modul

- [Isprobajte](#) primjere
- Prije Flexbox Layout modula, postojala su četiri načina za oblikovanje izgleda: blok (za sekcije na mrežnoj stranici), inline (za tekst), tablice (za dvodimenzionalne tabelarne podatke), pozicioniranje (za eksplicitnu poziciju elementa).
- Ovaj modul **pojednostavnjuje dizajniraje fleksibilne responzivne strukture** izgleda bez korištenja float ili pozicioniranja.
- Flexbox elementi: **Flexbox model započinje definiranjem flex spremnika** (roditeljskog elementa), npr. za tri flexbox stavke (u nastavku). Flex spremnik **postaje fleksibilan postavljanjem svojstva display na flex**:

```
<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div>3</div>
</div>
```

```
.flex-container {
  display: flex;
}
```

Flex kontejner svojstva su:

- [flex-direction](#) – **definira smjer** u kojem će se slagati flex stavke, vrijednost **column** postavlja ih vertikalno (od vrha do dna), a **column-reverse** obrnuto. Uz to postoje još i mogućnosti row i row-reverse.
- [flex-wrap](#) – određuje hoće li se prema potrebi flex stavke **omotavati ili prebacivati u novi red**. Vrijednost može biti **wrap** i **nowrap**, te **wrap-reverse**.
- [flex-flow](#) – ovo svojstvo je **skraćeno svojstvo** za [flex-direction](#) i [flex-wrap](#).
- [justify-content](#) – koristi se za **poravnanje flex stavki** (**center**, **flex-start**, **flex-end**, **space-around**, **space-between**)
- [align-items](#) – koristi se za **poravnanje flex stavki** (**center**, **flex-start**, **flex-end**, **stretch**, **baseline**).
- [align-content](#) – za **poravnanje flex linija** (npr. **space-between** vrijednost ispisuje flex linije sa jednakim razmakom između njih, **space-around** ispisuje sa razmakom prije, između i nakon njih, **stretch** razvuče flex linije kako bi zauzele preostali prostor (podrazumijevano), vrijednost **center** ispisuje flex linije u sredini spremnika, **flex-start** vrijednost ispisuje flex linije na početku spremnika, **flex-end** ispisuje na kraju)

Primjer za savršeno centriranje:

```
.flex-container {
  display: flex;
  height: 300px;
  justify-content: center;
  align-items: center;
}
```

- **Izravni potomci flex spremnika automatski postaju fleksibilne stavke (flex).**

Svojstva flex stavki su:

- **order** - određuje **poredak** flex stavki. Prva stavka u kodu ne mora se pojaviti kao prva u izgledu. Vrijednost za poredak mora biti broj, podrazumijevana vrijednost je 0. Primjer: `<div style="order: 1">4</div>`
- **flex-grow** - određuje koliko će se flex stavka **proširivati relativno prema ostalima**. Vrijednost mora biti broj, podrazumijevana vrijednost je 0. Primjer: `<div style="flex-grow: 8">3</div>`
- **flex-shrink** – određuje koliko će se flex stavka **sužavati relativno prema ostalima**. Vrijednost mora biti broj, podrazumijevana vrijednost je 1.
- **flex-basis** – određuje **inicijalnu dužinu** flex stavke. Primjer: `<div style="flex-basis: 200px">3</div>`
- **flex** – skraćeno svojstvo za **flex-grow**, **flex-shrink** i **flex-basis**. Primjer: `<div style="flex: 0 0 200px">3</div>`
- **align-self** – određuje **poravnanje** za odabranu stavku unutar flex spremnika. Ovo svojstvo pregazi podrazumijevano poravnanje postavljeno sa **align-items** svojstvom spremnika.

Primjer: `<div class="flex-container">  
 <div>1</div>  
 <div style="align-self: flex-start">2</div>  
 <div style="align-self: flex-end">3</div>  
 <div>4</div>  
</div>`

## CSS media queries

- **CSS2** uvodi tipove medija.
- **@media** rule omogućuje **definiranje različitih pravila** za stilove koji se primjenjuju **za različite tipove medija**. Npr, može se postaviti jedan skup pravila oblikovanja za računalne ekrane, drugi za pisače, treći za neke druge uređaje itd.
- **CSS3** je uveo Media Queries. Media queries u CSS3 proširuje CSS2 media types ideju: Umjesto da se prati tip uređaja, usredotočuje se na njegove **moćnosti**. Media queries mogu se koristiti za provjeru:
  - Širine i visine viewport-a
  - Širine i visine uređaja
  - orijentacije (landscape/portrait)
  - rezolucije
- Upotreba media queries je popularna tehnika za primjenjivanje kod mobilnih uređaja (npr. iPhone i Android telefona).

## Media Query Syntax

Media query sastoji se od media type i može sadržavati jedan ili više izraza koji mogu poprimiti vrijednost true ili false.

```
@media not|only mediatype and (expressions) {  
  CSS-Code;  
}
```

Rezultat upita je istina ako navedeni tip medija odgovara tipu uređaja na kojem se dokument ispisiuje i ako su svi izrazi u upitu istiniti. Kad je media query true, primjenjuje se odgovarajući stil ili pravila oblikovanja kao što je to uobičajeno.

Ako se ne koristi **not** ili **only** operatore, media type je opcionalan i primjenjuju se svi tipovi.

Mogu postojati različita oblikovanja (stylesheets) za različite medije:

```
<link rel="stylesheet" media="mediatype and|not|only (expressions)" href="print.css">
```

### Jednostavni media queries primjeri

```
@media screen and (min-width: 480px) {  body {  
  background-color: lightgreen;  
  }  
} //mijenja boju pozadine
```

```
@media screen and (min-width: 480px) {  
  #leftsidebar {width: 200px; float: left;}  
  #main {margin-left: 216px;}  
} //mijenja izgled navigacije
```

- Proučite i isprobajte ([p1](#), [p2](#)) [primjer3](#).
- Proučite [primjere](#) za realizaciju responzivnosti.

### PONAVLJANJE I PRAKTIČNA VJEŽBA:

1. U online [editoru](#) provježbajte naučeno.
2. Prema [primjeru](#) napravite mrežnu stranicu (galeriju) sa tematski i koloristički usklađenim odabirom od najmanje dvanaest kvalitetnih slika jednake širine, a različitih visina. Slike trebaju biti ravnomjerno udaljene jedne od drugih. Prikaz treba biti responzivan.

Ako ostane vremena napravite dva rada (za svakog učenika po jedan) ili prema [primjeru](#) napravite mrežnu stranicu o sebi.