Prezentacija:

jQuery III

Plugin-ovi

- jQuery plugin predstavlja metodu koju koristimo za proširivanje jQuery prototype objekta
- Na ovaj način svi jQuery objekti će naslediti sve metode koje budemo definisali
- Kao što smo već pričali, svaki put kada pozovete jQuery() , kreirate jQuery objekat sa svim nasleđenim metodama
- Danas postoje na hiljade i hiljade gotovih pluginova koje rešavaju određene probleme u frontend-u

Plugin-ovi

- Na sledećim sajtovima možete pronaći neke jQuery pluginove:
  - https://plugins.jquery.com
  - http://jquery-plugins.net
  - http://www.unheap.com
  - http://www.jque.re
  - http://jquer.in

- Ukoliko plugin za vaš problem nepostoji, bićete prinuđeni da sami nađete rešenje
- Da bi kod mogao što bolje da se iskoristi, bilo bi idealno napisati plugin

Plugin-ovi

- U kodu ispod možete videti jedan prost jQuery plugin:

```
$.fn.pocrveni = function() {
    this.css( "color", "red" );
};

$( "a" ).pocrveni(); // Ofarbaće sve linkove u crveno
```

Plugin-ovi

```
$( "a" ).pocrveni().addClass('red');
```

- Da bi lančanje profunkcionisalo, moramo vratiti nazad jQuery objekat :

```
$.fn.pocrveni = function() {
    this.css( "color", "red" );
    return this;
};
```

Plugin-ovi

- $ (dolar) promenjiva je dosta popularna među ostalim JavaScript bibliotekama, to može dovesti do konflikta
- Da bi rešili ovaj problem, potrebno je obaviti plugin na sledeći način:

```
(function ( $ ) {

var crvena = "#FF0000";

$.fn.pocrveni = function() {
    this.css( "color", crvena );
};

}( jQuery ));
```

Plugin-ovi

- Dobra je praksa pisati pluginove koji će zauzimati samo jedan “slot“ unutar $.fn objekta
- To možete videti u primeru ispod:

```
(function( $ ) {

    $.fn.openPopup = function() {
        // Open popup code.
    };

    $.fn.closePopup = function() {
        // Close popup code.
    };

}( jQuery ));
```

```
(function( $ ) {

    $.fn.popup = function( action ) {
        if ( action === "open") {
            // Open popup code.
        }

        if ( action === "close" ) {
            // Close popup code.
        }
    };

}( jQuery ));
```

Plugin-ovi

- Tipičan scenario u radu sa jQuery bibliotekom jeste dohvatanje više DOM elemenata tj. kolekcija
- Ukoliko je potrebna veća manipulacija tih elemenata biće potrebno koristiti $.each() petlju

```
(function ( $ ) {

$.fn.nekiPlugin = function() {
    return this.each(function() {
        // ovde raditi sa svakim pojedinacnim elementom
    });
};

}( jQuery ));
```

Plugin-ovi

- Kako vaši pluginovi postaju sve kompleksniji, dobra ideja je dati mogućnost pluginovima da se postave određene opcije:

```
(function ( $ ) {

$.fn.ofarbaj = function(options) {
    var settings = $.extend({
        // podrazumevane vrednosti
        boja: "#FF0000",
        klasa: "red"
    }, options );
    this.css( "color", boja ).addClass(klasa);
};

}( jQuery ));
```

Grunt

- Grunt predstavlja JavaScript Task Runner
- Najčešće se koristi za izvršavanje različitih zadatke vezanih za front end
- Neki od njih su:
  - minifikacija / optimizacija css-a
  - minifikacija / optimizacija js-ova
  - JShint provera
  - kopiranje fajlova
  - kompajliranje coffeescript-a
  - kompajliranje less fajlova
  - kompajliranje sass fajlova
  - ...

Grunt

- Pošto grunt napisan u Node-u, potrebno je i njega prethodno instalirati:

  https://nodejs.org

- Instalacija Grunt-a:

  npm install -g grunt-cli

Grunt

- Pošto je napisan u Node-u, potrebno je napraviti package.json fajl, koji se sastoji od svih grunt modula koje ćemo koristiti:

```
{
  "name": "ime-mog-projekta",
  "version": "0.1.0",
  "devDependencies": {
    "grunt": "~0.4.5",
    "grunt-contrib-jshint": "~0.10.0",
    "grunt-contrib-uglify": "~0.5.0"
  }
}
```

- Pozivanjem sledeće komande se instaliraju potreni paketi iz package.json fajla:
- npm install

Gruntfile.js

```
module.exports = function(grunt) {
  // Konfiguracija
  grunt.initConfig({
    pkg: grunt.file.readJSON('package.json'),
    uglify: {
      options: {
        banner: '/*! <%= pkg.name %> <%= grunt.template.today("yyyy-mm-dd") %> */\n'
      },
      build: {
        src: 'src/<%= pkg.name %>.js',
        dest: 'build/<%= pkg.name %>.min.js'
      }
    }
  });
  // Ucitaj plugin za "uglify" task
  grunt.loadNpmTasks('grunt-contrib-uglify');
  // Podrazumevani taskovi
  grunt.registerTask('default', ['uglify']);
};
```

Jquery Boilerplate

- Danas postoji veći broj već gotovih šablona za pisanje jQuery pluginova
- Njihova struktura zavisi od problema koji treba da reše
- Jedan od najpoznatijih je jQueryBoilerplate koji sadrži predefinisanu strukturu foldera, grunt fajl, kao i sam plugin
- Takođe poseduje “comand-line“ generator
- Više o projektu možete pronaći na sledećem linku:
- http://jqueryboilerplate.com

PITANJA?