



Osnove web programiranja

HTTP

Termin 1

Sadržaj

1. WWW
2. HTTP uvod
3. Postman i HTTP komunikacija
4. Google Chrome i HTTP komunikacija
5. HTTP zahtev
6. HTTP odgovor
7. Slanje podataka iz formi
8. Isporuka www sadržaja

Dodatno:

1. URL
2. Port

Sadržaj www sajta

- HTML stranice
- multimedijalni elementi (slike, animacije, itd)
- drugi tipovi datoteka
- Prostor u kome se nalaze dokumenti kategorisani po *Uniform Resource Locator* (URL)
- www server i klijent komuniciraju preko HTTP protokola

URL

Uniform Resource Locator (URL)

- standard definisan u IETF RFC 1738 predstavlja podatke koji se mogu upotrebiti za dobavljanje resursa
- URL format - protokol:putanja-do-resursa

Protokol://Adresa_računara:port/Dinamički ili statički resurs

Primeri

<http://mojSajt.com:80/pictures> - Simbolička adresa i dinamički adresa
NAPOMENA 80 je podrazumevani port za http protokol

<http://192.168.0.1:4203/help.html> – Numerička adresa i statički resurs

Primer

The diagram illustrates the components of the URL `http://www.domain.com:1234/path/to/resource?a=b&x=y`. Red horizontal bars are placed under each component, with a vertical line connecting the label to the bar.

- protocol**: `http`
- host**: `www.domain.com`
- port**: `1234`
- resource path**: `/path/to/resource`
- query**: `?a=b&x=y`

Port

Port

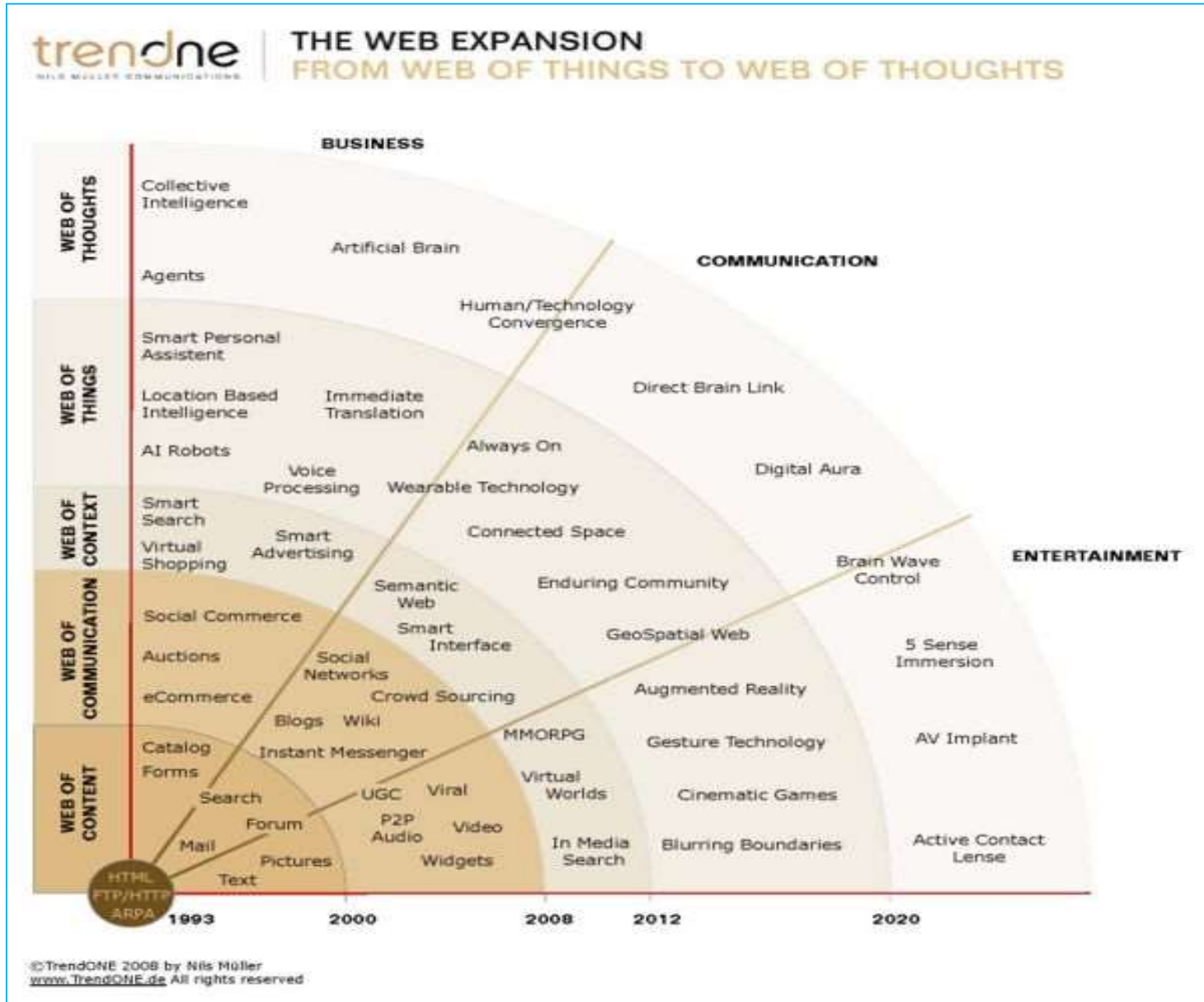
- U računarskoj mreži *port* je softverski zadat kanal kojim komuniciraju aplikacije putem računarskih mreža.
- Predstavlja broj u opsegu 0-63535. Neki od ovih brojeva su predifinisani (0-1023) dok ostale portove mogu da koriste korisničke aplikacije.
- Portom razlikujemo aplikacije na računarima u domenu mrežne komunikacije.

World Wide Web

- Web 1.0 – Korisnici su puki “čitaoci” informacija (wikipedia). Statički sadržaj uglavnom. Server traži resurs u svom fajl sistemu
- Web 2.0 (1999) – Dinamički sadržaj. Koncept Web as a Platform. Nema više desktop aplikacija. Korisnici kreiraju sadržaj. Personalizovani korisnici (user account) WebShops, Social Networks. Rich User Experience, Cloud Computing, Software as a Service (SaaS).
- Web 3.0 (2006) - web of content where the meaning can be processed by machines
- Web 4.0, 5.0 – MobileOpen Linked, Intelligent, VR, AR

WWW

World Wide Web



HTTP uvod

Uvod

- HTTP je sraćenica od *HyperText Transfer Protocol*.
- HTTP je fundamentalni protokol na koji se bazira rad WWW

Protokol definiše :

- kako se formatiraju i prenose poruke WWW
- koje akcije bi trebalo da preduzmu veb serveri i pregledači kao odgovor na različite komande
- Verzije:
 - HTTP/1.0
 - HTTP/1.1 (permanent/persistent/**keep-alive** connection) -1997
 - HTTP/2.0 (performance improvements, header compression, usage of encryption, and prioritization of requests)- 2014

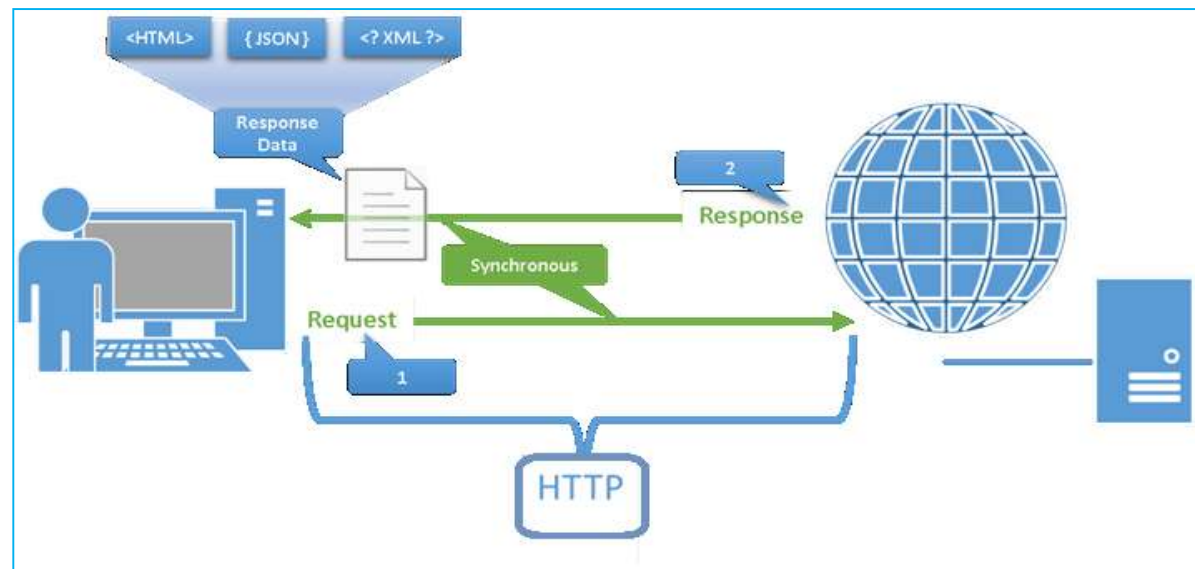
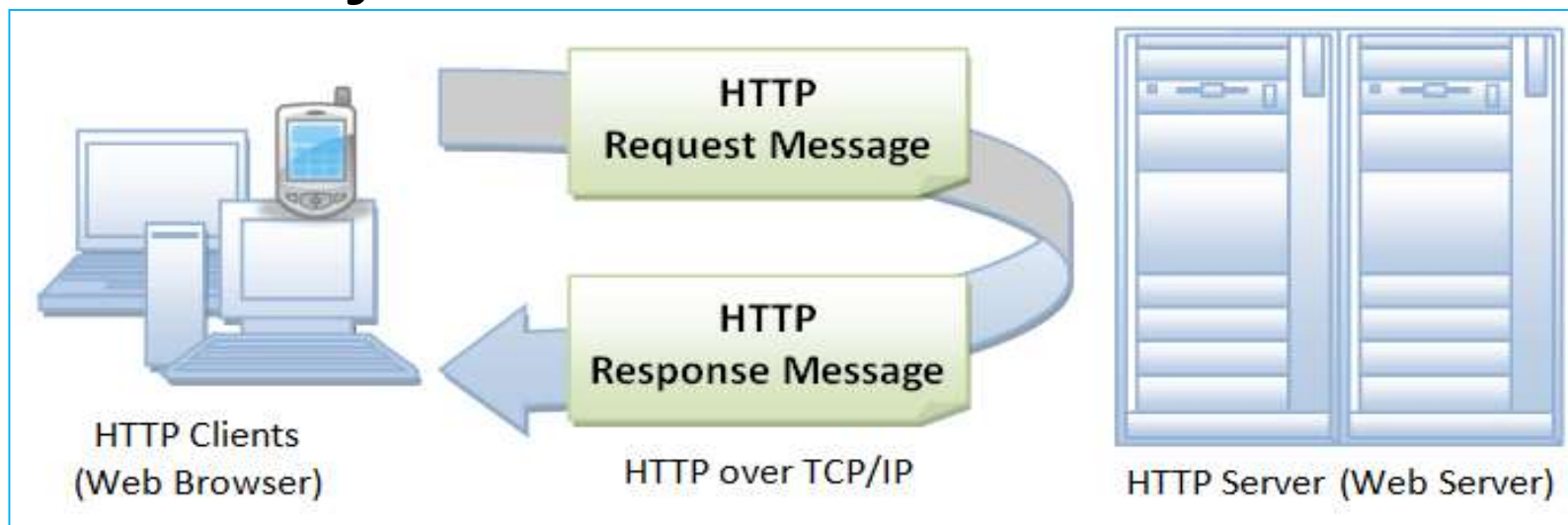
HTTP uvod

HTTP komunikacija

- zasnovana na zahtev/odgovor principu
- svaki par zahtev/odgovor se smatra nezavisnim od ostalih
- ne omogućava praćenje korisničke sesije, tj. niza zahteva upućenih od strane istog klijenta

HTTP uvod

HTTP komunikacija



HTTP uvod

Korisnička sesija

HTTP je stateless protokol koji ne zateva od servera čuvanje statusa klijenta ili korisničke sesije klijenta tj. niza zahteva upućenih od strane istog klijenta.

HTTP serveri prevazilaze prethodno tako što implementiraju različite metode za održavanje i upravljanje sesijom, tipično se oslanjajući na jedinstveni identifikator *cookie* ili neki drugi parametar koji omogućava praćenje zahteva koji originiraju od istog klijenta (npr. URL Rewriting mehanizam), kreirajući stateful protokol iznad HTTP protokola.

HTTP uvod

HTTP verzije

- U verziji 1.0 po završetku isporuke odgovora klijentu konekcija se zatvara (za novu komunikaciju klijenta sa serverom opet treba da se uspostavi konekcija).
- U verziji 1.1 konekcija se ne zatvara tj. konekcija ostaje otvorena (keep-alive). Klijent će istu konekciju da koristi pri slanju novog zahteva ka serveru. Konekcija ostaje otvorena sve dok neko od stana u komunikaciji (klijent ili server) ne odluči da je neophodno da završi komunikaciju sa drugom stranom, što će uraditi tako što će zatvoriti konekciju.

HTTP uvod

HTTP verzije 1.1 prednosti i mane

Prednosti ver 1.1 :

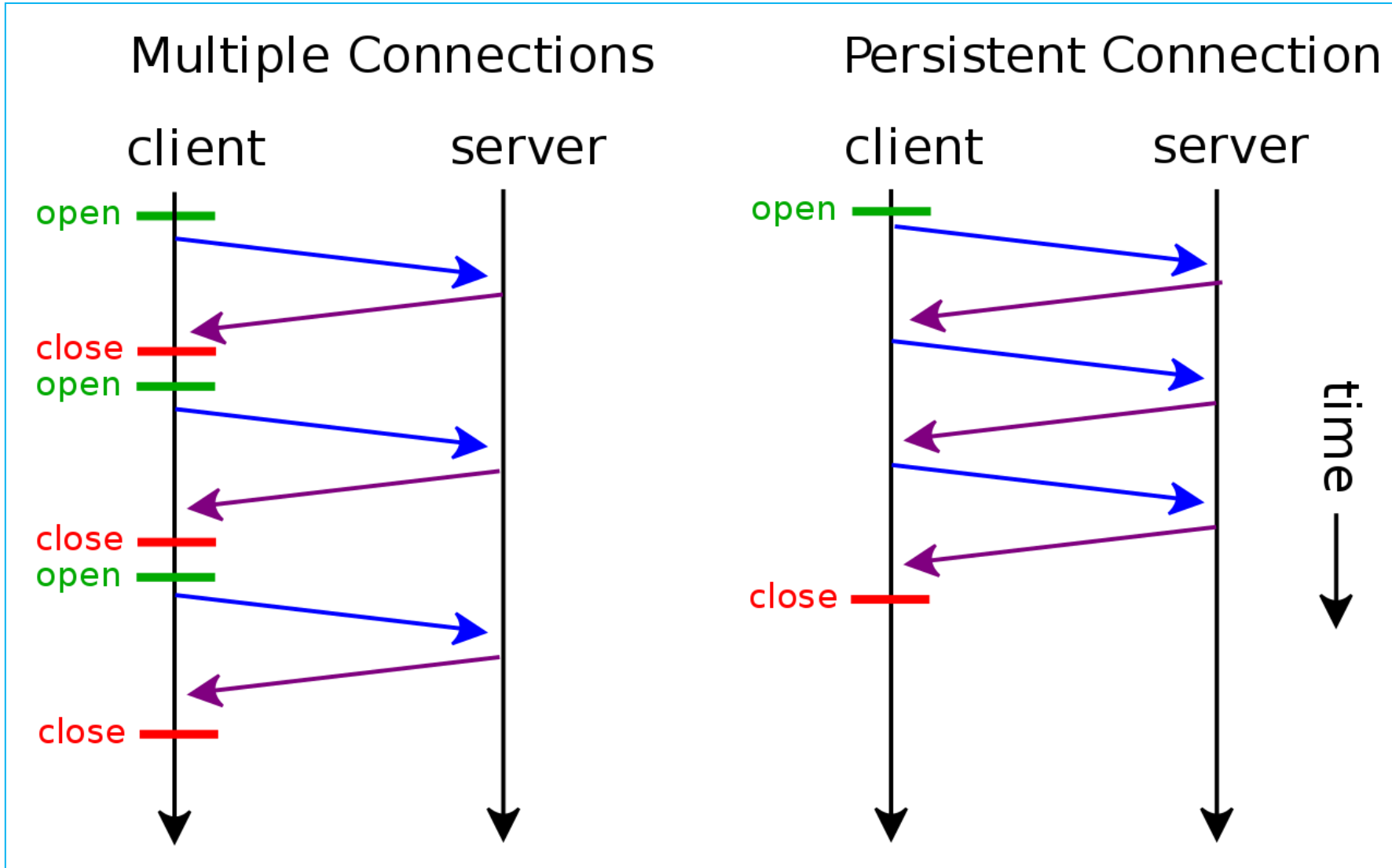
- Smanjeno zauzeće CPU jer je smanjen broj poruka koje se kreiraju, obrađuju i šalju mrežom.
- Smanjeno zagušene mreže (manje poruka za kreiranja TCP konekcija).

Mana ver 1.1 :

- Situacija u kojoj je klijent preuzeo sve podatke od servera ali nije zatvorio konekciju je problem. U takvoj situaciji server nepotrebno troši resurse za otvorenu vezu, umesto da te resurse mogu da koriste drugi klijenti.
- Prethodno može da utiče na dostupnost servera da prima nove zahteve klijenta, ako je na serveru ograničen broj klijenta koje istovremeno server opslužuje.
- Server će izvršiti zatvaranje konekcije koja je idle u zavisnosti od konfiguracije.

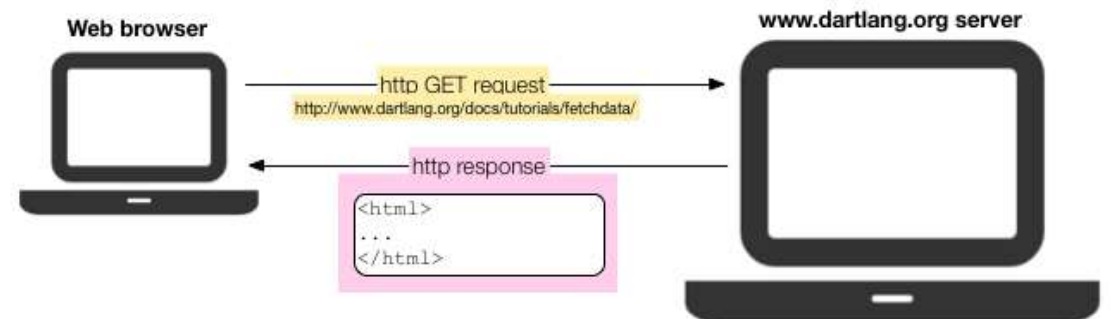
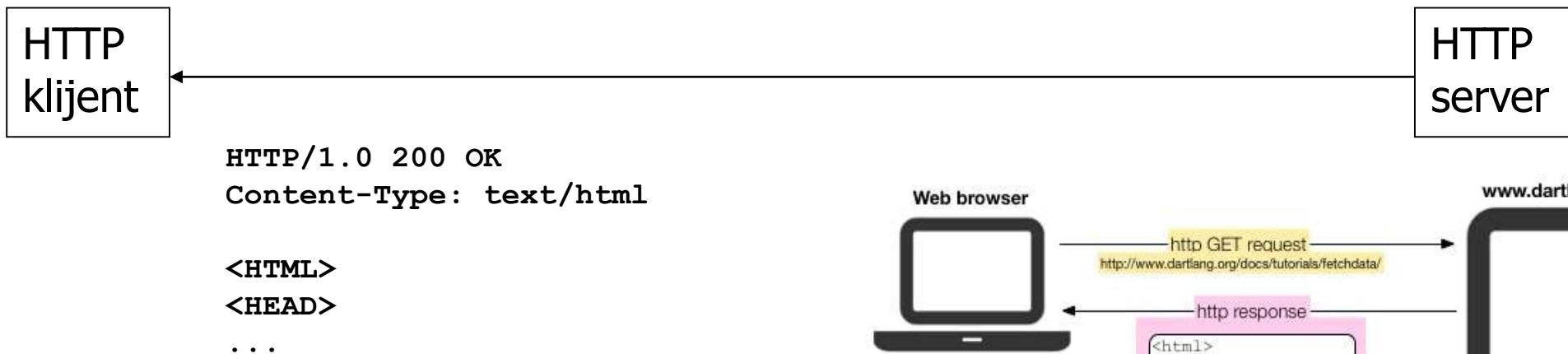
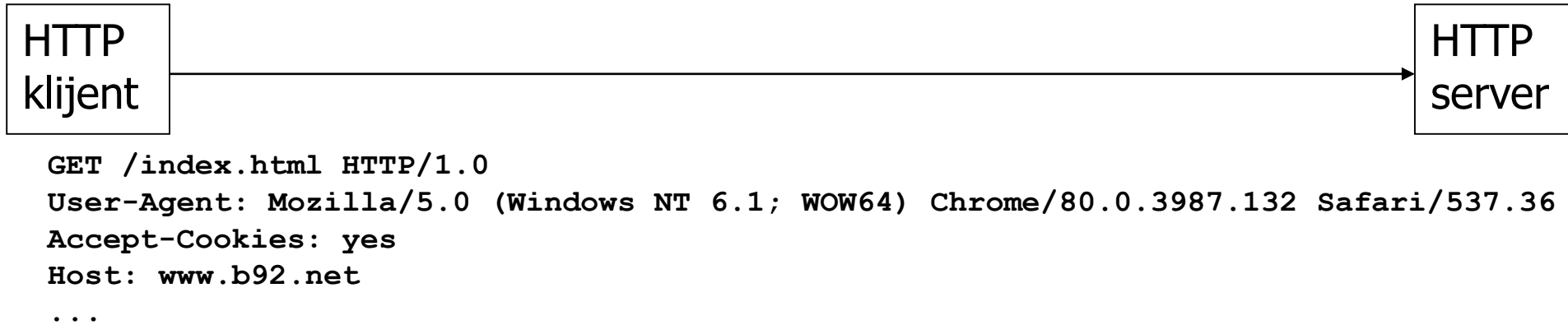
HTTP uvod

HTTP 1.0 i HTTP 1.1 komunikacija



HTTP uvod

HTTP razmena poruka



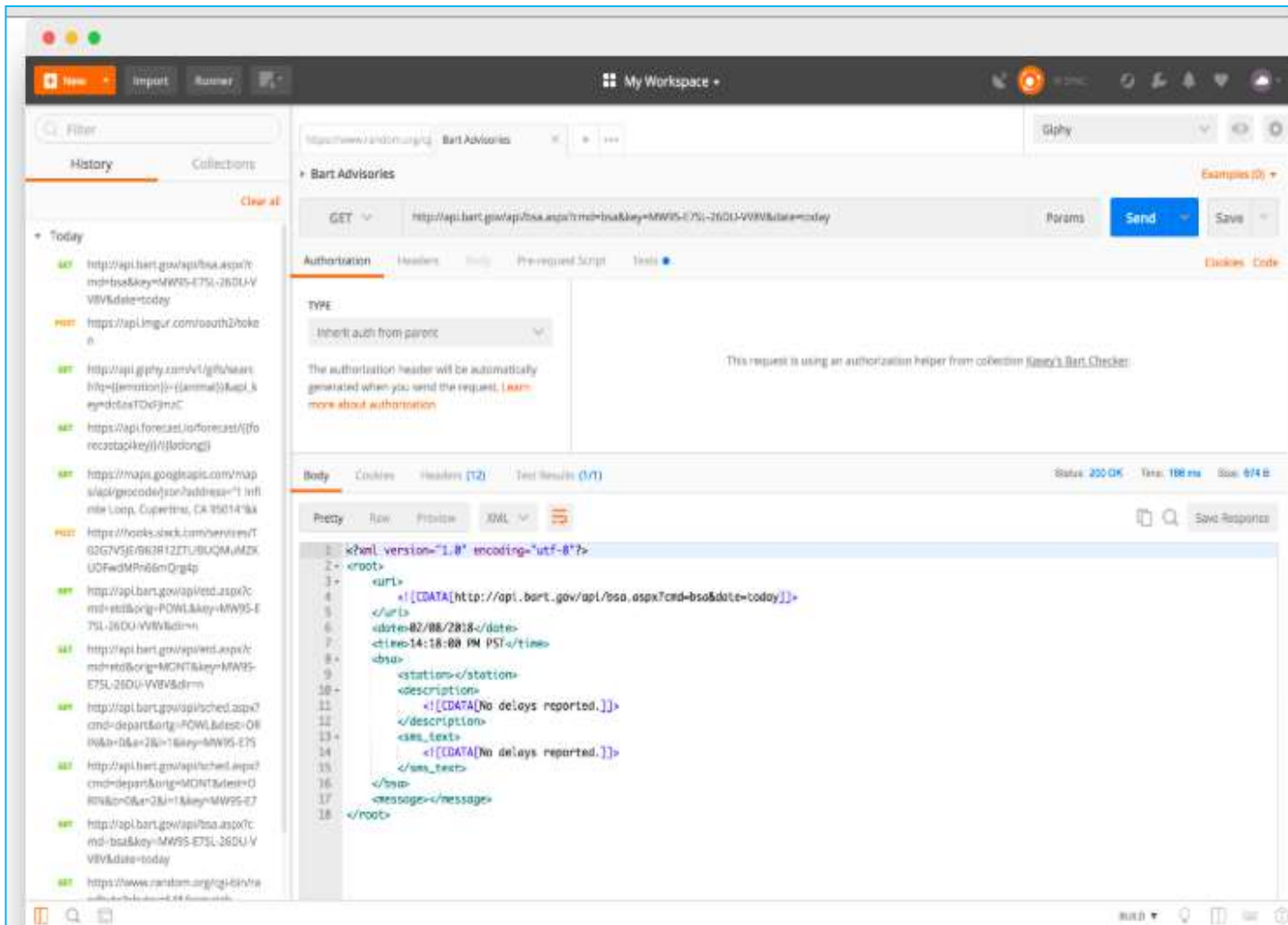
Chrome browser i Postman

<https://www.b92.net/index.html>

<https://www.youtube.com/watch?v=QV2iYFI5eSk>

Postman i HTTP komunikacija

Postman



Postman is the most complete toolchain for API development

- The most-used REST client worldwide
- Designed from the ground up to support the API developer
- Intuitive user interface to send requests, save responses, add tests, and create workflows

[Read the docs](#)

Postman i HTTP komunikacija

Postman

✓ Define complex requests

Send any type of request in Postman. Create and save custom methods and send requests with the following body types:

- URL-encoded—The default content type for sending simple text data
- multipart/form-data—For sending large quantities of binary data or text containing non-ASCII characters
- Raw body editing—For sending data without any encoding
- Binary data—For sending image, audio, video, or text files

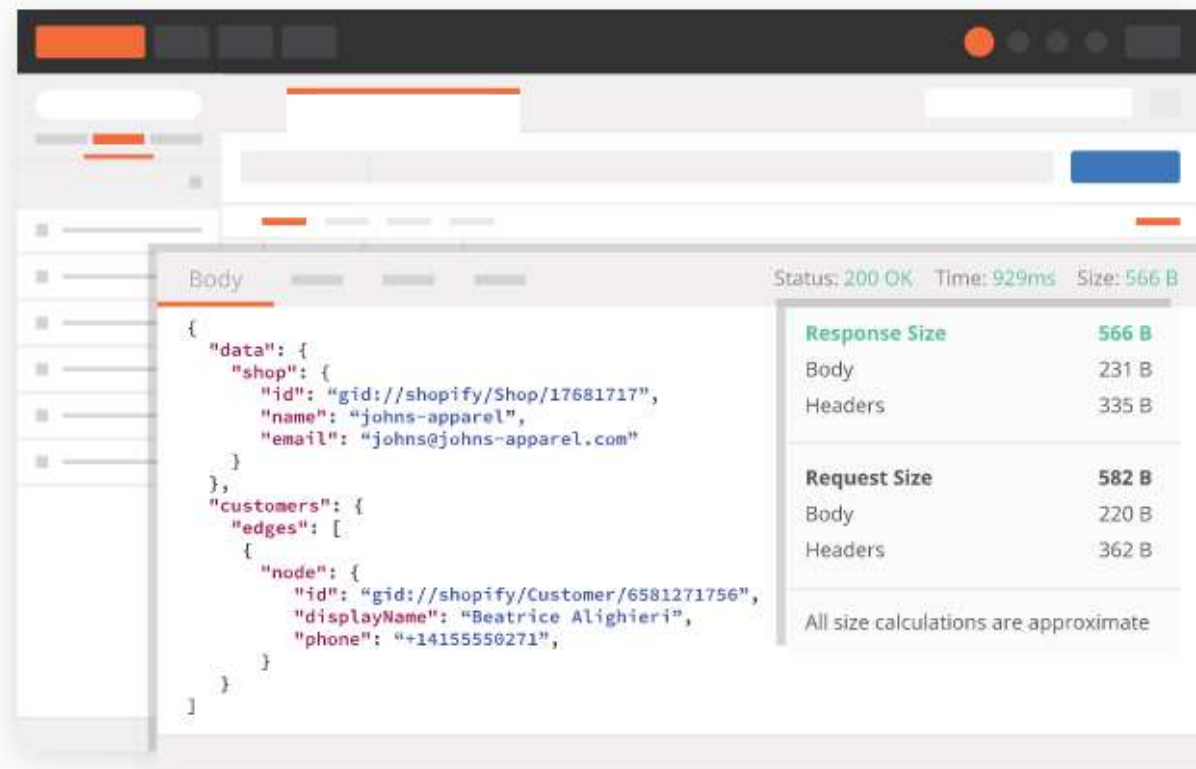
✓ Get up and running in seconds

Instead of creating calls manually to send over the command line, all you need is a Postman Collection. Import a collection directly or generate one with one click from:

- An API schema in the RAML, WADL, OpenAPI, or GraphQL format
- A data file containing the cURL commands

✓ Inspect Responses

View the status code, response time, and response size. Postman's automatic language detection, link and syntax highlighting, search, and text formatting make it easy to inspect the response body.



Postman i HTTP komunikacija

Generisanje HTTP zahteva

GET

https://www.b92.net/index.html

Send

Save

ParamsAuthHeaders (7)BodyPre-req.TestsSettingsCookiesCode

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Postman i HTTP komunikacija

Generisanje HTTP zahteva

Odabir metode

URL

Atributi zaglavlja

Parametri forme u telu HTTP zahteva

Parametri forme u URL HTTP zahteva

Pregled kolačića

Pregled poslatog zahteva

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> mojParametarForme	AAA	
Key	Value	Description

Postman i HTTP komunikacija

Generisanje HTTP zahteva

Generated code for HTTP

[Contribute on GitHub](#)

```
1 GET /index.html?mojParametarForme=AAA HTTP/1.1
2 Host: www.b92.net
3 mojAtributZaglavlja: BBB
4
```

Generated code for HTTP

[Contribute on GitHub](#)



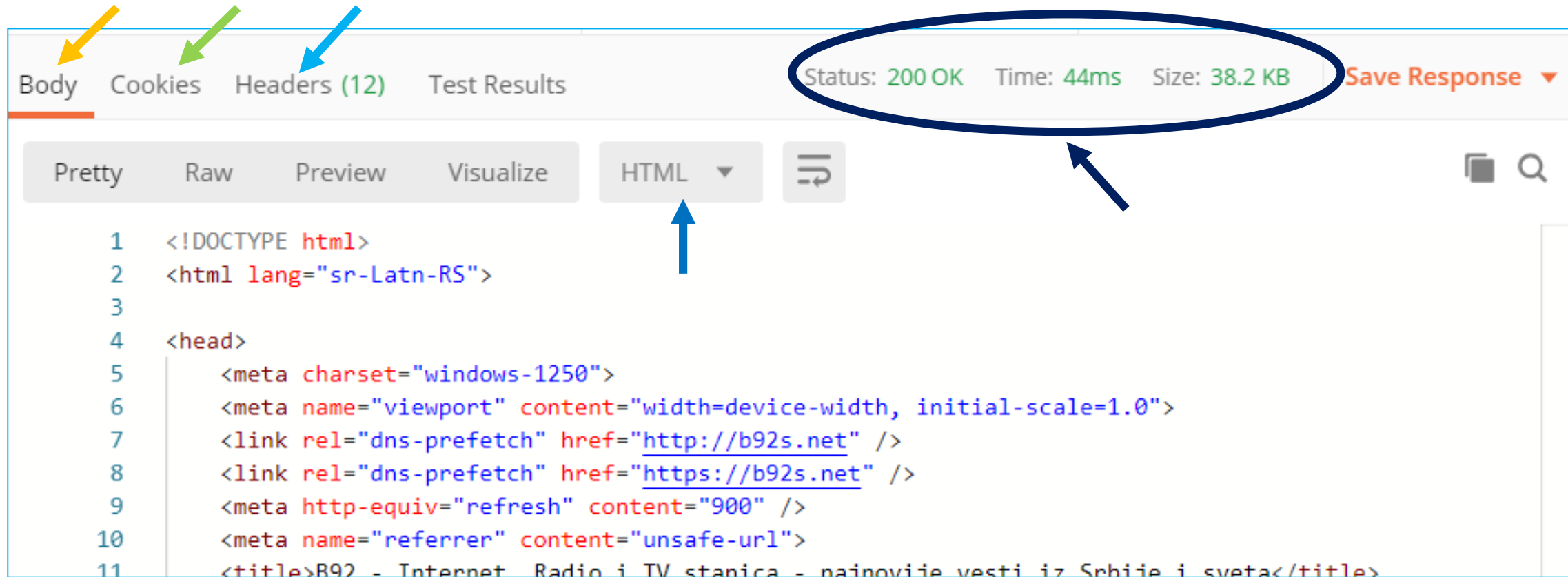
Settings



```
1 POST /index.html HTTP/1.1
2 Host: www.b92.net
3 mojAtributZaglavlja: BBB
4 Content-Type: application/x-www-form-urlencoded
5 |
6 mojParametarForme=AAA
```

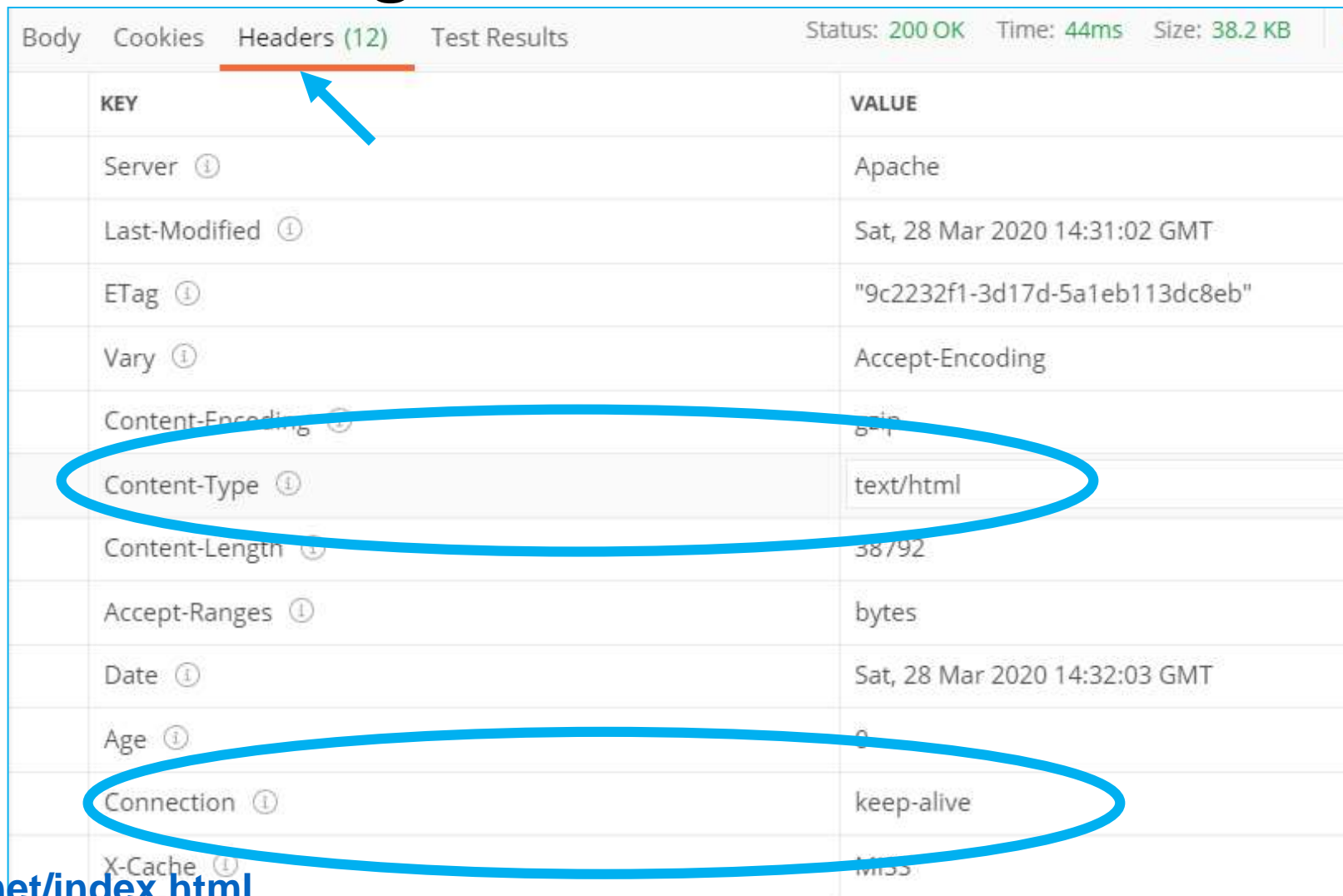
Postman i HTTP komunikacija

Inspekcija HTTP odgovora



Postman i HTTP komunikacija

Inspekcija HTTP odgovora



Body Cookies **Headers (12)** Test Results Status: 200 OK Time: 44ms Size: 38.2 KB S

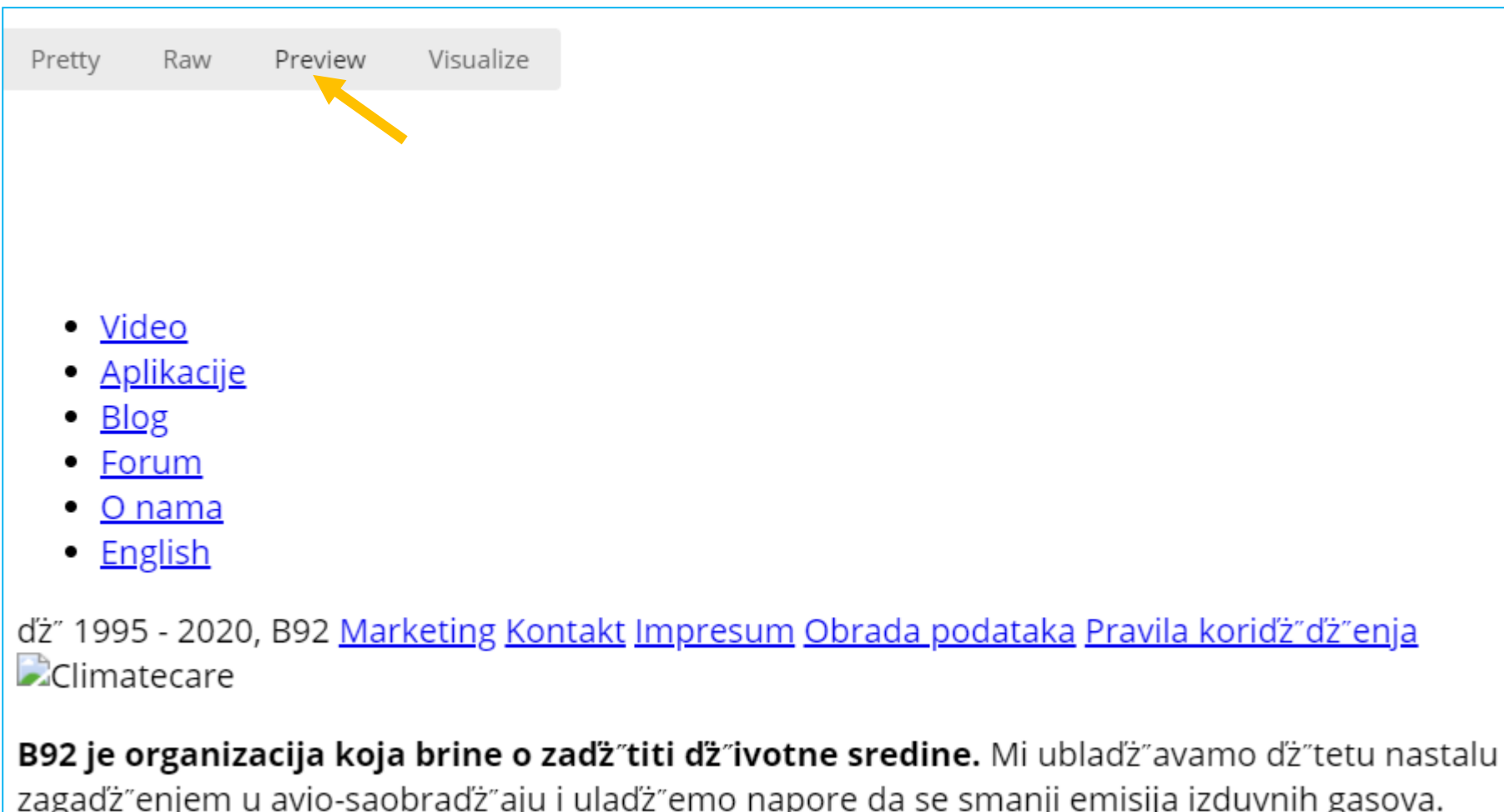
KEY	VALUE
Server ⓘ	Apache
Last-Modified ⓘ	Sat, 28 Mar 2020 14:31:02 GMT
ETag ⓘ	"9c2232f1-3d17d-5a1eb113dc8eb"
Vary ⓘ	Accept-Encoding
Content-Encoding ⓘ	gzip
Content-Type ⓘ	text/html
Content-Length ⓘ	38792
Accept-Ranges ⓘ	bytes
Date ⓘ	Sat, 28 Mar 2020 14:32:03 GMT
Age ⓘ	0
Connection ⓘ	keep-alive
X-Cache ⓘ	MISS

Postman

<https://www.b92.net/index.html>

Postman i HTTP komunikacija

Inspekcija HTTP odgovora



The screenshot shows the 'Preview' tab in Postman, which is highlighted by a yellow arrow. The response body contains a list of links, a footer with copyright information and navigation links, and a paragraph about B92.

• [Video](#)

• [Aplikacije](#)


• [Blog](#)

• [Forum](#)

• [O nama](#)

• [English](#)

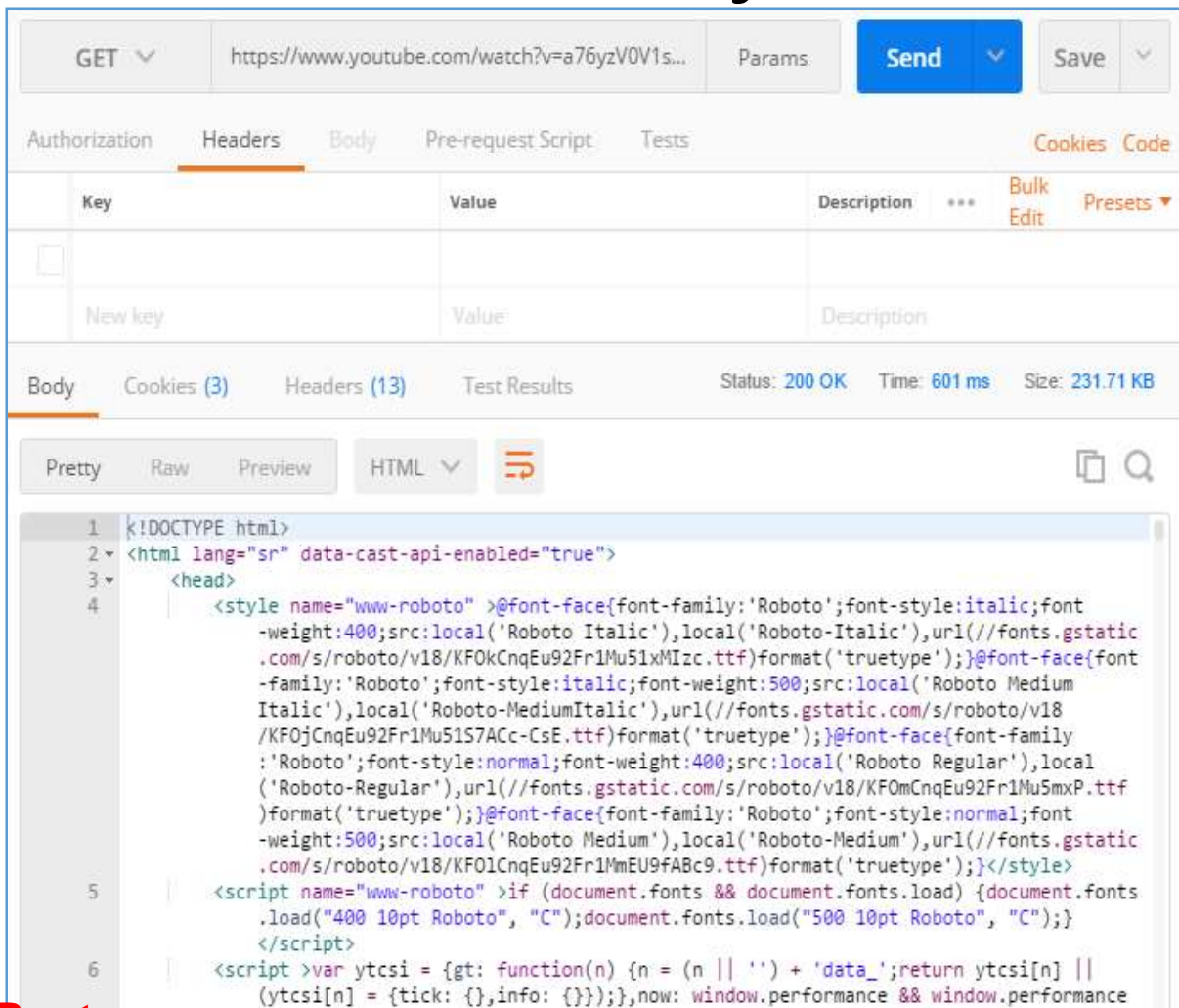
dž 1995 - 2020, B92 [Marketing](#) [Kontakt](#) [Impresum](#) [Obrada podataka](#) [Pravila korišćenja](#)

 Climatecare

B92 je organizacija koja brine o zadžiti dživotne sredine. Mi ublađžavamo džtetu nastalu zagadženjem u avio-saobradžaju i uladžemo napore da se smanji emisija izduvnih gasova.

Postman i HTTP komunikacija

HTTP komunikacija



GET `https://www.youtube.com/watch?v=a76yzV0V1s...` Params Send Save

Authorization Headers Body Pre-request Script Tests Cookies Code

Key	Value	Description	...	Bulk Edit	Presets
New key	Value	Description			

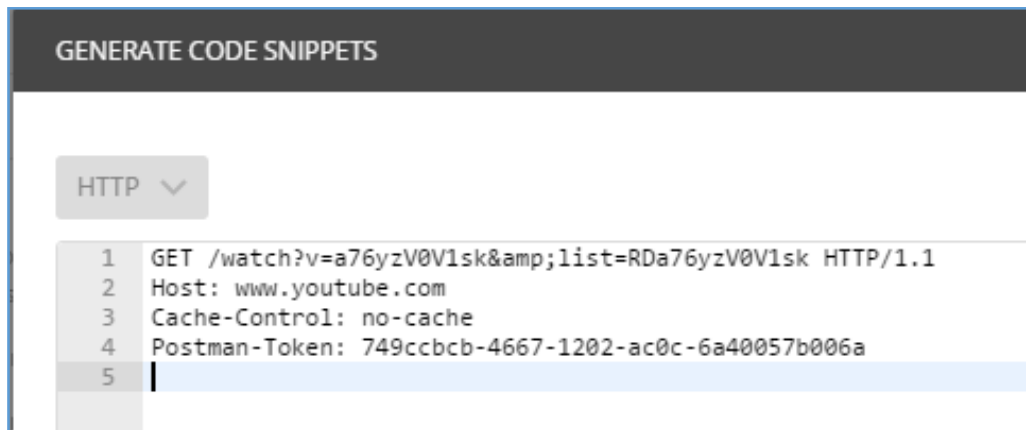
Body Cookies (3) Headers (13) Test Results Status: 200 OK Time: 601 ms Size: 231.71 KB

Pretty Raw Preview HTML

```
1 <!DOCTYPE html>
2 <html lang="sr" data-cast-api-enabled="true">
3   <head>
4     <style name="www-roboto" >@font-face{font-family:'Roboto';font-style:italic;font-weight:400;src:local('Roboto Italic'),local('Roboto-Italic'),url(//fonts.gstatic.com/s/roboto/v18/KFOkCnqEu92Fr1Mu51xMIzc.ttf)format('truetype');}@font-face{font-family:'Roboto';font-style:italic;font-weight:500;src:local('Roboto Medium Italic'),local('Roboto-MediumItalic'),url(//fonts.gstatic.com/s/roboto/v18/KFOjCnqEu92Fr1Mu51S7ACc-CsE.ttf)format('truetype');}@font-face{font-family:'Roboto';font-style:normal;font-weight:400;src:local('Roboto Regular'),local('Roboto-Regular'),url(//fonts.gstatic.com/s/roboto/v18/KFOmCnqEu92Fr1Mu5mxP.ttf)format('truetype');}@font-face{font-family:'Roboto';font-style:normal;font-weight:500;src:local('Roboto Medium'),local('Roboto-Medium'),url(//fonts.gstatic.com/s/roboto/v18/KFOlCnqEu92Fr1MmEU9fABc9.ttf)format('truetype');}</style>
5     <script name="www-roboto">if (document.fonts && document.fonts.load) {document.fonts.load("400 10pt Roboto", "C");document.fonts.load("500 10pt Roboto", "C");}
6     <script>var ytcsi = {gt: function(n) {n = (n || '') + 'data_';return ytcsi[n] || (ytcsi[n] = {tick: {},info: {}});},now: window.performance && window.performance
```

Postman

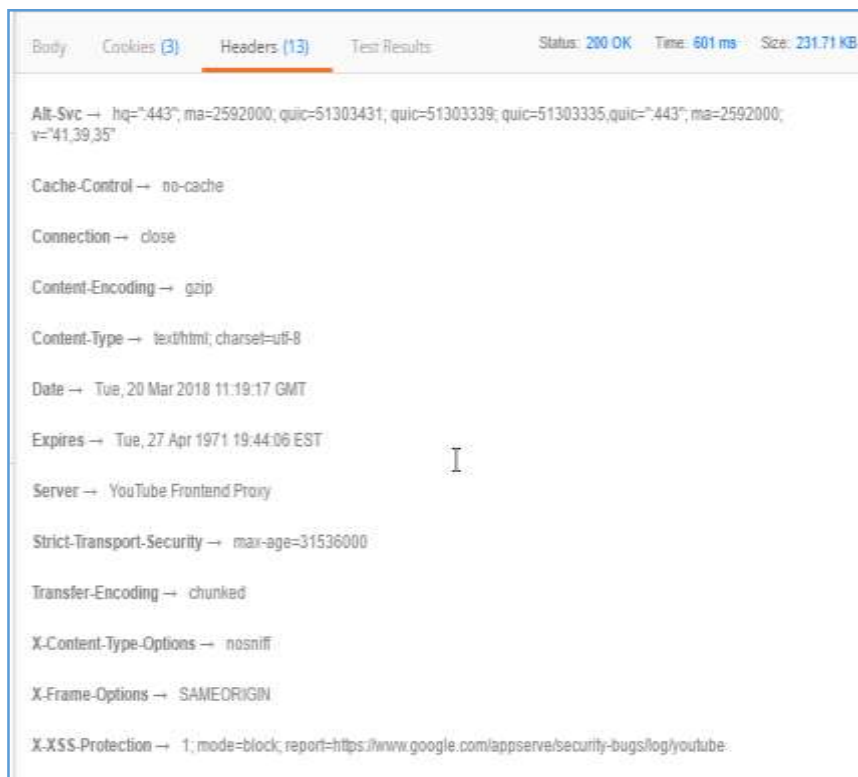
<https://www.youtube.com/watch?v=QV2iYFI5eSk>



GENERATE CODE SNIPPETS

HTTP

```
1 GET /watch?v=a76yzV0V1sk&amp;list=RDa76yzV0V1sk HTTP/1.1
2 Host: www.youtube.com
3 Cache-Control: no-cache
4 Postman-Token: 749ccbc-b4667-1202-ac0c-6a40057b006a
5
```



Body Cookies (3) Headers (13) Test Results Status: 200 OK Time: 601 ms Size: 231.71 KB

Alt-Svc → hq="443";ma=2592000;quic=51303431;quic=51303339;quic=51303335;quic="443";ma=2592000;v="41.39.35"

Cache-Control → no-cache

Connection → close

Content-Encoding → gzip

Content-Type → text/html; charset=utf-8

Date → Tue, 20 Mar 2018 11:19:17 GMT

Expires → Tue, 27 Apr 1971 19:44:06 EST

Server → YouTube Frontend Proxy

Strict-Transport-Security → max-age=31536000

Transfer-Encoding → chunked

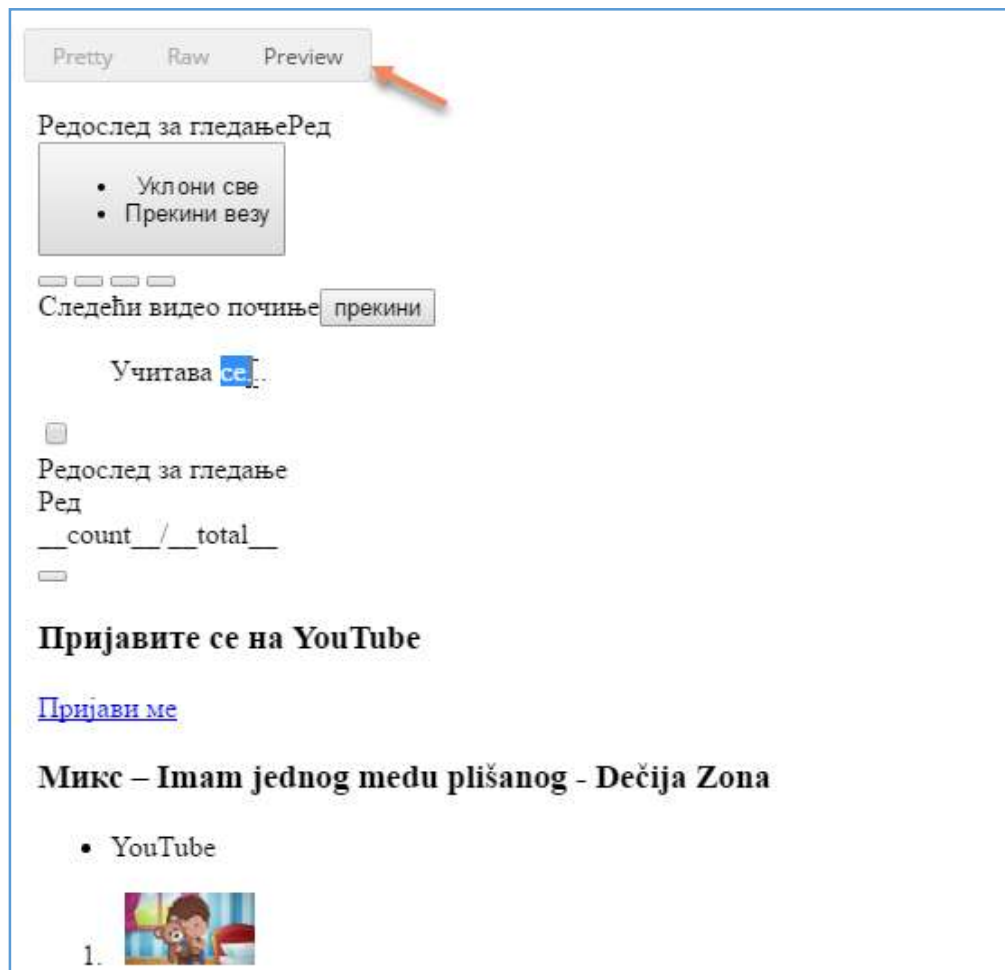
X-Content-Type-Options → nosniff

X-Frame-Options → SAMEORIGIN

X-XSS-Protection → 1; mode=block; report=https://www.google.com/appservise/security-bugs/log/youtube

Postman i HTTP komunikacija

Inspekcija HTTP odgovora

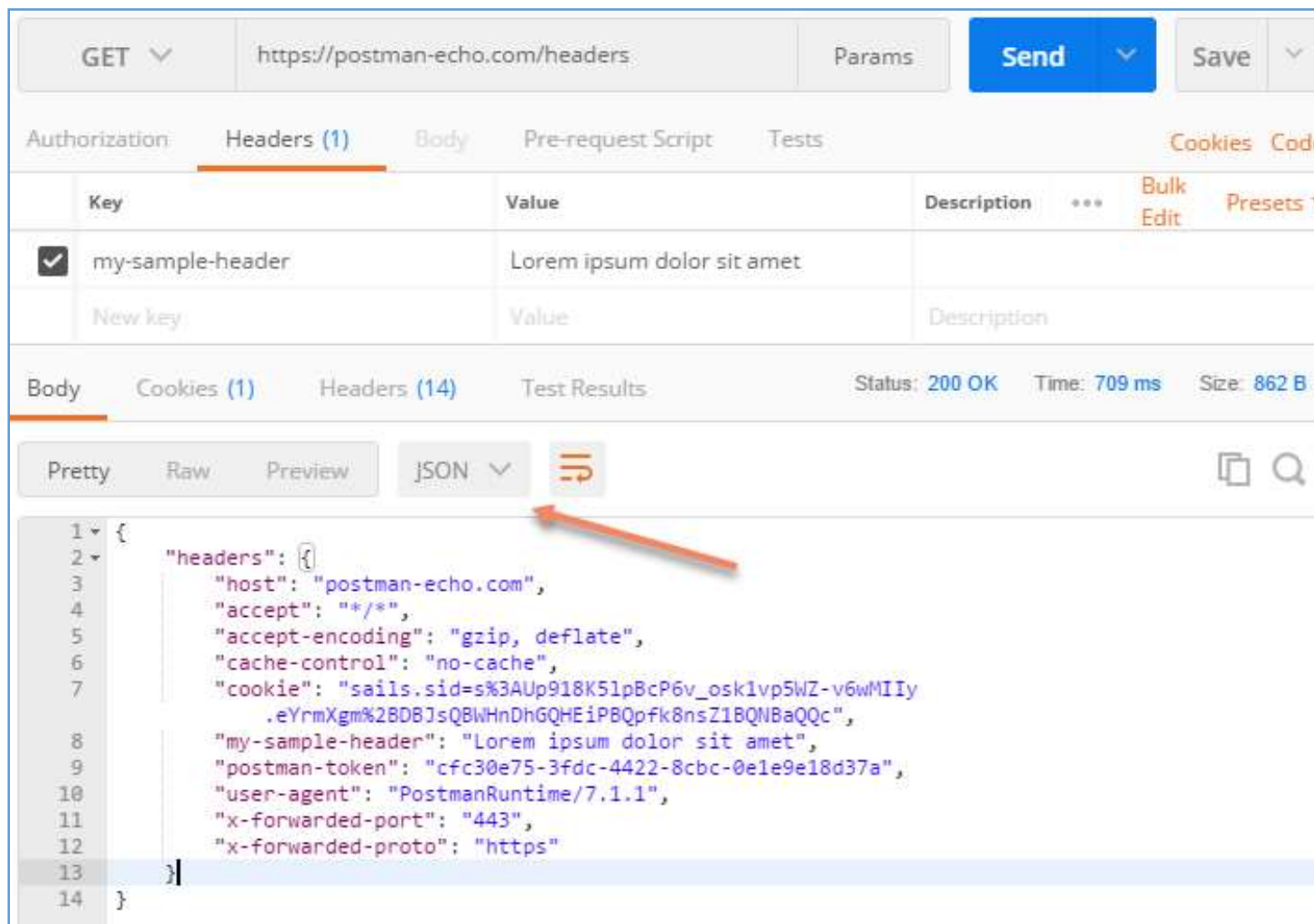


Postman

<https://www.youtube.com/watch?v=QV2iYFI5eSk>

Postman i HTTP komunikacija

HTTP komunikacija

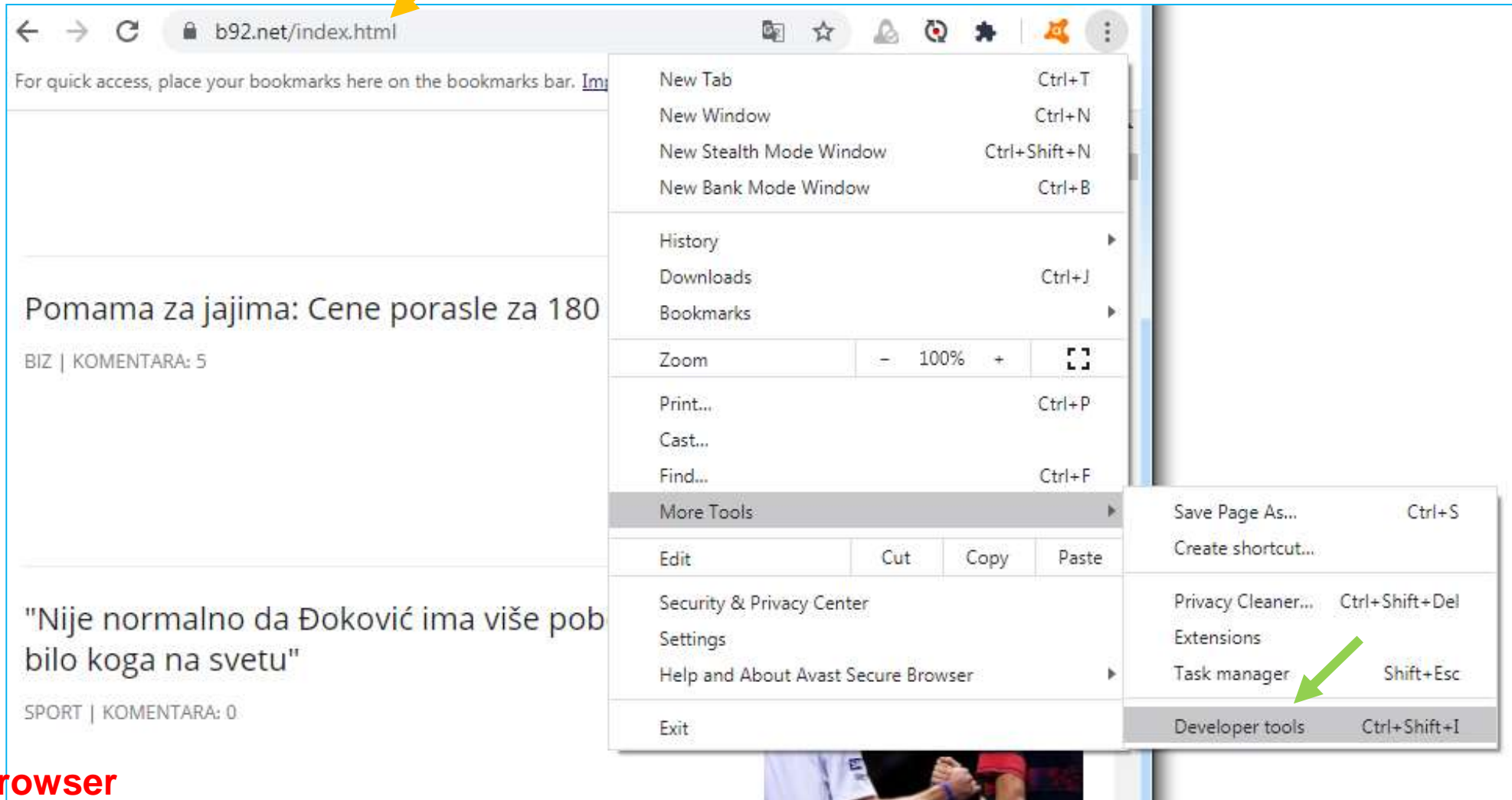


Postman

<https://postman-echo.com/headers>

Google Chrome i HTTP komunikacija

Developer tools

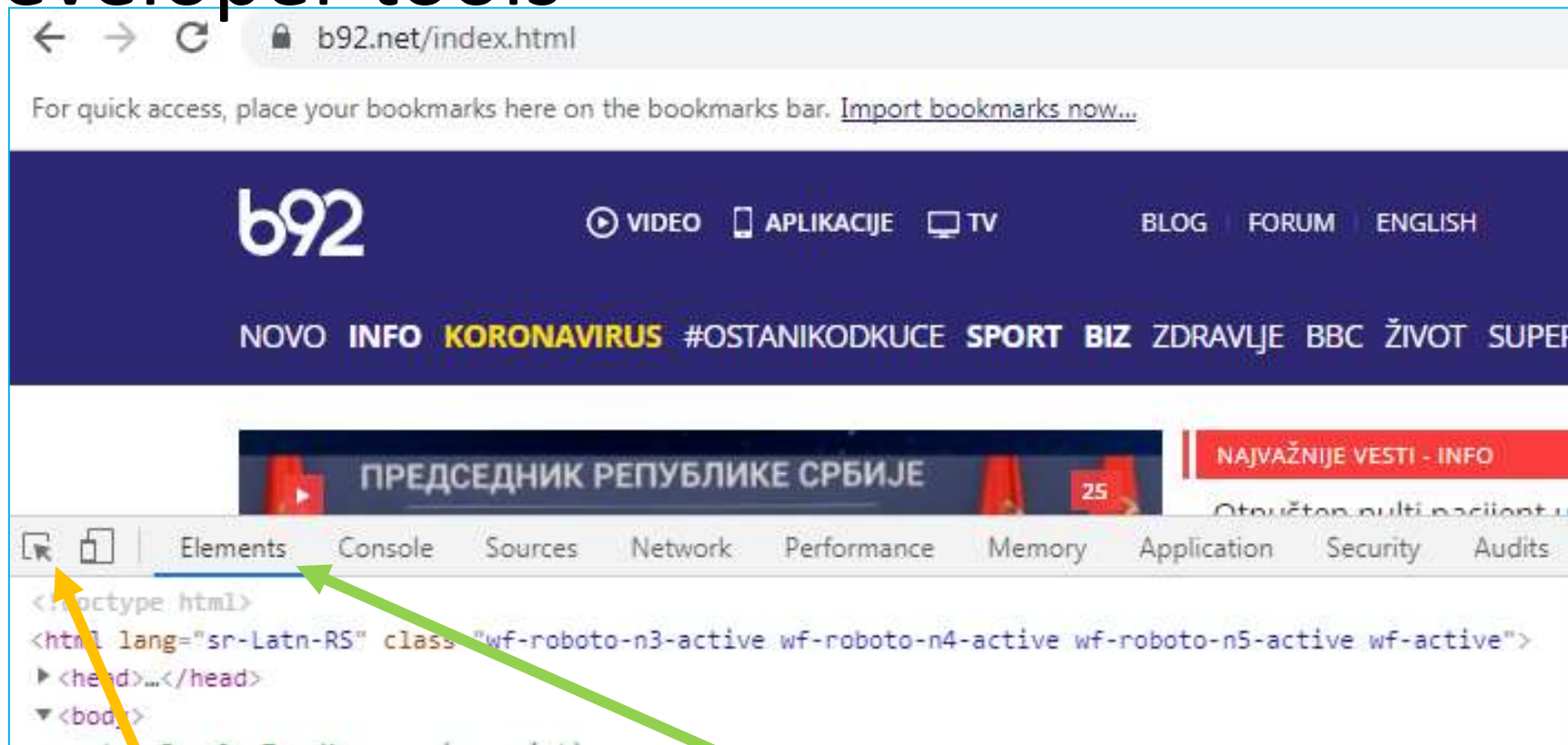


Chrome browser

<https://www.b92.net/index.html>

Google Chrome i HTTP komunikacija

Developer tools



Selekcija HTML elemenata na stranici

HTML kod

Chrome browser

<https://www.b92.net/index.html>

Google Chrome i HTTP komunikacija

Developer tools

Selektovan HTML elementat

The screenshot displays the Google Chrome Developer Tools interface. The top navigation bar includes links for VIDEO, APLIKACIJE, TV, BLOG, FORUM, and ENGLISH. Below this, a header section contains various news categories like KORONAVIRUS, SPORT, and BIZ. The main content area shows a list of news items, with the first item being a logo for 'a.logo' with dimensions 56x30. The Developer Tools panel is open, showing the 'Elements' tab on the left and the 'Styles' tab on the right. The 'Elements' tab displays the HTML structure, with the selected element being a link with class 'logo' and href 'http://www.b92.net'. The 'Styles' tab shows the CSS rules for the selected element, including a rule for 'color' that is currently set to 'internal-root-color'. A green arrow points from the text 'CSS kod za element' to the 'Styles' tab. A diagram at the bottom right illustrates the box model for the selected element, showing the margin, border, padding, and the content area with dimensions 56x30.

Elements Console Sources Network Performance Memory Application >>

```
<div id="fb-root" class="fb_reset">...</div>
<script>...</script>
<script>...</script>
<script>...</script>
<div class="banner-top above">...</div>
<header id="top">
  <div class="header-top">
    <div class="container upscore-pos-1">
      <div class="header-master">
        <a class="logo" href="http://www.b92.net"></a>
        <ul class="services" style="margin-left: 100px;">...</ul>
        <ul class="us">...</ul>
        <div class="search">...</div>
      </div>
    </div>
  </div>
</header>
</div>
```

Styles Computed Event Listeners DOM Breakpoints Properties Accessibility

Filter

Inherited from html

```
html {
  color: internal-root-color;
}
```

margin -

border -

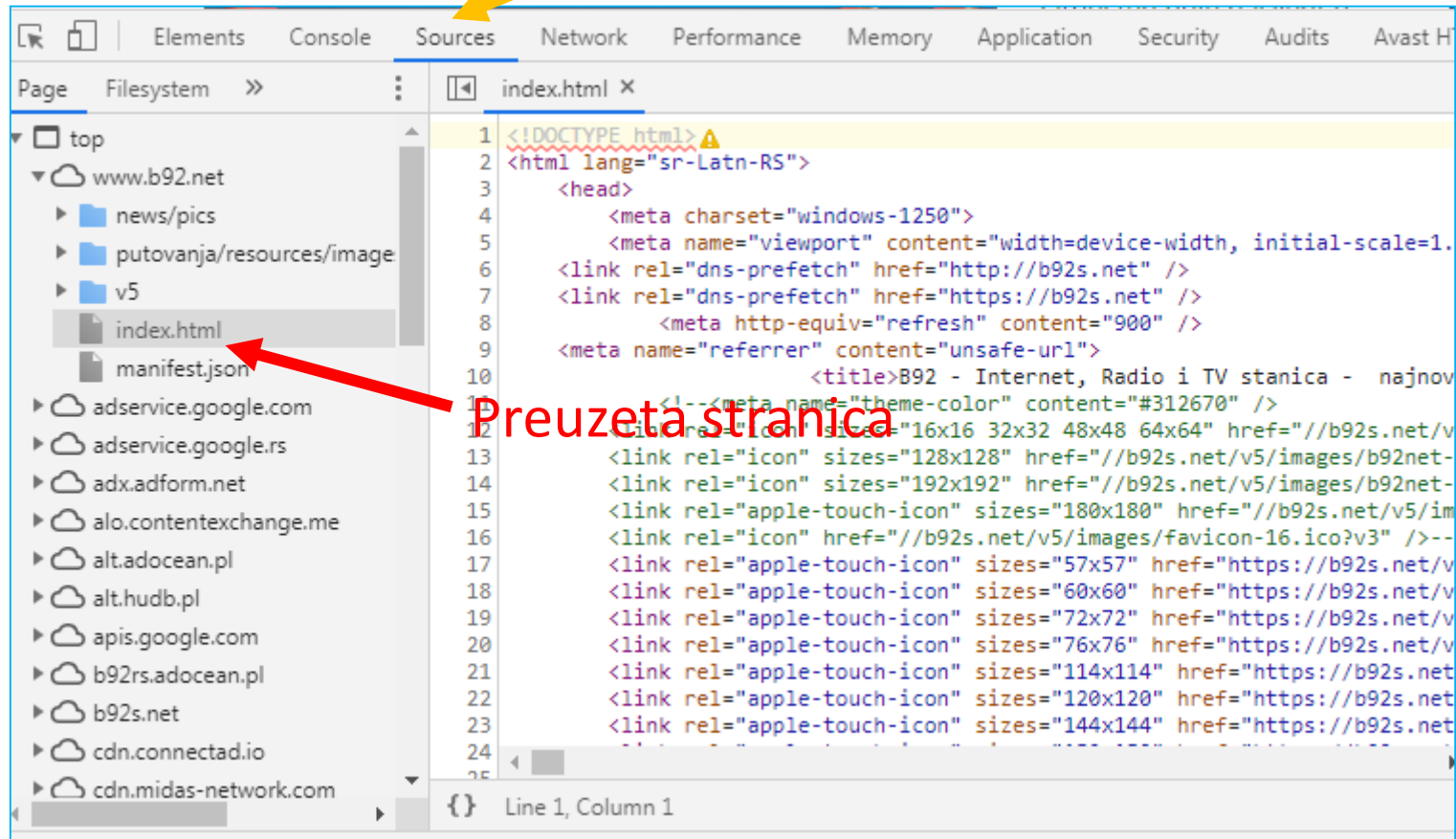
padding -

56 x 30

Google Chrome i HTTP komunikacija

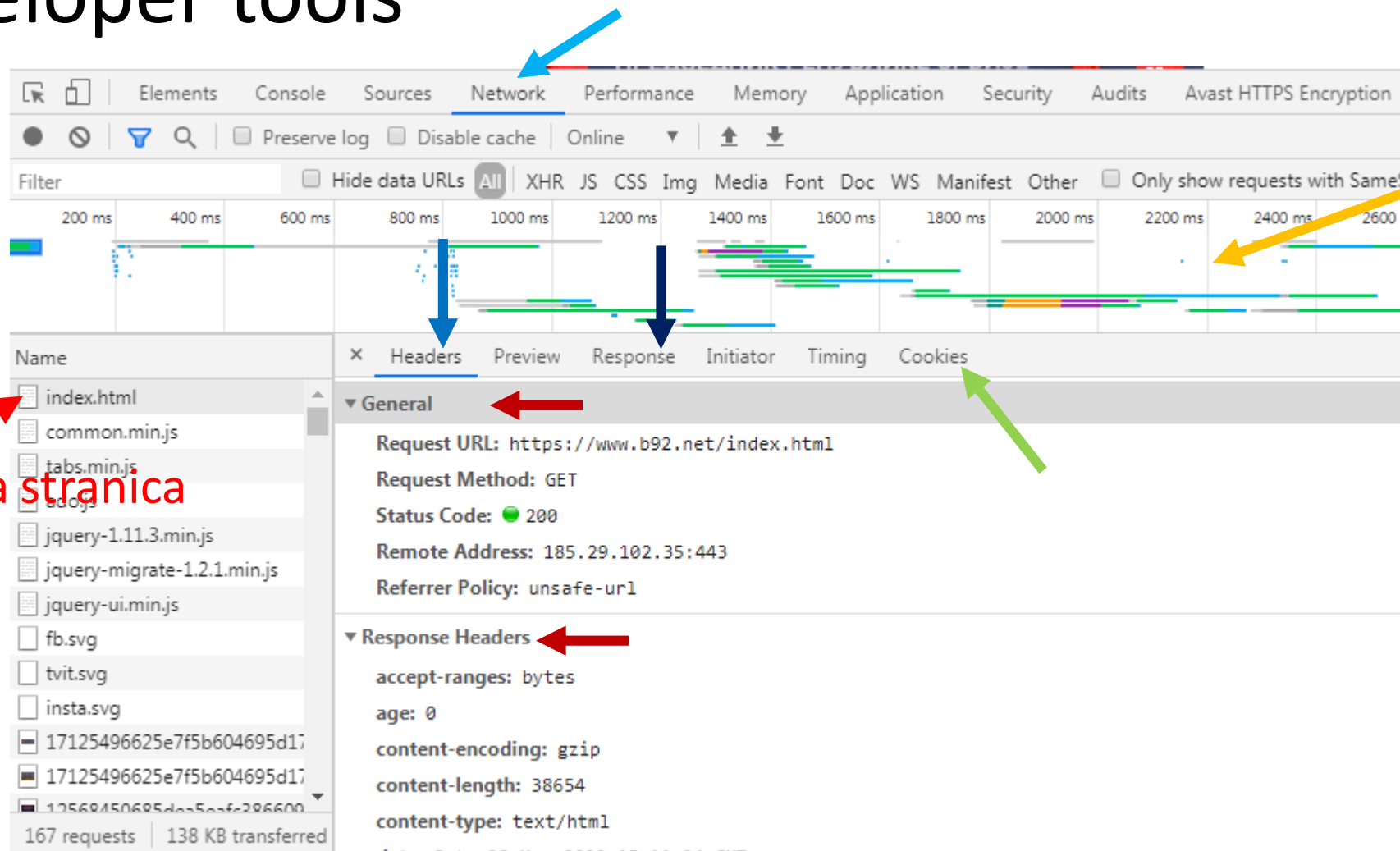
Developer tools

Preuzeti resursi



Google Chrome i HTTP komunikacija

Developer tools



Vremenska osa

Preuzeta stranica

HTTP zahtev

Struktura

- Počinje redom:
METHOD /putanja HTTP/verzija
- METHOD je:
 - GET,
 - POST, i dr.
- dodatni redovi sadrže attribute oblika:
Ime: vrednost
- prazan red na kraju
 - Ako je POST zahtev posle praznog reda idu parametri forme

HTTP zahtev

Metode

- GET – zahteva resurs od web servera
- POST – šalje parametre forme i traži odgovor
- HEAD – zahteva samo HTTP odgovor (response), bez slanja samog resursa
- PUT – omogućava klijentu da pošalje datoteku na web server
- OPTIONS – od web servera se traži spisak metoda koje podržava
- DELETE – omogućava klijentu da obriše resurs sa web servera

HTTP zahtev

Atributi u HTTP zahtevu

- User-Agent – identifikuje web browser

User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.2) Gecko/20070219 Firefox/2.0.0.2

- Accept – definiše koje tipove resursa navigator prihvata kao odgovor na ovaj zahtev

Accept:

text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5

- Accept-Language – definiše koji jezike očekuje kao odgovor

Accept-Language: en-us,en;q=0.5

- Accept-Encoding – definiše koje kodiranje očekuje kao odgovor

Accept-Encoding: gzip,deflate

HTTP zahtev

Atributi u HTTP zahtevu

- Accept-Charset – definiše koju kodnu stranu očekuje

Accept-Charset: ISO-8859-1,utf-8;q=0.8,ASCII;q=0.7,*;q=0.6

- Cookie – definiše mehanizam praćenja sesije

Cookie: id1172566682241_1=1172566682241_1

- Referer – definiše URL sa kojeg se došlo na ovu stranicu
 - koristi se za statistiku
 - hotlinking

Referer: http://localhost/

- Connection – HTTP1.1 "kaže" serveru da ne zatvara konekciju po isporuci resursa

Connection: Keep-Alive

- q= broj definiše *qvalue*, a predstavlja **relative quality factor** odn. floating point vrednost "težine" parametra
- favorizovani Charset je ISO-8859-1 ili utf-8, ali ukoliko oni nisu podržani može i ASCII, a ako ništa od prethodnog nije podržano prihvaću i * (bilo koji drugi)

Primer HTTP zahteva

GET /index.html **HTTP/1.1**


User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)

Host: www.b92.net

Accept-Language: en-us

Accept-Encoding: gzip, deflate

Connection: Keep-Alive



Ako je vrednost **Close**,
konekcija se zatvara

HTTP zahtev

Keep Alive

HTTP persistent connection, also called HTTP keep-alive, or HTTP connection reuse, is the idea of using a single TCP connection to send and receive multiple HTTP requests/responses, as opposed to opening a new connection for every single request/response pair.

The newer HTTP/2 protocol uses the same idea and takes it further to allow multiple concurrent requests/responses to be multiplexed over a single connection.

HTTP odgovor

Struktura

- Počinje redom:
HTTP/verzija kod tekstualni_opis
- dodatni redovi sadrže attribute:
Ime: vrednost
- prazan red
- sledi sadržaj datoteke

HTTP odgovor

Kodovi odgovora

<https://www.restapitutorial.com/httpstatuscodes.html>

1xx Informational

100 Continue

101 Switching Protocols

102 Processing (WebDAV)

2xx Success

★ 200 OK ←

203 Non-Authoritative Information

206 Partial Content

226 IM Used

★ 201 Created ←

★ 204 No Content ←

207 Multi-Status (WebDAV)

202 Accepted ←

205 Reset Content

208 Already Reported (WebDAV)

3xx Redirection

300 Multiple Choices

303 See Other

306 (Unused)

301 Moved Permanently ←

★ 304 Not Modified ←

307 Temporary Redirect

302 Found ←

305 Use Proxy

308 Permanent Redirect (experimental)

4xx Client Error

★ 400 Bad Request ←

★ 403 Forbidden ←

406 Not Acceptable

★ 409 Conflict

412 Precondition Failed

415 Unsupported Media Type

418 I'm a teapot (RFC 2324)

423 Locked (WebDAV)

426 Upgrade Required

431 Request Header Fields Too Large

450 Blocked by Windows Parental Controls (Microsoft)

★ 401 Unauthorized ←

★ 404 Not Found ←

407 Proxy Authentication Required

410 Gone

413 Request Entity Too Large

416 Requested Range Not Satisfiable

420 Enhance Your Calm (Twitter)

424 Failed Dependency (WebDAV)

428 Precondition Required

444 No Response (Nginx)

451 Unavailable For Legal Reasons

402 Payment Required

405 Method Not Allowed

408 Request Timeout

411 Length Required

414 Request-URI Too Long

417 Expectation Failed

422 Unprocessable Entity (WebDAV)

425 Reserved for WebDAV

429 Too Many Requests

449 Retry With (Microsoft)

499 Client Closed Request (Nginx)

5xx Server Error

★ 500 Internal Server Error ←

503 Service Unavailable ←

506 Variant Also Negotiates (Experimental)

509 Bandwidth Limit Exceeded (Apache)

598 Network read timeout error

501 Not Implemented ←

504 Gateway Timeout

507 Insufficient Storage (WebDAV)

510 Not Extended

599 Network connect timeout error

502 Bad Gateway ←

505 HTTP Version Not Supported

508 Loop Detected (WebDAV)

511 Network Authentication Required

HTTP odgovor

Atributi u HTTP odgovoru

- Content-type – definiše tip odgovora

Content-Type: text/html

- Cache-Control – definiše kako se keš na klijentu ažurira
 - koristi se i Pragma: no-cache

Cache-Control: no-cache

- Location – definiše novu adresu kod redirekcije

Location: new.html

- Connection – potvrda klijentu da li da zatvori konekciju ili da je ostavi otvorenu

Connection: Keep-Alive

Primer HTTP odgovora

HTTP/1.0 200 OK

Date: Tue, 04 May 02004 08:55:09 GMT

Status: 200

Servlet-Engine: Tomcat Web Server/3.1 (JSP 1.1; Servlet 2.2; Java 1.4.2_02; Windows XP 5.1 x86; java.vendor=Sun Microsystems Inc.)

Content-Type: text/html

Last-Modified: Fri, 24 Oct 02003 16:07:24 GMT

Content-Length: 2524

Content-Language: en

```
<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
```

```
<html>
```

```
<head>
```

```
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

```
  <meta name="GENERATOR" content="Mozilla/4.72 [en] (WinNT; U) [Netscape]">
```

```
  <meta name="Author" content="Anil K. Vijendran">
```

```
  <title>Tomcat v3.1</title>
```

```
</head>
```

```
<body></body>
```

```
</html>
```

Slanje podataka iz formi

Get metoda

- Parametri iz forme se za GET metodu smeštaju u zaglavlje GET zahteva

- GET HTTP zahtev:

GET /PrihvatanjePodataka?ime=pera&prezime=peric HTTP/1.1

- HTML kod na klijentu

```
<form action="PrihvatanjePodataka" method="get" accept-charset="UTF-8">
```

```
  Ime :<input type="text" name="ime" /><br/>
```

```
  Prezime :<input type="text" name="prezime" /><br/>
```

```
  <input type="submit" value="Posalji" />
```

```
</form>
```

Slanje podataka iz formi

Get metoda i preuzimanje podataka

localhost:8080/DrugiServletiProjekat/formaGet.html

Forma za unos Osoba, metoda Get

Ime :	<input type="text" value="pera"/>
Prezime :	<input type="text" value="peric"/>
<input type="button" value="Posalji"/>	

Pošalji

localhost:8080/DrugiServletiProjekat/PrihvatanjePodataka?ime=pera&prezime=peric

Prihvatanje podataka

Poslali ste:pera peric

Headers Preview Response Initiator Timing Cookies

▼ General

Request URL: http://localhost:8080/DrugiServletiProjekat/PrihvatanjePodataka?ime=pera&prezime=peric

Request Method: GET

Status Code: 200

Remote Address: [::1]:8080

Referrer Policy: no-referrer-when-downgrade

▼ Query String Parameters view source view URL encoded

ime: pera

prezime: peric

inputFormGet.html

Slanje podataka iz formi

Post metoda

- Parametri iz forme se za POST metodu smeštaju u telo POST zahteva

- POST HTTP zahtev:

POST /PrihvatanjePodataka HTTP/1.1

- HTML kod na klijentu

```
<form action="PrihvatanjePodataka" method="post" accept-charset="UTF-8">
```

```
  Ime :<input type="text" name="ime" /><br/>
```

```
  Prezime :<input type="text" name="prezime" /><br/>
```

```
  <input type="submit" value="Posalji" />
```

```
</form>
```

Slanje podataka iz formi

Post metoda i preuzimanje podataka

localhost:8080/DrugiServletiProjekat/formaPost.html

Forma za unos Osoba, metoda Post

Ime :	<input type="text" value="pera"/>
Prezime :	<input type="text" value="peric"/>
<input type="button" value="Posalji"/>	

Pošalji



localhost:8080/DrugiServletiProjekat/PrihvatanjePodataka

Prihvatanje podataka

Poslali ste:pera peric

Headers Preview Response Initiator Timing Cookies

▼ General

Request URL: http://localhost:8080/DrugiServletiProjekat/PrihvatanjePodataka

Request Method: POST

Status Code: 200

Remote Address: [::1]:8080

Referrer Policy: no-referrer-when-downgrade

▼ Form Data view source view URL encoded

ime: pera

prezime: peric

inputFormPost.html

Slanje podataka iz formi

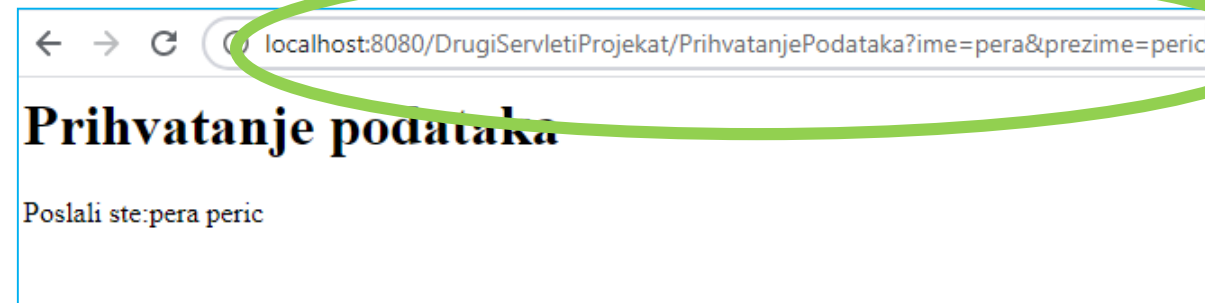
Get i Post zahtevi

Kod GET metode se parametri forme nalaze u heder delu HTTP request poruke

GET /PrihvatanjePodataka?ime=pera&prezime=peric HTTP/1.1

...

Kod HTTP **GET** metode, posle znaka ? zapisuju se vrednosti URL promenljivih tipa *ključ=vrednost&ključ=vrednost...*



Kod HTTP **POST** metode se vrednosti parametara forme smeštaju na kraju HTTP zahteva (posle praznog reda), i posle vrednosti parametara nema “\r\n” karaktera na kraju reda

Kod POST metode se parametri forme nalaze u body delu HTTP request poruke

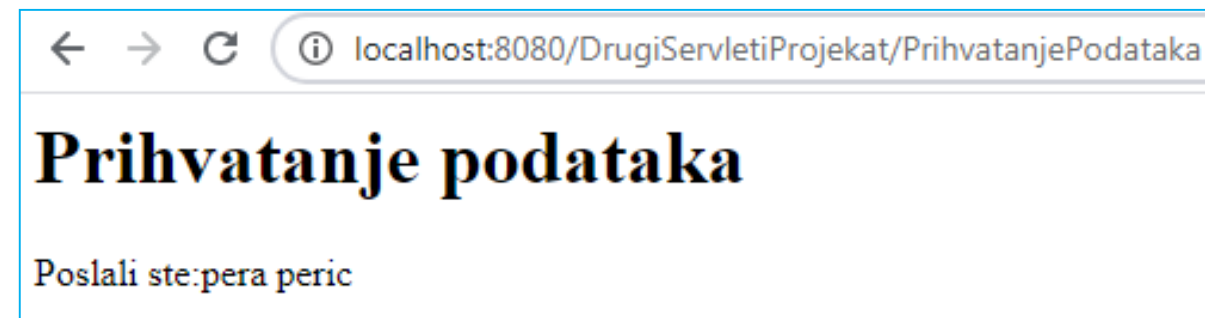
POST /PrihvatanjePodataka HTTP/1.1

...

Content-length: 22

...

ime=pera&prezime=peric



Slanje podataka iz formi

Content type: multipart/form-data

- Ukoliko je potrebno uz pomoć forme poslati datoteku sa klijenta na server koristi se HTTP Post metoda pri čemu se atribut forme enctype mora postaviti na vrednost multipart/form-data

```
<form action="PrihvatanjePodatakaDatoteka" method="post"
accept-charset="UTF-8" enctype="multipart/form-data">
```

```
Ime: <input type="text" name="ime" /><br/>
```

```
Prezime: <input type="text" name=" prezime" /> <br/>
```

```
Datoteka: <input type="file" name="datoteka" /><br/>
```

```
<input type="submit" value="Posalji" />
```

```
</form>
```


Slanje podataka iz formi

Content type: multipart/form-data

- Prethodno rezultuje da se u poslatom HTTP zahtevu vrednost atribut zaglavlja Content-Type definiše kao multipart/form-data koji će imati definisan boundary. Takođe u atributu Content-Length definiše se količina poslatih podataka
- Boundary predstavlja graničnik između različitih input delova forme koji su poslati i obično je oblika boundary =----kod npr.
-----WebKitFormBoundarygTBBZYVATsP8QhhM

▼ Request Headers

[view source](#)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image

Accept-Encoding: gzip, deflate, br

Accept-Language: en-US,en;q=0.9,sr;q=0.8

Cache-Control: max-age=0

Connection: keep-alive

Content-Length: 473

Content-Type: multipart/form-data; boundary=-----WebKitFormBoundarygTBBZYVATsP8QhhM

Slanje podataka iz formi

Content type: multipart/form-data

- Izgled tela HTTP zahteva bio bi u formi da se svaki pojedinačni parametar forme okruži boundary vrednošću

-----WebKitFormBoundarygTBBZYVATsP8QhhM
Content-Disposition: form-data; name="ime"

steva

-----WebKitFormBoundarygTBBZYVATsP8QhhM
Content-Disposition: form-data; name="prezime"

marković

-----WebKitFormBoundarygTBBZYVATsP8QhhM
Content-Disposition: form-data; name="datoteka";
filename="zaduzenja opreme.txt"
Content-Type: text/plain

Petar Petrović kamera

marko Marković slusalice

Jova Jovanović lap top

-----WebKitFormBoundarygTBBZYVATsP8QhhM--

Isporuka WWW sadržaja

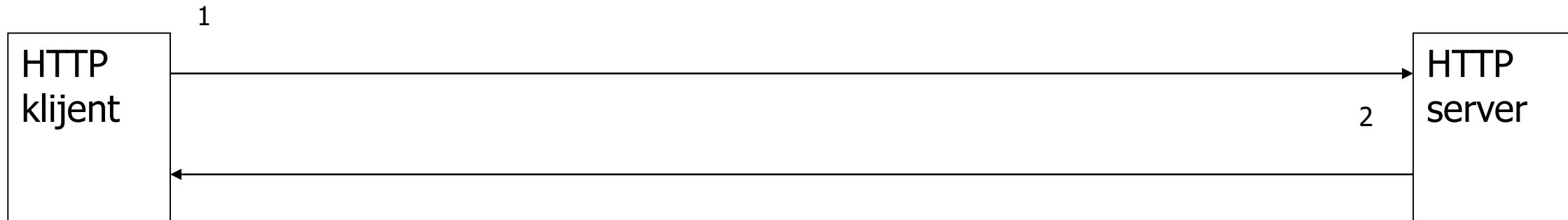
Vrste WWW sadržaja

- statički (unapred uskladišteni)
- dinamički (generisani po zahtevu)

Isporuka WWW sadržaja

Isporuka statičkog sadržaj

statički sadržaji se nalaze u okviru datoteka WWW servera

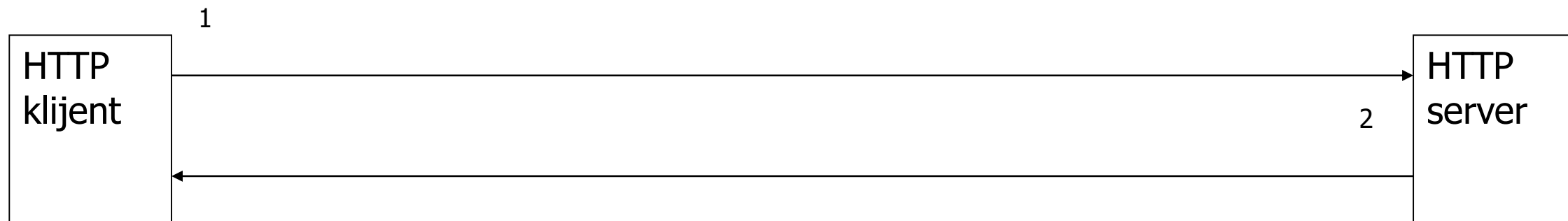


1. klijent zahteva datoteku
2. server je učitava sa svog fajl-sistema i šalje je klijentu

Isporuka WWW sadržaja

Isporuka dinamičkog sadržaja

traženi sadržaj se generiše po zahtevu i šalje klijentu



1. klijent zahteva "datoteku"
2. server je generiše i šalje klijentu. Datoteka se ne snima u fajl sistemu servera