

Web programiranje – Vežbe 7

JQuery i AJAX

Primer – registracija korisnika

HTML stranica **users.html** sadrži tabelu sa listom korisnika dobijenu sa servera. HTML stranica **registration.html** sadrži formu za registrovanje korisnika. Novi korisnik se šalje preko mreže serveru. Ovi fajlovi se nalaze u **WebContent** folderu **OnlineService** projekta.

JQuery je biblioteka za JavaScript koja služi da olakšava programiranje klijentskih aplikacija. Za početak je potrebno uključiti ovu biblioteku u HTML. Poslednja verzija JQuery biblioteke je dostupna [ovde](#) (kompresovana verzija je za prave projekte, nama je svejedno). Dodati kod ispod u head element fajla **users.html**.

```
<script src="jquery-3.3.1.min.js"></script>
```

NAPOMENA: JQuery fajl uključiti pre vaših JavaScript fajlova da bi bio dostupan u njima.

Glavna funkcija JQuery-ja je znak dolara: **\$**. Ova funkcija se najčešće koristi za selektovanje elemenata iz DOM stabla i funkcionalno je ekvivalentna pozivu **document.querySelector**. Selektori su dostupni [ovde](#). Bitna razlika je u tome što vraćeni objekti nisu čisti DOM elementi, već su to JQuery objekti koji ih reprezentuju (ovo se zove *wrapper* objekat). Sve funkcije nad DOM objektima su moguće i olakšane kroz JQuery: [obrada događaja](#), [atributi i klase](#), [css](#), [dodavanje/manipulacija DOM stablom](#) itd. Puna dokumentacija je dostupna [ovde](#). Kod ispod postavlja boju i bolduje zaglavlja tabele. Dodati ga u fajl **users.js**.

```
// $(document).ready(function(){ ... }) je ekvivalentan kodu
document.addEventListener('ready', function(){ ... })
$(document).ready(function() {
    // Postavi stil th elementima kroz JQuery
    $('th').css('background', 'lightblue').css('font-weight', 'bold');
})
```

AJAX (Asynchronous JavaScript And XML) je biblioteka u okviru JQuery i služi za klijent-server komunikaciju. Asinhrona je u smislu da kada se zahtev pošalje na server, program nastavlja normalno da se izvršava. Odgovor se zatim obrađuje tek kada ga server vrati, jer bi se u suprotnom čitava aplikacija zamrzнула dok se ne dobije odgovor. Funkcija koja se poziva kada se vrati odgovor se zove **callback**.

Kroz AJAX se šalju **JSON objekti** (JavaScript Object Notation). To su objekti kao što se definišu u JavaScriptu: objekti u obliku {key: value}, a liste u obliku [obj1, obj2...]. AJAX može da šalje zahtev na server na dva načina:

- [\\$.get](#) ili [\\$.post](#) – Šalju GET/POST zahtev na server. Prima sledeće parametre (u ovom redosledu):
 1. **url** – Putanja do resursa na koji se šalje zahtev.

2. **data** – Sadržaj zahteva: parametri zahteva (iza upitnika u URL-u) za GET i telo HTTP zahteva za POST. Podrazumevano je da je JSON objekat u
 3. **success** – Funkcija koja će se pozvati kada se dobije odgovor na zahtev (callback funkcija). Prvi parametar ove funkcije je sadržaj odgovora koji se može čitati u toj funkciji.
- **\$.ajax** – HTTP zahtev bilo koje vrste. Moguće je menjati url, sva zaglavlja, metodu i sadržaj zahteva. Ovo je generalizovana verzija gorepomenutih funkcija.

tabele. Dodati ga u fajl **users.js**.

Kod ispod šalje zahtev na servlet u Java projektu **OnlineService** i popunjava tabelu korisnicima dobijenim u odgovoru. Ovaj Servlet se nalazi u paketu **servlets** i definisan je klasom **UserServlet**. Služi da lista korisnike i da ih registruje. Dodati ovaj kod u **\$(document).ready** funkciju u fajlu **users.html**.

```
//Šalje se GET zahtev na adresu ispod
$.get({
  url: '/OnlineService/UserServlet',
  success: function(data) {
    // Odgovor se nalazi u data
    // Pošto očekujemo JSON listu, možemo da iteriramo kroz data
    for (let user of data) {
      let tbody = $('#korisnici tbody');
      // Kreiramo DOM elemente za svakog korisnika uz
      // pomoć JQuery biblioteke
      let firstName = $('<td>' + user.firstName + '</td>');
      let lastName = $('<td>' + user.lastName + '</td>');
      let username = $('<td>' + user.username + '</td>');
      let email = $('<td>' + user.email + '</td>');
      let phoneNumber = $('<td>' + user.phoneNumber + '</td>');
      // Dodajemo sva polja u red, koji dodajemo u telo tabele
      let tr = $('<tr></tr>');
      tr.append(firstName)
        .append(lastName)
        .append(username)
        .append(email)
        .append(phoneNumber);
      tbody.append(tr);
    }
  }
});
```

Zadaci

1. Preraditi fajl **registration.html** tako da polje u koje se trenutno unosi vrednost ima obojenu pozadinu, kao na slici **registracija.jpg**. HINT: pogledati događaje vezane za input.
2. Validirati formu da:
 - Ime i prezime počinju velikim slovom. Ako polje nije validno, pored njih crvenim slovima prikazati poruku "Ime i prezime moraju početi velikim slovom!".
 - Broj telefona je broj. Ako polje nije validno, pored njega crvenim slovima napisati "Morate uneti broj!"
 - Ostala polja ne smeju biti prazna. Ako su polja prazna, pored njih crvenim slovima napisati "Morate uneti vrednost!"

Validacija se vrši na klik submit dugmeta.

3. Preraditi formu za registraciju tako da se:
 1. Spreči podrazumevano ponašanje forme.
 2. Ako je forma validna, pošalje se POST zahtev na url **/OnlineService/UserServlet** koji sadrži unešene podatke o korisniku (iz input polja tabele).
 3. Pri odgovoru od servera prikazati dijalog "Korisnik <username> uspešno registrovan!"

Korisni linkovi

- JQuery dokumentacija: <https://api.jquery.com/jquery.ajax/>
- JQuery download stranica: <https://jquery.com/download/>
- AJAX dokumentacija: <https://api.jquery.com/category/ajax/>