Prezentacija:

GIT

Uvod

- GIT predstavlja distibuted version control system – sistem za deljenje koda
- Kreirao ga je Linus Torvalds, kreator Linux krenela
- Repozitorujum (repository) predstavlja kolekciju komitova I svih izmena koje se nalaye na vašoj mašini ili na udaljenom serveru
- Komit (commit) predstavlja “snimak“ tj. Trenutno stanje fajlova u određenom vremenskom trenutku

Korišćenje GIT-a

- Podešavanje GIT-a
  - git config –global user.name “Marko Markovic”
  - git config –global user.email marko@markovic.com
- Neke od osnovnih komandi:
  - git init – inicijalizacija repozitorijuma u željenom folderu
  - git status – provera trenutnog stanja u git repozitorijumu
  - git add – dodavanje fajla, odnosno spremanje pre samog komita
  - git commit -m – dodatkom -m omogućujemo da odmah otkucamo poruku prilikom komitovanja
  - git push – postavljanje novih komitova na udaljeni server
  - git log – pregled svih logova, komitova, vremena kada je ko komitovao, kao i onsnovne informacije o autoru

Programi za GIT

- Standalone Programi
  - SmartGit
  - GitEye
  - GitHub for Windows/Mac
  - Tower
  - ...
- IDE rešenja sa integrisanim GIT klijentima
  - Netbeans
  - Eclipse (http://eclipse.github.io/)
  - Webstorm
  - ...

GIT hosting

- Da bi delili kod koji napišete sa vašim timom, potrebno ga je postaviti na server

- GitHub - https://github.com
- Bitbucket - https://bitbucket.org

- Gogs - http://gogs.io

Prezentacija:

JavaScript

Sadržaj

- JavaScript
  - Uvod
  - Promenjive i osnovni tipovi podataka
  - Nizovi
  - Operatori
  - Objekti
  - If-else, switch i ostale naredbe
  - Petlje
  - Funkcije
  - Bitni objekti

Uvod

- JavaScript je skriptni programski jezik.
- Predstavlja dinamički, slabo tipiziran jezik sa skromnom podrškom za objektno orijentisano porgramiranje.
- Izvršava se unutar web brauzera, ne na web serveru.
- Uključuje se unutar web stranica ili se može referencirati kao zaseban fajl.
- Koristi se za dodavanje dinamike i interaktivnosti na klijentskoj strani.

Promenjive i osnovni tipovi podataka

- Deklaracija promenjivih:
  - **var** name = "Marko";

- Tipovi podatka:
  - Celi brojevi (integer)
  - Realni brojevi (float)
  - Logički (boolean)
  - Stringovi (string)

- Složeni tipovi:
  - Nizovi
  - Objekti

Konverzija tipova podataka

- Ne morate da specificirate tip podataka. Podatak se automatski konvertuje u tip podatka prilikom izvršavanja
- var odgovor = 42;
  odgovor = "Ovo je string";
- Pri operacijama koje uključuju brojeve i stringove, broj se automatski pretvara takodje u string
- x = "Ovo je string " + 42;
- y = 42 + " je string.";

Celi brojevi (integer)

- Celi brojevi (integer) se mogu izraziti kao decimalnom, oktalnom I heksadecimalnom obliku.
- Primeri:
  - var dec = -25;
  - var okta = 033;
  - var heksa = 0xFFF;
- Funkcije:
  - parseInt() - parseInt("234")
  - toString() - dec.toString()

Realni brojevi (float)

- Realni brojevi (float) se sastoje od celog broja, decimalne tačke i frakcije.
- Primeri:
  - var real = 3.1415; real = -3.1E12; real = .1e12; real = 2E-12;
- Funkcije:
  - parseFloat() - parseFloat("3.14")
  - poString() - real.toString()

Logički (boolean)

- Logički (boolean) tip može imati dve vrednosti: **true** and **false**.
- Primeri:
  - var isActive = true;
  - var isVisible = false;

Stringovi (String)

- Stringovi (string) predstavljaju nula ili više karaktera koji su zatvoreni sa jednostrukim (') ili **dvostrukim** (") navodnicima.
- Primeri:
  - var test = "test";
  - var test = 'test';
  - var test = "1234";
  - var test = "prva \n druga";
  - Var test = "O.Š. \"Petar Kočić\" "
- Atributi:
  - length – vraća dužinu stringa

Karakter	Značenje
\n	Novi red
\t	Tabulator
\\	Znak " \ "

Stringovi (String)

- Funkcije:
  - charAt() – vraća karakter stringa sa određenog indeksa
  - concat() -spaja dva ili više stringova i vraća kopiju spojenog stringa
  - match() - pretražuje string na osnovu regularnog izraza
  - replace() – zamenjuje podstring u stringu sa nekim drugim
  - slice() - odseca deo stringa i vraća novi string
  - split() - pravi niz od stringa, seče string na određeni seprator
  - substr() - vraća podstring stringa, od početnog indeksa do određene dužine
  - substring() - vraća podstring stringa, između dva indeksa
  - toLowerCase() - pretvara karaktere stringa u mala slova
  - toUpperCase() - pretvara karaktere stringa u velika slova
  - toString() - vraća vrednost stringa
  - trim() - čisti praznine iz stringa

Nizovi (Arrays)

- Niz predstavlja objekat koji se sastoji od više vrednosti i ima različite atribute.
- Svaka vrednost se čuva kao poseban element, početni indeks elementa kreće od nule.
- Deklaracija nizova:
  - var niz = new Array();
  - var niz = [];
- Inicijalizacija elementa niza:
  - var niz = new Array();
  - niz[5] = 'plava';
- Atributi:
  - length – vraća broj elemenata niza

Nizovi (Arrays)

- Funkcije:
  - concat() - spaja dva ili više nizova i vraća kopiju spojenog niza
  - indexOf() - pretražuje niz po elementu i vraća njegovu poziciju(index)
  - join() - spaja sve elemente niza u string
  - lastIndexOf() - pretražuje niz po elementu, sa kraja i vraća njegovu poziciju(index)
  - pop() - uklanja poslednji element niza i vraća taj element
  - push() - dodaje element na kraj niza i vraća novi broj elemenata niza
  - reverse() - okreće raspored elemenata u nizu
  - shift() - uklanja prvi element niza i vraća taj element
  - slice() - selektuje deo niza i vraća ga kao novi niz
  - sort() - sortira elemente niza
  - splice() - dodaje/uklanja elemente iz niza
  - toString() - pretvara string u niz i vraća novi string
  - unshift() - dodaje element na početak niza i vraća novi broj elemenata niza

Operatori

- JavaScript poseduje više različitih operatora:
  - Operatori dodele
  - Operatori poređenja
  - Artitmetički operatori
  - Logički operatori

- Operatori dodele
  - x += y x = x + y
  - x -= y x = x - y
  - x *= y x = x * y
  - x /= y x = x / y

Operatori

- Operatori poređenja
  - Jednako (= =)
  - Različito (!=)
  - Veće (>)
  - Veće ili jednako (>=)
  - Manje (<)
  - Manje ili jednako (<=)
- Aritmetički operatori
  - myVar++
  - myVar--

Operatori

- Logički operatori
  - i (&&)
  - ili (||)
  - ne (!)

Objekti

- Objekti predstavljaju glavni tip podatka u JavaScript-u. "Sve" u JavaScript-u predstavlja objekat.
- Objekti se sastoje od atributa(properties) i metoda(methods).
- Definicija:
  - var osoba = {ime:"Marko", prezime:"Markovic", godina:30};
  - var osoba = new Object();
  - osoba.ime = "Marko";
  - osoba.prezime = "Markovic";
  - osoba.godina = 30;
- Uklanjanje atributa:
  - delete osoba.prezime;
  - delete osoba[“godina“];

Objekti

- Ovo su neki od JavaScript objekata:
  - var x = new Date(); // Date object = current time
  - var x = new Array(); // Array object = []
  - var x = new String(); // String object = ""
  - var x = new Number(); // Number object = 0
  - var x = new Boolean(); // Boolean object = false
  - var x = new RegExp(); // Pattern matching object = /(?:)/
  - var x = new Function(); // Function object = anonymous() {}
  - var x = new Object(); // Object object = {}

If-else naredba

- **IF** naredba izvršava kod samo ako je određeni uslov ispunjen
- Definicija:

```
if (uslov) {
    naradba1
} elseif(uslov) {
    naradba3
} else {
    naradba3
}
```

Switch naredba

- Definicija:
  - switch(n)
    {
    case 1:
        naredba 1
        break;
    case 2:
    case 3:
    case 4:
        naredba 2
        break;
    default:
         naredba default
    }

For Petlja

- Petlja izvršava naredbe unutar petlje sve dok je uslov zadovoljen
- Definicija:

```
var n = 0;
for (var i = 0; i < 3; i++) {
    n += i;
    alert("Vrednost n = " + n);
}
```

`break;` - prekida izvrsenje petlje

`continue;` - preikda tekuću iteraciju

While Petlja

- Definicija:

```
var n = 0;
var x = 0;
while(n < 3) {
    n++;
    x += n;
    alert("Vrednost n = " + n + ". Vrednost x = " + x);
}
```

Funkcije

- Funkcija u JavaScript-u predstavlja proceduru, set naredbi koje izvršavaju određeni zadatak (task).
- Definicija:

```
function popupalert() {
    alert('Alert!');
}

var popupalert = function () {
    alert('Alert!');
}

popupalert();
```

- poziv funkcije

Namespace polution

- Treba obratiti pažnju na deklarisanje promenjivih unutar globalnog namespace-a
- Primer:

```
var x = 10;
function test() {
    var x = 20;
    console.log(x);
}
test();
console.log(x);
```

```
var x = 10;
function test() {
    x = 20;
    console.log(x);
}
test();
console.log(x);
```

Komentari

- Komentari se pišu na sledeći način:
  - // komentar u jednoj liniji
  - /* viselinijski komentar u jednoj liniji */
  - /* ovo je komentar
  - * u vise linija...
  - * jos jedna linija. */

DOM

- **DOM** (Document Object Model) predstavlja konvenciju za prezentaciju i interakciju sa objektima (nodovima) unutar HTML dokumenta.
- Nodovi svakog dokumenta su organizovani u strukturi stabla zvanoj “DOM Tree“
- Javascript manipulacija DOM-om podrazumeva kreiranje, dodavanje, uklanjanje, modifikaciju, stilizovanje i pristupanje elementima DOM-a.

DOM

- Kreiranje elementa
  - var element = document.createElement(tagName);
- Dohvatanje elemenata
  - a) var element = document.getElementById(id);
  - b) var elements = document.getElementsByClassName(names);
  - **>= IE8**
  - **c) var elements = document.getElementsByTagName(name);**
  - d) var element = document.querySelector(selectors);
  - e) var elements = document.querySelectorAll(selectors);

DOM

- Dodavanje elementa
  - var child = element.appendChild(child);
- Dohvatanje i izmena atributa elemenata
  - var content = element.**innerHTML**;
  - element.**innterHTML** = content;
  - var class = element.className;
  - otherElement.className = “class-name“;
  - var id = element.id;
  - otherElement.id = “id-name“;

DOM

- Stilizovanje elemenata
  - element.style.color = "#ff3300";
  - element.style.marginTop = "30px";
  - element.style.paddingBottom = "30px";
- Pristupanje relacijama elemenata
  - element.childNodes
  - element.nextSibling
  - element.children
  - element.nextElementSibling
- više o ostalim atributima i funkcijama na : http://www.w3schools.com/jsref/dom_obj_all.asp

Events

- Interakcija sa elementima DOM-a se obavlja preko događaja (**events**)
- Neki od dogadjaja su:
  - onclick
  - onsubmit
  - onmouseover
  - onmouseout
  - onkeydown
  - onfocus
  - onblur
  - ...
- više o ostalim event atributima na : http://www.w3schools.com/tags/ref_eventattributes.asp

Math

- Metode Math objekta:
  - abs(x) vraća apsolutnu vrednost x-a
  - ceil(x) Vraća x, zaokružen na najbliži veći broj
  - exp(x) Returns the value of Ex
  - floor(x) Vraća x, zaokružen na najbliži manji broj
  - log(x) logaritam
  - max(x,y,z,...,n) vraća najveći broj skupa
  - min(x,y,z,...,n) vraća najmanji broj skupa
  - pow(x,y) stepenovanje
  - random() vraća slučajan broj od 0 do 1
  - round(x) zaokružuje broj na najbliži celi broj
  - sqrt(x) korenovanje
  - sin(x), asin(x), cos(x), acos(x), tan(x), atan(x), atan2(y,x) – sinus, arkus sinus, ...
- Konstante:
  - Math.E
  - Math.PI
  - Math.SQRT2
  - Math.SQRT1_2
  - Math.LN2
  - Math.LN10
  - Math.LOG2E
  - Math.LOG10E

Date

- Date objekat se koristi za rad sa datumima i vremenima.
- Deklaracija:
  - var danas = new Date();

    var dan1 = new Date("January 15, 1988 12:00:00");

    var dan2 = new Date(76,5,21);

    var dan3 = new Date(76,5,21,10,30,0);

- Primeri:
  - dan2.setDate(dan1.getDate()+5);

    dan3.setFullYear(2014,0,15);

    if(dan3 > dan1){
        alert("Dan 3 je veci!");
    }

- Moment.js - http://momentjs.com/

Window & Screen

- Window objekat:
  - window.innerHeight – unutrasnja visina brauzerovog prozora
  - window.innerWidth - unutrasnja sirina brauzerovog prozora
  - window.open() - otvori novi prozor
  - window.close() -zatvori tekuci prozor
  - ….
- Screen objekat:
  - screen.availWidth – dostupna sirina ekrana
  - screen.availHeight - dostupna visina ekrana

Location, History & Navigator

- Location objekat:
  - location.hostname – vraca domen
  - location.pathname – vraca relativnu putanju(bez domena)
  - location.href - vraca punu URL putanju
  - location.port – vraca port, obicno je to 80 ili 443
  - location.protocol – vraca korisceni protokol (http:// ili https://)
- History objekat:
  - history.back() - korak unazad
  - history.forward() - korak unapred
- Navigator objekat:
  - navigator.userAgent – ime brauzera
  - navigator.platform – operativni sistem / platforma
  - navigator.cookieEnabled – da li su ukljuceni kolacici
  - ...

Boxes

- Alert Box:
  - alert("Tanja SEO Expert !");
- Confirm Box:
  - confirm("Potvrdite?");

```
var odg=confirm("Izaberite opciju...");
if (odg==true) {
    x="OK!";
} else {
    x="Cancel!";
}
alert("Odgovor: "+x);
```

- Prompt Box:
  - var ime= prompt("Vase ime?","Janko");

```
if (ime!=null) {
    x="Cao! Kako si " +ime+ "?";
    alert(x);
}
```

Timing

- setInterval() - izvrsava funkciju(stalno) nakon odredjenog intervala zadatog u milisekundama
- clearInterval() - zaustavlja setInterval funkciju
- setTimeout() - izvrsava funkciju (samo jednom), nakon odredjenog intervala zadatog u milisekundama
- clearTimeout() - zaustavlja setTimeout funkciju

Cookies

- Kolacici (Cookies) predstavljaju podatke koji se cuvaju u malim tekstualnim fajlovima na klijentskom racunaru.
- Definicija:
  - document.cookie="ime=Marko; expires=Thu, 15 May 2014 12:00:00 GMT; path=/";
- document.cookie - vraca sve kolacice u jednom stringu
  - cookie1=value; cookie2=value; cookie3=value;

- http://www.w3schools.com/js/js_cookies.asp
- https://github.com/ScottHamper/Cookies

RegExp

- Regularni izraz (RegExp) predstavlja sekvencu karaktera koja formira uzorak(pattern) za pretragu.
- Definicija:
  - var uzorak = /marko/i
- Uzorak pretrazuje string sa reci "marko" i ignorise mala i velika slova
- Koristi se sa string funkcijama "search" i "replace"
  - var test_string = "Moje ime je Marko Petrovic.";
  - var n = test_string.search(/marko/i);
  - var zamenjeno = test_string.replace(uzorak, "*****");

- http://regexpal.com - alat za testiranje regularnih izraza

JSON

- JSON predstavlja sintaksu za cuvanje I razmenu informacija, slicno kao kod XML-a. JSON je format koji je daleko jednostavniji od XML-a.
- **J**ava**S**cript **O**bject **N**otation
- JSON.stringify(objekat) - pretvara objekat u string
- JSON.parse(string) -

  pretvara string u objekat

```
var firma = {
"radnici": [
{ "ime":"Marko" , "prezime":"Markovic" },
{ "ime":"Janko" , "prezime":"Jankovic" },
{ "ime":"Petar" , "prezime":"Petrovic" }
]
};
```

PITANJA?