

# Programska potpora komunikacijskim sustavima

7. predavanje, 3. svibnja 2023.

doc. dr. sc. Jelena Božek

## Tehnologije weba (HTTP, HTML, CSS, JS)



- Predavanja izradio izv. prof. dr. sc. Marin Vuković, 2022.
- Predavanja doradila doc. dr. sc. Jelena Božek, 2023.

# Creative Commons



- **slobodno smijete:**

- **dijeliti** — umnožavati, distribuirati i javnosti priopćavati djelo
- **remiksirati** — prerađivati djelo



- **pod sljedećim uvjetima:**

- **imenovanje.** Morate priznati i označiti autorstvo djela na način kako je specificirao autor ili davatelj licence (ali ne način koji bi sugerirao da Vi ili Vaše korištenje njegova djela imate njegovu izravnu podršku).
- **nekomercijalno.** Ovo djelo ne smijete koristiti u komercijalne svrhe.
- **dijeli pod istim uvjetima.** Ako ovo djelo izmijenite, preoblikujete ili stvarate koristeći ga, preradu možete distribuirati samo pod licencom koja je ista ili slična ovoj.



U slučaju daljnjeg korištenja ili distribuiranja morate drugima jasno dati do znanja licencne uvjete ovog djela. Najbolji način da to učinite je linkom na ovu internetsku stranicu.

Od svakog od gornjih uvjeta moguće je odstupiti, ako dobijete dopuštenje nositelja autorskog prava. Ništa u ovoj licenci ne narušava ili ograničava autorova moralna prava.

Tekst licencije preuzet je s <http://creativecommons.org/>.

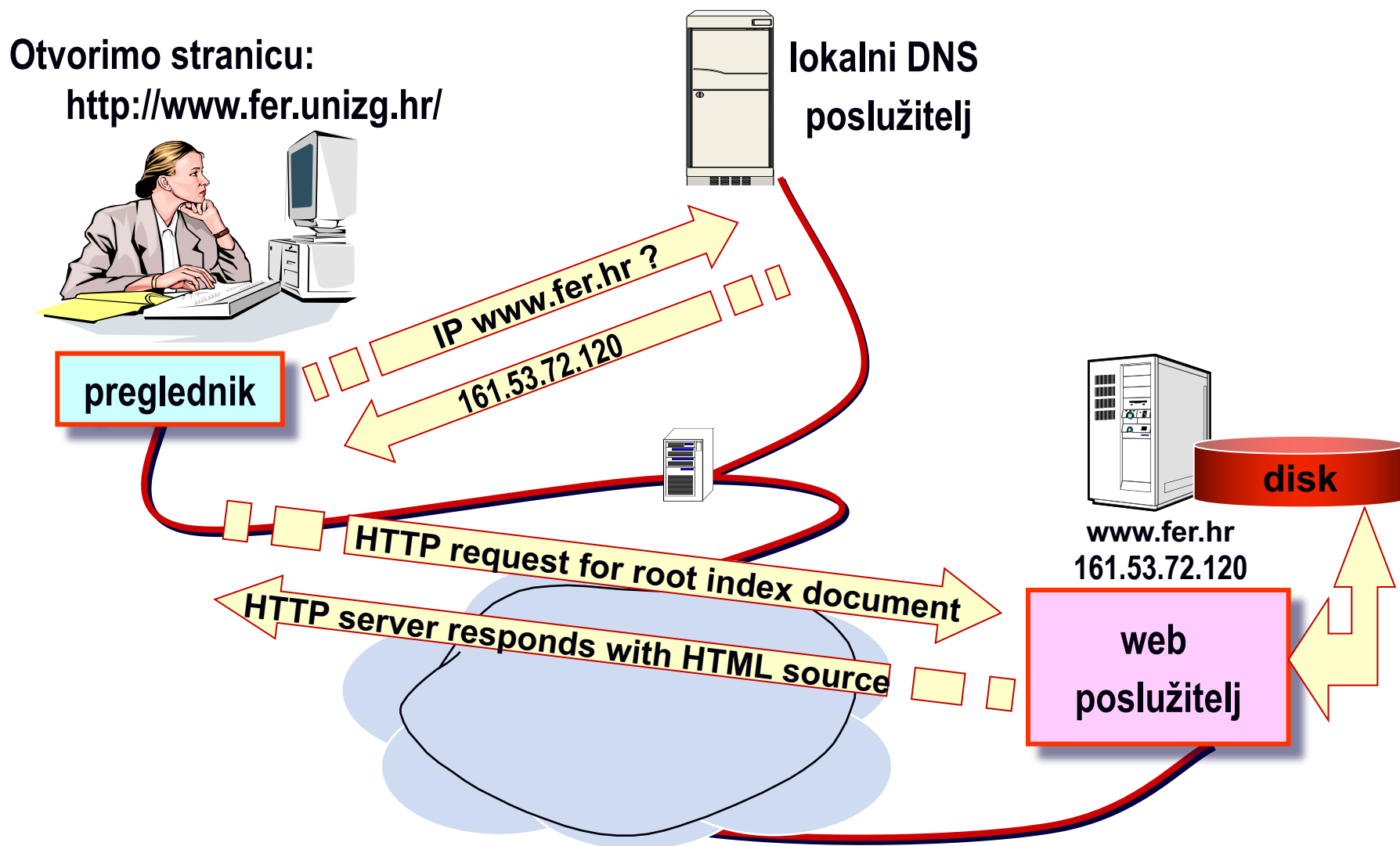
# Sadržaj predavanja

- Web
- Protokol HTTP
- HTML
- CSS
- JavaScript

# Web

- WWW – World Wide Web

# Kako radi WWW – osnovna ideja



# Primjer: Osnovna stranica FER-a – elementi

The screenshot shows the homepage of the Faculty of Electrical Engineering and Computing (FER) at the University of Zagreb. The browser address bar shows 'fer.unizg.hr'. The page features a dark blue header with the FER logo and navigation links: 'STUDIJI', 'ISTRAŽIVANJE', 'ŽIVOT@FER', 'O FAKULTETU', and 'NOVOSTI I OBJAVE'. A 'Prijava' (Login) button is highlighted with a red box and labeled 'poveznica na obrazac'. A large banner below the header reads 'STUDY @FER!' and 'Studiraj FER na engleskom jeziku!'. A red box labeled 'aktivni element - menu' points to the navigation links. A red box labeled 'slika' points to the banner image. Below the banner is an orange bar with the text 'Obavijesti vezane uz koronavirus >>>'. The main content area is titled 'Studiji' and contains four columns of study programs: 'PREDDIPLOMSKI STUDIJ', 'DIPLOMSKI STUDIJ', 'DOKTORSKI STUDIJ', and 'SPECIJALISTIČKI STUDIJ'. The first column is highlighted with a red box and labeled 'tekst'. The text in the first column reads: 'Studij je organiziran kroz dva trogodišnja studijska programa (smjera), a započinje zajedničkom prvom godinom.' The text in the second column reads: 'Diplomski studij organizira se kroz tri studijska programa, traje dvije godine, a izvode se po sustavu preduvjeta.' The text in the third column reads: 'Fakultet je nositelj doktorskog studija iz područja tehničkih znanosti, znanstvenog polja elektrotehnike i znanstvenog polja računarstva.' The text in the fourth column reads: 'Provodimo šest poslijediplomskih specijalističkih studija na kojima se stručnjake iz gospodarstva poučava novim tehnologijama.' A red box labeled '(css - stil)' points to the 'SPECIJALISTIČKI STUDIJ' column.

aktivni element - menu

poveznica na obrazac

slika

(css - stil)

tekst

# Procesiranje izvornog koda u HTML-u

- U ovom primjeru HTML sadrži:

- CSS - stil

```
<link rel="stylesheet" type="text/css" href="/_pub1683014340/themes_static/fer2016/default/style.css">
```

- slike

```

```

- JavaScript

```
<script type="text/javascript" src="/lib109/json2.js"></script>
```

- tekst

```
<p class="block_link_abstract">Studij je organiziran kroz dva ... </p>
```

- ostali elementi (hiperlinkovi, nabranja, obrasci, ...)

- Kada se procesira HTML potrebno je dohvatiti i druge elemente (slike, stil, JavaScript)

- Svi elementi se s poslužitelja dohvaćaju protokolom HTTP

- Klijent može pokrenuti novu konekciju ili koristiti postojeću za dohvaćanje ostalih elemenata



# Protokol HTTP

- Hyper Text Transfer Protocol

# Protokol Hypertext Transfer Protocol (HTTP)

- internetski protokol aplikacijskog sloja
- definira format i način razmjene poruka
  - tekstualan zapis, sličan formatu e-mail poruke i standarda MIME
- vrste poruka:
  - zahtjev ("metoda")
    - definira operaciju (metodu), resurs, protokol
    - naziv "metoda" potječe iz područja objektno-orijentiranog programiranja
  - odgovor (rezultat)
    - rezultat (uspjeh, neuspjeh, greška,...) opisan statusnim kôdom
    - neke vrste odgovora u tijelu imaju sadržaj zatraženog resursa

# Primjer HTTP-zahatjeva

```
1. GET /predmet/rassus HTTP/1.1
2. Host: www.fer.unizg.hr
3. User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.6; rv:7.0.1) Gecko/20100101
  Firefox/7.0.1
4. Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5. Accept-Language: hr,en;q=0.7,en-us;q=0.3
6. Accept-Encoding: gzip, deflate
7. Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
8. Connection: keep-alive
9. Cache-Control: max-age=0
```

**početni redak**

**polja zaglavlja:**

opća zaglavlja

zaglavlja karakteristična za zahtjev ili

odgovor

zaglavlja karakteristična za entitet

**prazan redak**

**tijelo  
poruke**

# Primjer HTTP-zahatjeva

```
1. GET /predmet/rassus HTTP/1.1
2. Host: www.fer.unizg.hr
3. User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.6; rv:7.0.1) Gecko/20100101
   Firefox/7.0.1
4. Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5. Accept-Language: hr,en;q=0.7,en-us;q=0.3
6. Accept-Encoding: gzip, deflate
7. Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
8. Connection: keep-alive
9. Cache-Control: max-age=0
```

Obavezni dijelovi

# Metode zahtjeva

- metoda zahtjeva određuje što se traži od resursa
- HTTP/1.1 definira 8 metoda i omogućuje dodavanje novih metoda (extensions):
  - OPTIONS - informiranje i mogućnostima resursa i poslužitelja
  - GET - za dohvaćanje resursa (najčešća)
  - HEAD - za dohvaćanje podataka o resursu (npr. veličina, postojanje)
  - POST - aktiviranje resursa (npr. slanje podataka obrazaca)
  - PUT - postavljanje entiteta (npr. promjena podataka - kod REST-a)
  - DELETE - brisanje resursa (npr. kod REST-a)
  - TRACE - za dijagnostiku
  - CONNECT - za buduću uporabu (ne implementira se)

# Primjer HTTP-odgovora

1	<code>.HTTP/1.1 200 OK</code>	početni redak
2	<code>.Date: Wed, 12 Oct 2011 08:19:32 GMT</code>	
3	<code>.Server: Apache/2.2.20 (FreeBSD) mod_ssl/2.2.20 OpenSSL/0.9.8q mod_fcgid/2.3.6</code>	
4	<code>.Expires: Thu, 19 Nov 1981 08:52:00 GMT</code>	
5	<code>.Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0</code>	
6	<code>.Pragma: no-cache</code>	
7	<code>.P3P: CP="NOI CURa ADMA DEVa TAIa PSAa PSDa IVAa IVDa HISa OTPa OUR BUS IND UNI COM NAV INT"</code>	
8	<code>.Set-Cookie: CMS=2p72ge55hqm; expires=Wed, 19-Oct-2011 20:19:32 GMT; path=/;</code>	
9	<code>.Vary: Accept-Encoding</code>	
10	<code>.Transfer-Encoding: chunked</code>	
11	<code>.Content-Type: text/html; charset=utf-8</code>	
12	<code>.d9e7</code>	
13	<code>.&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"</code>	
14	<code>"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"&gt;</code>	
15	<code>&lt;html xmlns="http://www.w3.org/1999/xhtml"&gt;</code>	
16	<code>.</code>	prazan redak
17	<code>.&lt;head&gt;</code>	
18	<code>.&lt;meta http-equiv="Content-Type" content="text/html; charset=utf-8" /&gt;</code>	
19	<code>. ...</code>	
		tijelo poruke

# HTTP-odgovor

- početni redak (protokol, statusni kod i opis)
- kategorije statusnih kodova:
  - **1xx** – Informativne - ni uspjeh, ni neuspjeh
  - **2xx** – Uspjeh - poslužitelj je primio, razumio i ispunio zahtjev
  - **3xx** – Preusmjerenje - potrebno poduzeti dodatne akcije
  - **4xx** – Greška na klijentu - zahtjev je neispravan
  - **5xx** – Greška na poslužitelju - zahtjev je ispravan, ali ga poslužitelj ne može ispuniti
- u tijelu odgovora se obično prenosi reprezentacija resursa (“entitet”) koju preglednik treba prikazati korisniku (npr. HTML)
- primjer nekih polja zaglavlja:
  - **Content-Type:** format entiteta
  - **Content-Length:** duljina entiteta u tijelu (u oktetima)

# HTML

- Hyper Text Markup Language



# World Wide Web Consortium (W3C)

- [www.w3.org](http://www.w3.org)
- Aktivnosti organizirane u [radne grupe](#)
- HTML i CSS
  - <https://www.w3.org/standards/webdesign/htmlcss>
- Edukacija: <http://www.w3schools.com>

# Povijest HTML-a

- 1991. nacrt u CERN-u
- 1993. javna pojava HTML-a
- 1995. HTML 2
  - RFC 1866 + dodatni RFC-ovi
- 1996. HTML 3
  - IETF odustaje od standardizacije HTML-a
- 1997. HTML 3.2
  - izdaje ga W3C
- 1998. HTML 4
  - Podrška za video sadržaj (Flash)
- 2000. XHTML 1.0 ekvivalent HTML 4.01 Strict
- 2004. počinje standardizacija HTML 5
  - 2008. prvi javni nacrt
  - 2012. radni nacrt
  - [28.10.2014. postao standard](#)
- 2014.-sada – radi se na poboljšanju
  - [5.2. standard](#) 14.12.2017.
  - [Living Standard](#) – kontinuirano

# Primjer HTML-dokumenta

- `<!DOCTYPE html>` ← deklaracija
- `<html>` ← korijenski element
- `<head>` ← zaglavlje s metapodacima
- `<title>Naslov</title>` ← naslov
- `</head>`
- `<body>` ← tijelo – vidljivi elementi
- `<h1>Naslov</h1>` veliki naslov
- `<p>Neki tekst.</p>` odlomak teksta
- `</body>`
- `</html>`

# Oznake

`<oznaka>` - otvarajuća

`</oznaka>` - zatvarajuća

`<oznaka/>` - otvarajuća i zatvarajuća

- U HTML-u možemo imati i samo otvarajuću oznaku bez zatvarajuće
  - Npr. `<br>`
- Atributi:
  - Npr. `<a href="poveznica"> ...`

# Važni atributi

- `id` – jedinstveni identifikator oznake
- `name` – ime oznake, ali ne treba biti jedinstveno
- `class` – jedna ili više klasa oznake (referira se na klasu u stilu)
- `style` – opisuje CSS stil u HTML dokumentu

# Popis bitnih oznaka

- h1, h2, h3, ... h6 – naslovi
- p – odlomak
- Oznake za formatiranje:
  - b – podebljano
  - strong – važni tekst
  - i – koso
  - em – naglašeno
- Komentar <!-- -->
- Poveznice: <a href="url">tekst</a>
- Slika:
  - 
- div, span – služe za označavanje dijela dokumenta
- br, hr – prelazak u novi red i horizontalna linija

# Tablica

```
<table style="width:100%">
  <tr>
    <th>Ime</th>
    <th>Prezime</th>
    <th>Broj telefona</th>
  </tr>
  <tr>
    <td>Pero</td>
    <td>Perić</td>
    <td>203</td>
  </tr>
  ...
</table>
```

Diagram illustrating the structure of an HTML table with annotations:

- tablica**: Points to the opening `<table>` tag.
- red**: Points to the opening `<tr>` tag.
- element zaglavlja**: Points to the opening `<th>` tag.
- element u redu**: Points to the opening `<td>` tag.

# Liste

Neporedana lista

<ul>

<li>Meso</li>

<li>Luk</li>

<li>Salata</li>

</ul>

- Meso
- Luk
- Salata

Poredana lista

<ol>

<li>Meso</li>

<li>Luk</li>

<li>Salata</li>

</ol>

1. Meso
2. Luk
3. Salata

Opisna lista

<dl>

<dt>Meso</dt>

<dd>- teletina</dd>

<dt>Luk</dt>

<dd>- crveni</dd>

</dl>

Meso  
- teletina  
Luk  
- crveni



# CSS

- Cascading Style Sheets

# CSS

- Opisuje kako će biti prikazani elementi iz HTML-a na ekranu, papiru ili nekom drugom mediju
- Sintaksa:

```
selektor {  
    svojstvo1:vrijednost1;  
    svojstvo2:vrijednost2;  
    ...  
}
```

# Selektori

- #id – odabire oznaku s identifikatorom id
- .klasa – odabire oznaku po klasi
- oznaka – odabire elemente po oznaci
- Npr. p.klasa – odabire oznake p koje imaju atribut klasa
- Grupiranje – odabire neki od selektora
  - Npr. h1, h2 – odabire oznake h1 i h2
  - Npr. h1 h2 – odabire h2 koji je unutar h1

# Uključivanje stilova u dokument

- Eksterni stilovi
  - `<head>`
  - `<link rel="stylesheet" type="text/css" href="neki.css">`
  - `</head>`
- Interni stilovi
  - `<head>`
  - `<style>`
  - ...
  - `</style>`
  - `</head>`
- Stilovi u oznaci
  - `<h1 style="color:blue">Naslov</h1>`

# Primjer (HTML): primjer.html

```
<!DOCTYPE html>
<html>
<head>
<title>Primjer</title>
<link rel="stylesheet" href="primjer.css">
</head>
<body>
<h1>Tekst u boji</h1>
<p>Sljedeći elementi imaju tekst u boji:</p>

<div class="text-red">
  <h2>Crveno</h2>
  <p>Tekst u crvenoj boji.</p>
</div>
<div class="text-blue">
  <h2>Plavo</h2>
  <p>Ovaj tekst je u plavoj boji.</p>
</div>
</body>
</html>
```

# Primjer (CSS): primjer.css

```
.text-red {  
  color:#f44336  
}
```

```
.text-blue {  
  color:#2196F3  
}
```

## Tekst u boji

Sljedeći elementi imaju tekst u boji:

**Crveno**

Tekst u crvenoj boji.

**Plavo**

Ovaj tekst je u plavoj boji.

# JavaScript

# Skripte na klijentu

- uključene u HTML ili u posebnoj datoteci koja je povezana
- obično se koristi:
  - JavaScript (Netscape), JScript (Microsoft), ECMAScript
- ECMAScript – standardiziran
  - specifikacije ECMA-262 i ISO/IEC 16262
  - svojstva: dinamički, slabo povezan, objektni, funkcijski
- nema veze s Javom
- svrha:
  - dinamički elementi
  - interakcija s korisnikom
  - provjera obrazaca
  - komunikacija s poslužiteljem (AJAX)
  - ...
- tutorial: <http://www.w3schools.com/JS/>

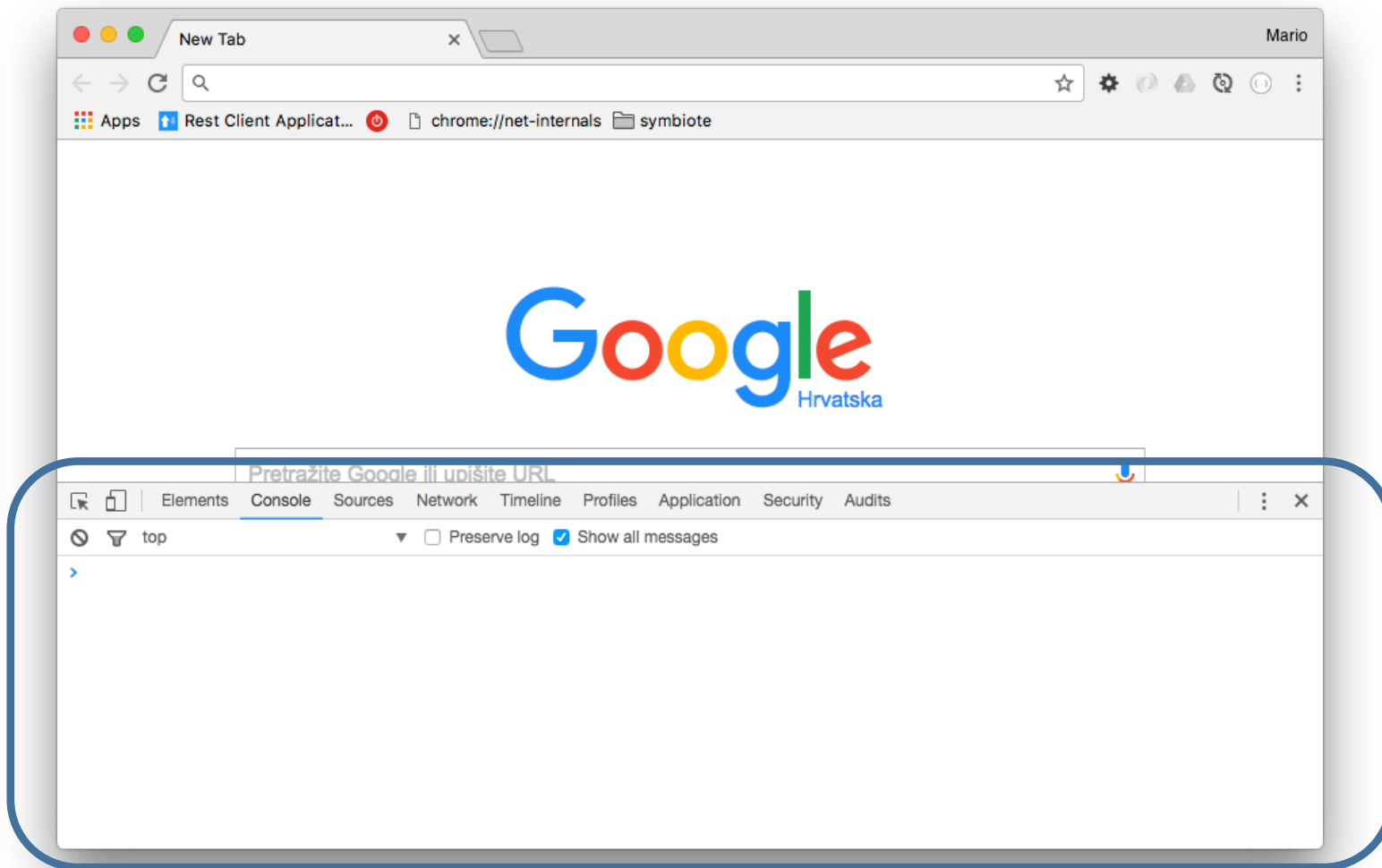


# JavaScript (ECMAScript)

- Poboljšanja u preglednicima
  - Alati za debugiranje
  - `console.log()` – ispis u posebnu konzolu
  - Objekt `windows.JSON`
    - Metoda `parse` – korišće ugrađenog parsera koji je puno brži
    - Metoda `stringify` – pretvaranje objekta u string
  - Jedinstveno rukovanje događajima u svim preglednicima
    - `addEventListener()` – dodavanje funkcije za obradu događaja
    - `dispatchEvent()` – kreiranje novog događaja
- Verzije ([podrška](#)):
  - ECMAScript 5 – podržavaju je većina preglednika
  - ECMAScript 6 – 2015. – to bi obično željeli koristiti
  - ECMAScript 7 – 2016.
  - ECMAScript 8 – 2017.
  - ECMAScript 9 – 2018.
  - ECMAScript 10 – 2019.

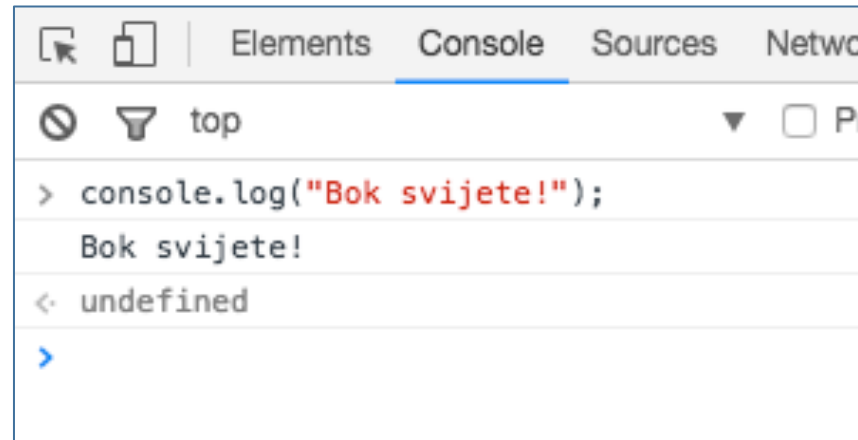
# JavaScript u Chromeu

- View → Developer → JavaScript Console



# Rad u konzoli preglednika

- `console.log("Bok svijete!");`

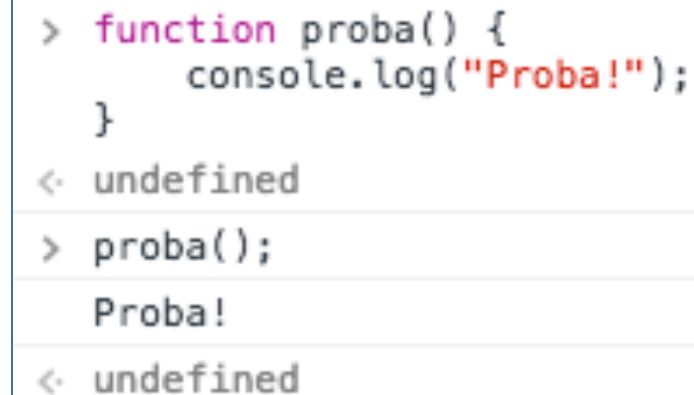


## // deklariranje globalne funkcije

```
function proba() {  
    console.log("Proba!");  
}
```

## // poziv funkcije

```
proba();
```

A screenshot of a web browser's developer console. It shows two log entries. The first entry is the function declaration: '> function proba() { console.log("Proba!"); }' in red, followed by the return value '< undefined'. The second entry is the function call: '> proba();' in blue, followed by the output 'Proba!' and the return value '< undefined'.

# Variable

```
var a = 5; // deklariraju se pomoću ključne riječi var
```

```
var b = a * 2;
```

```
console.log(b);
```

```
b = a * 3;
```

```
console.log(b);
```

```
// funkcije mogu biti spremljene u variable
```

```
var f = function(x) {
```

```
  return x + 10;
```

```
}
```

```
console.log(f(3));
```

# Tipovi podataka

```
var n = 5; // brojevi
```

```
n = 5.6;
```

```
var s = "Neki tekst!"; // string
```

```
var o = {firstName:"Ivan", lastName:"Horvat"}; // objekt
```

```
o.firstName;
```

```
var b = true; // boolean
```

```
b = false;
```

```
// polje koje je zapravo objekt
```

```
var a = [1, "Auto", {owner: o}];
```

```
a[0];
```

```
a.length;
```

```
// operator typeof
```

```
typeof a;
```

# JavaScript u HTML-u

- Dodamo oznaku script u head ili body

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
document.write("Ovo je napisano iz prvog JavaScripta!");
```

```
</script>
```

```
</body>
```

```
</html>
```

- Uključivanje koda iz datoteke

```
<script type="text/javascript" src="skripta.js"></script>
```

- U HTML-u 5 se type može izostaviti (podrazumijevana vrijednost)

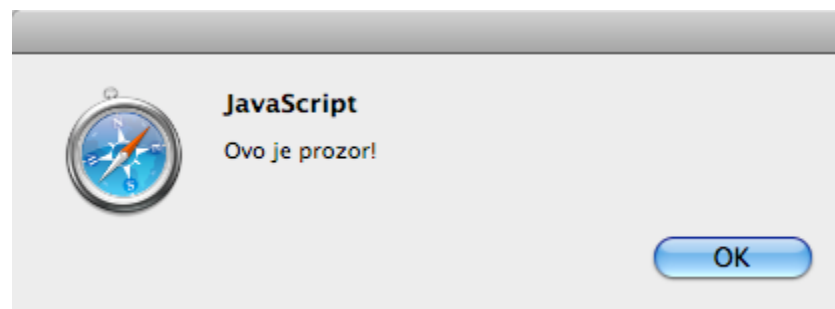


# Prozori

```
<html>
<head>
<meta charset="UTF-8" />
<script>
function disp_alert() {
    alert("Ovo je prozor!");
}
</script>
</head>
<body>

<input type="button" onclick="disp_alert()"
value="Prikaži prozor" />

</body>
</html>
```



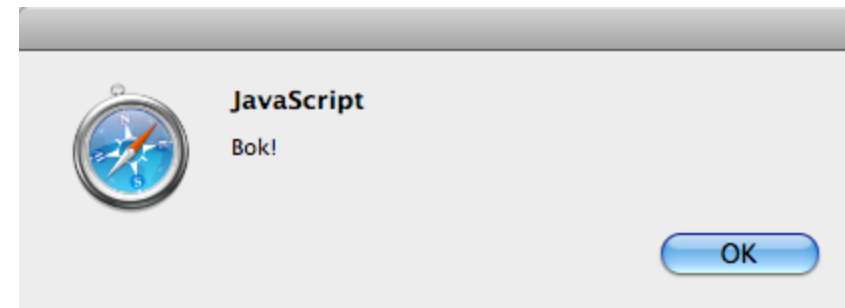
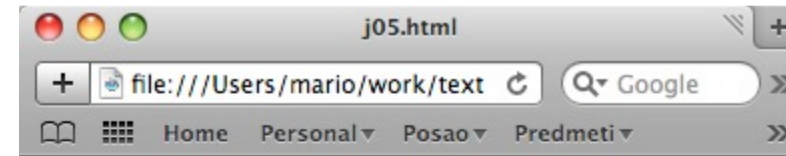
```
confirm("pitanje"); // vraća true/false

// vraća upisani odgovor
prompt("pitanje", "ponuđeni odgovor");
```

# Funkcije s parametrima i povezivanje s gumbom

```
<html>
<head>
<meta charset="UTF-8" />
<script>
function myfunction(txt) {
    alert(txt);
}
</script></head>
<body>

<form>
<input type="button"
onclick="myfunction('Bok!)" value="Pozovi funkciju">
</form>
</body>
</html>
```





# Pojednostavljen API za odabir elemenata u HTML-u 5

- `getElementById()` – atribut `id`
- `getElementsByName()` – atribut `name`
- `getElementsByTagName()` – ime elementa
- `querySelector()` – prvi element s određenim CSS-om
- `querySelectorAll()` – svi elementi s određenim CSS-om

# Upravljanje elementima

```
<html><head><title>Neki naslov</title></head>
<body><script>
document.write("Naslov:" + document.title + "<br>");

function addToDiv() {
    x = document.getElementById("mojId");
    x.innerHTML = x.innerHTML + "*";
}

function addToSpan() {
    x = document.getElementsByName("imeOznake");
    x[0].innerHTML = x[0].innerHTML + "$";
}
</script>
<div id="mojId" onclick="addToDiv()">div: </div><br>
<span name="imeOznake" onclick="addToSpan()">span: </span>
</body>
</html>
```



# Javascript u HTML-u 5

- Poboljšanja u preglednicima
  - Alati za debugiranje
  - `console.log()` – ispis u posebnu konzolu
  - Objekt `windows.JSON`
    - Metoda `parse` – korištenje ugrađenog parsera koji je puno brži
    - Metoda `stringify` – pretvaranje objekta u string
  - Jedinstveno baratanje događajima u svim preglednicima
    - `addEventListener()` – dodavanje funkcije za obradu događaja
    - `dispatchEvent()` – kreiranje novog događaja

# Poznate knjižnice u JavaScriptu

- ◆ popis se stalno mijenja i stalno raste
  - <https://www.javascripting.com>
- ◆ DOM manipulation -  
<https://www.javascripting.com/dom/>
- ◆ GUI - radni okvir -  
<https://www.javascripting.com/user-interface/>

# AJAX (Asynchronous JavaScript and XML)

