



**FER3**

**Preddiplomski studij**

**Računarstvo**

## **Razvoj programske podpore za web**

Druga laboratorijska vježba – JavaScript  
– modifikacija pripreme

## 1 Uvod

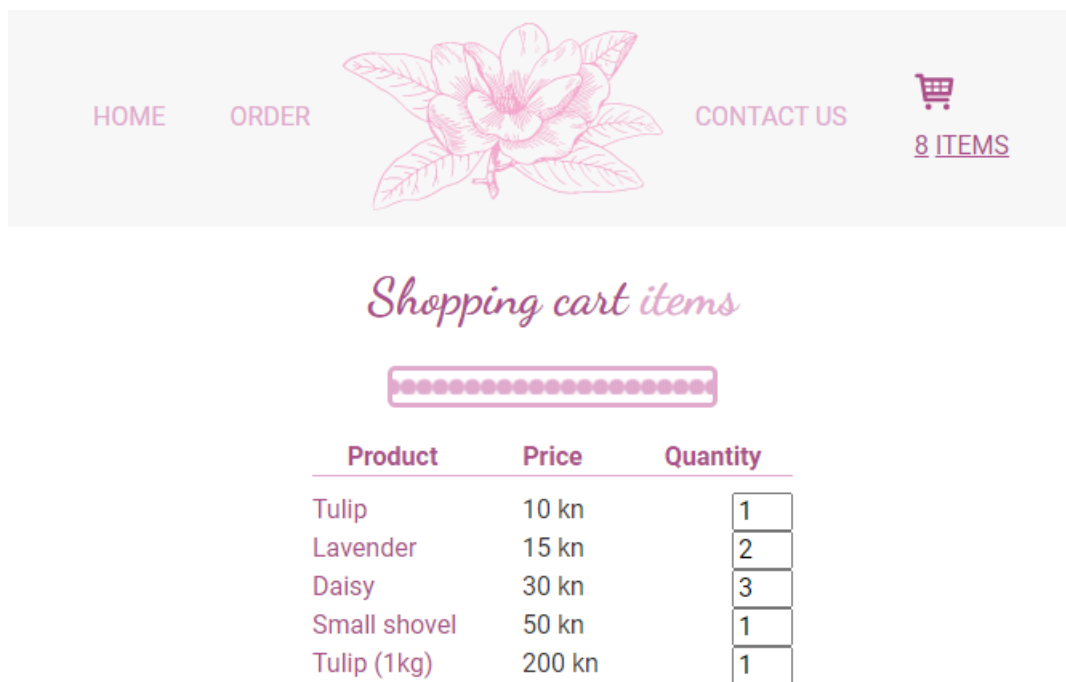
Uz izradu pripreme, potrebno je modificirati pripremu. Modifikacija pripreme sadrži košaricu i varijaciju modifikacije.

**Napomena:** Nije potrebno implementirati vlastite stilove i strukturu HTML-a, na Moodleu zajedno s dokumentom zadatka priložene su datoteke: *cart.html* i *cart.js* datoteke. HTML predlošci ili isječci za zadatke su sadržani u *cart.html*, dok *main.css* iz pripreme sadrži unaprijed definirane CSS klase potrebne za zadatke. Za potrebe laboratorijske vježbe potrebno je izmijeniti samo *js* datoteke.

## 2 Zadatak - stranica s pregledom košarice

Za prikaz košarice potrebno je izmijeniti datoteku **cart.js**. Postupak implementacije je sličan pripremi, potrebno je dodati JavaScript kôd koji nedostaje označen s `INSERT CODE HERE`.

Pretpostavka za prikaz stranice je košarica pohranjena u lokalnom spremniku iz zadatka zadanog u pripremi. Podaci u košarici se dohvaćaju pomoću funkcije `getCart` koju je potrebno napisati. Od funkcije `getCart` očekuje se mapa u obliku `Map([[id, količina], ... ])`, npr. mapa `Map([[ 1, 5], [3, 2]])` sadrži proizvode s identifikatorima 1 i 3 te pripadajućim količinama 5 i 2.



Slika 1: Stranica za pregled košarice **cart.html**

Kako bi se stavke košarice prikazale potrebno je dodati JavaScript kôd za dohvat JSON podataka o proizvodu (`INSERT CODE HERE`) iz lokalne datoteke *lab2.json*. U primjeru 1 za proizvod *Tulip* prikazani su njegovi JSON podaci.

```
{
  "id": 1,
  "name": "Tulip",
  "price": 10,
  "categoryId": 1,
  "imageUrl": "images/tulip.jpg"
}
```

Primjer 1: JSON podaci o proizvodu *Tulip*

Nakon dohvata podataka, manipulacijom DOM-a pomoću gotovih predložaka prikazanih na primjeru 2 prikazuju se zaglavlje i stavke košarice. Stavka košarice se sastoji od naziva proizvoda, njegove cijene i trenutne količine u košarici. Potrebno je dodati samo naredbe (INSERT CODE HERE) koje postavljaju cijenu proizvoda.

```
<template id="cart-template-header">
  <div class="cart-header">
    <div class="cart-header-title">Product</div>
    <div class="cart-header-title">Price</div>
    <div class="cart-header-title">Quantity</div>
  </div>
</template>
<template id="cart-template-item">
  <div class="cart-item">
    <div class="cart-item-title">Tulips</div>
    <div class="cart-item-price">15 kn</div>
    <input type="number" class="cart-item-quantity"></input>
  </div>
</template>
```

Primjer 2: HTML predlošci za zaglavlje i sadržaj košarice

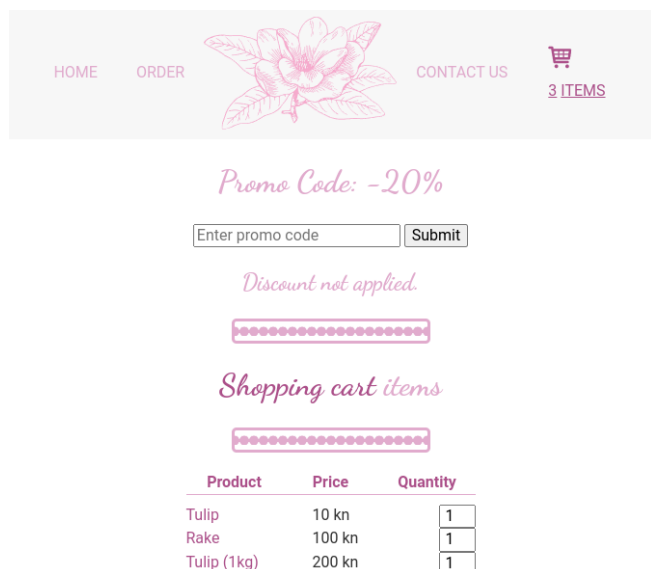
### 3 Varijacija modifikacije - Sniženje cijena putem promotivnih kôdova

Potrebno je dodati funkcionalnost upisivanja promotivnog kôda u svrhu ostvarivanja popusta na odabrane proizvode u košarici.

- Ako korisnik upiše ispravan promotivni kod *WEBLAB2*, ostvaruje popust na sve trenutne, ali i buduće proizvode u košarici, kao što je prikazano na slici 3. Pri tome se tekst stanja primjene promotivnog kôda mijenja u: *20% Discount applied. Enjoy!*
- Ako korisnik ne upiše promotivni kod ili pak upiše neispravan kod cijene proizvoda se vraćaju na originalan iznos - slika 2. Tekst stanja se mijenja na: *Discount not applied.*
- Stanje ostvarenosti popusta mora biti očuvano u lokalnom spremniku (npr. odlazak na *index.html* i povratak na prikaz košarice ne utječe na stanje ostvarenosti popusta).

#### Napomena

- Pritisak na tipku *Submit* mijenja prikaz košarice (tekst stanja i iznose proizvoda) ako prelazimo iz stanja bez popusta u stanje s popustom i obratno.
- Za rješavanje ovog zadatka potrebno je promijeniti samo *cart.js*.



Slika 2: Početni prikaz košarice, korisnik nije upisao promotivni kôd.



Slika 3: Promotivni kôd je uspješno aktiviran i popust je primijenjen.