

Uvod u Veb i Internet tehnologije

JavaScript – jQuery biblioteka

Filip Marić
Vesna Marinković
Milan Banković

jQuery osnove

- jedna od najčešće korišćenih JavaScript biblioteka
- osnovni ciljevi JQuery biblioteke:
 - pojednostavljena manipulacija DOM elementima i njihovim svojstvima
 - jednostavno obavljanje uobičajenih zadataka u JavaScript-u (animacije, AJAX, upravljanje događajima)
 - nezavisnost od pregledača
- <https://jquery.com/>

Korišćenje jQuery biblioteke

- JQuery biblioteka se preuzima sa veb lokacije:
<https://jquery.com/download/>
- Kompresovana verzija:
 - preporučljivo koristiti jer se brže učitava
 - <https://code.jquery.com/jquery-3.3.1.min.js>
- Nekompresovana verzija:
 - čitljiva verzija – korisna kao početna tačka za dalji razvoj
 - <https://code.jquery.com/jquery-3.3.1.js>
- Biblioteka se uključuje na kraju body elementa, na uobičajen način:

```
<body>
...
<script
  src="jquery-3.3.1.min.js" type="text/javascript">
</script>
</body>
```

jQuery kolekcije

- Osnovne strukture podataka u jQuery biblioteci su **jQuery kolekcije**
- jQuery kolekcija sadrži jedan ili više JavaScript objekata koji predstavljaju HTML DOM elemente
- Nudi uniforman interfejs za pristup HTML elementima
- Metode jQuery kolekcija obično vraćaju tu istu kolekciju, što omogućava jednostavno **ulančavanje metoda**
- Elementima u jQuery kolekciji se može manipulirati istovremeno (npr. promeniti im boju, postaviti rukovalac događajem, i sl.)

jQuery() funkcija

- Osnovna funkcija u JQuery biblioteci je `jQuery()` funkcije (kraći naziv ove funkcije je `$()`)
- Ima više mogućih primena ove funkcije:
 - `$("selektor")` – vraća JQuery kolekciju koja sadrži sve elemente u HTML dokumentu koji zadovoljavaju dati CSS selektor
 - `$("html kod")` – vraća JQuery kolekciju koja sadrži HTML elemente u datom HTML kodu
 - `$(dom_element)` – za dati JavaScript objekat koji predstavlja DOM element vraća JQuery kolekciju koja sadrži samo taj element
 - `$(funkcija)` – data funkcija se izvršava kada se učita ceo dokument

jQuery selektori

- jQuery selektori su zasnovani na CSS selektorima, uz neke dopune
- Svi selektori počinju znakom \$ i navode se u zagradama: `$(...)`
- Osnovni selektori:
 - ime elementa: `$("p")` – kolekcija svih p elemenata u dokumentu
 - identifikator elementa: `$("#istaknuti_pasus")` – kolekcija sa jedinstvenim elementom
 - identifikator klase: `$(".istaknut")` – kolekcija svih elemenata sa klasom istaknut
- Dodatni selektori:
 - `$("*")` – označava sve elemente
 - `$("p:first")` – prvi element `<p>`
 - `$("tr:even")` i `$("tr:odd")` – parni i neparni redovi tabele
 - `$(" [src]")` – svi elementi sa atributom src
 - `$(" a[target='_blank']")` i `$(" a[target!='_blank']")` – svi linkovi kod kojih je vrednost atributa target jednaka, odnosno nije jednaka `'_blank'`

jQuery i DOM elementi

- Kreiranje JQuery kolekcije koja sadrži neki DOM element `e1` postiže se sintaksom `$(e1)`
- Ovo je zgodno ako želimo da manipuliramo elementom koristeći jednostavan i moćan JQuery interfejs (nasuprot standardnom DOM interfejsu)
- Sintaksom `$("<div></div>")` kreira se JQuery kolekcija koja sadrži samo jedan element `div` koji nije u stablu dokumenta
- Ovako kreiran element se može naknadno ubaciti u stablo na željenu poziciju

Dodavanje novih elemenata u stablo

- Elementi se mogu dodavati u stablo tako što se dodaju u sadržaj nekog elementa
- U tu svrhu se koriste metode `append`, `prepend`, `before` i `after`
- Primer:

```
$("#mydiv").append($("#<p>Hello JQuery</p>"));  
$("#mydiv").before($("#<p>Hello again</p>"));
```

- Sadržaj nekog elementa se može potpuno zameniti novim, metodom `html`:

```
$("#mydiv").html("<span>Novi sadrzaj</span>");
```

- Metoda `remove` uklanja iz stabla i dealocira sve objekte date kolekcije
- Metoda `detach` uklanja iz stabla elemente, ali ih ne dealocira:

```
var dspan = $("#myspan").detach();  
$("#mydiv").append(dspan); // premestamo span u div
```


Manipulacija CSS-om korišćenjem jQuery-a

- `addClass()` – dodaje jednu ili više klasa datom elementu
- `removeClass()` – uklanja jednu ili više klasa sa datog elementa
- `toggleClass()` – naizmenično dodaje i uklanja jednu ili više klasa sa datog elementa
- `css()` – postavlja ili vraća stilski atribut
 - vrednost svojstva se vraća: `css("naziv_svojstva")`
 - vrednost jednog svojstva se postavlja sa:
`css("naziv_svojstva", "vrednost")`
 - vrednost više svojstava se postavlja sa:
`css({"naziv_svojstva": "vrednost", "naziv_svojstva": "vrednost", ...})`
- Proizvoljni atribut elementa se može na sličan način očitavati i postavljati `attr()` metodom

jQuery događaji

- postavljanje rukovaoca: `on()` metoda:

```
$("#mydiv").on("click", function(event)
    { alert("Zdravo"); });
```

- uklanjanje rukovaoca: `off()` metoda
- Za neke česte tipove događaja postoje i posebne metode:

```
$("#mydiv").click(function(event)
    { alert("Zdravo"); });
```

```
$("#mydiv").mouseover(function(event)
    { "Coords: " + alert(event.clientX + " " +
```

- Događaji se mogu izazivati `trigger()` metodom

jQuery efekti

- HTML elementi se mogu sakriti i ponovo prikazati metodama `hide()`, `show()` i `toggle()`
- Element se može postepeno pojavljivati ili postepeno gubiti sa slike metodama `fadeIn()`, `fadeOut()`, `fadeToggle()`
- Element se može postepeno pojavljivati “klizanjem naniže”, odnosno postepeno nestajati “klizanjem naviše” metodama `slideUp()`, `slideDown()`, `slideToggle()`
- Sve ove funkcije mogu imati vrednost argumenata “slow”, “fast” ili broj milisekundi
- Metod `stop()` se koristi za zaustavljanje efekta pre nego što se završi

Ulančavanje metoda

- U jQuery-u je moguće ulančavati akcije/metode čime se u jednoj naredbi izvršava veći broj metoda na istom elementu
- Ovo je moguće jer metode za postavljanje vrednosti vraćaju objekat na kome su pozvane
- Na ovaj način pregledači ne moraju više puta da pronalaze isti element
- Primer:

```
$("#mydiv").css("color", "black")  
            .css("background-color", "red")  
            .text("Hello world");
```