



Национална програма  
"Обучение за ИТ умения и кариера"  
<https://it-kariera.mon.bg/e-learning/>

Министерството на  
образованието и науката  
<https://www.mon.bg>



# HTTP

Клиент, сървър, заявка, отговор

# Съдържание

1. Основи на HTTP
2. Формуляри
3. URL
4. MIME и типове медии
5. HTTP Заявки
6. HTTP Отговори



http://

# ОСНОВИ НА HTTP

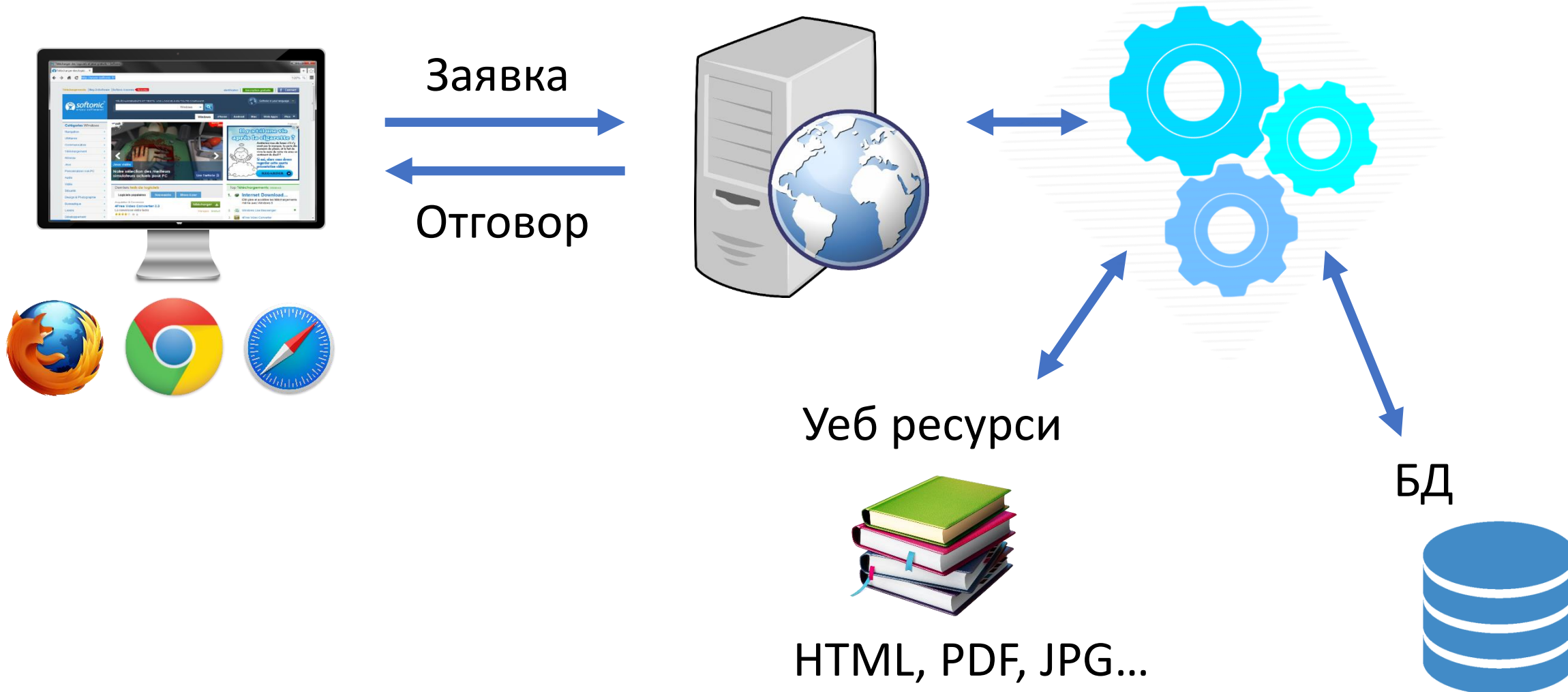
Заявка и отговори

# Работен модел на уеб сървър

Уеб клиент

Уеб сървър

Технология



# Работен модел на уеб сървър

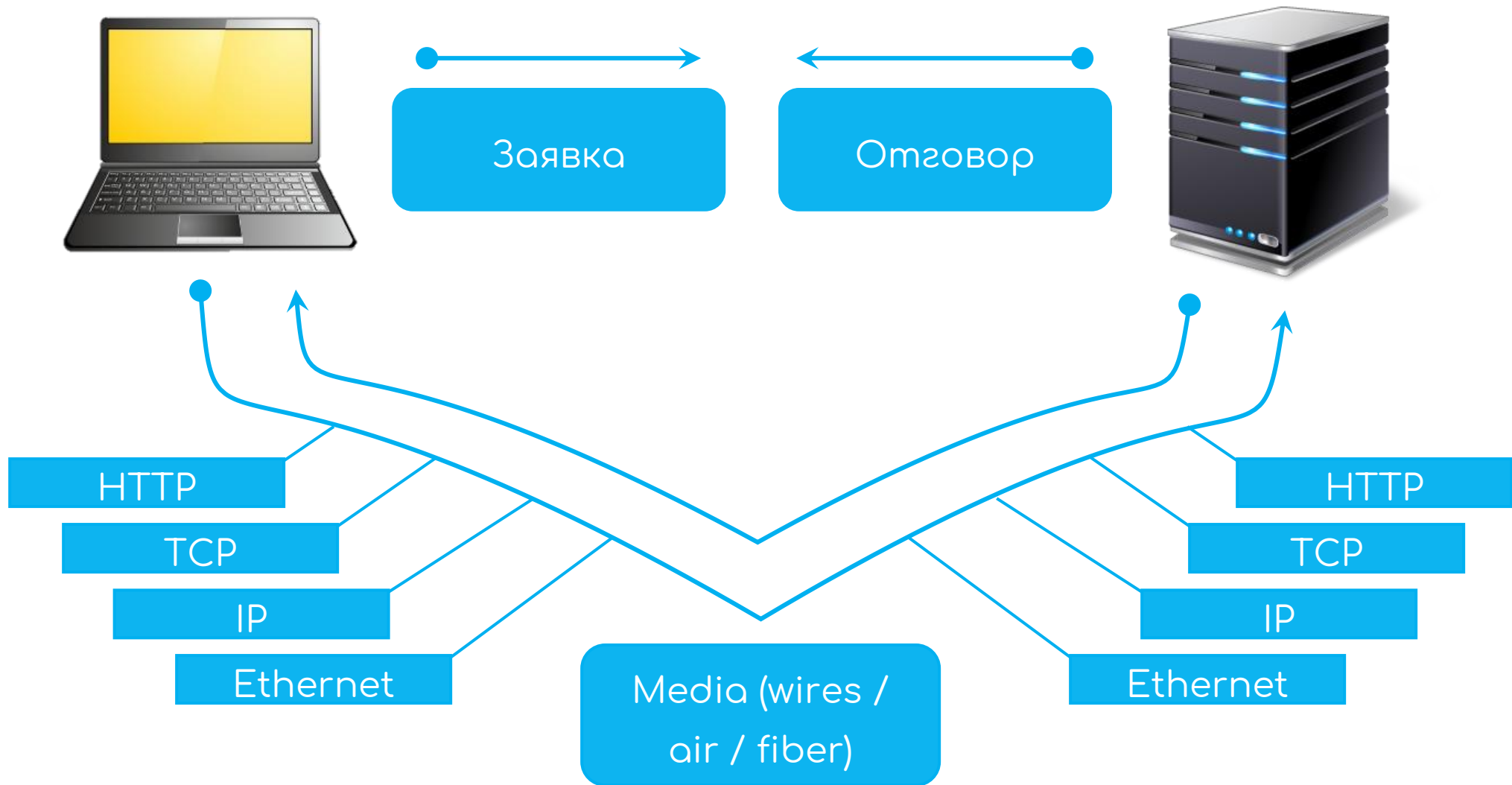
Уеб клиент

Уеб сървър

Технология



# HTTP (Hyper Text Transfer Protocol)



# HTTP методи за заявка

- HTTP дефинира методи за посочване на желаното действие, което трябва да се извърши върху идентифицирания ресурс

Метод	Описание
GET	Извличане / зареждане на ресурс
POST	Създаване / съхраняване на ресурс
PUT	Актуализиране на ресурс
DELETE	Премахване на ресурс

# HTTP общуване: Пример

- HTTP заявка:

```
GET /courses/javascript HTTP/1.1
Host: www.mon.bg
User-Agent: Mozilla/5.0
<CRLF>
```

Празният ред обозначава  
края на хедъра на заявката

- HTTP отговор:

```
HTTP/1.1 200 OK
Date: Mon, 5 Jul 2010 13:09:03 GMT
Server: Microsoft-HTTPAPI/2.0
Last-Modified: Mon, 12 Jul 2014 15:33:23 GMT
Content-Length: 54
<CRLF>
<html><title>Hello</title>
Welcome to our site</html>
```

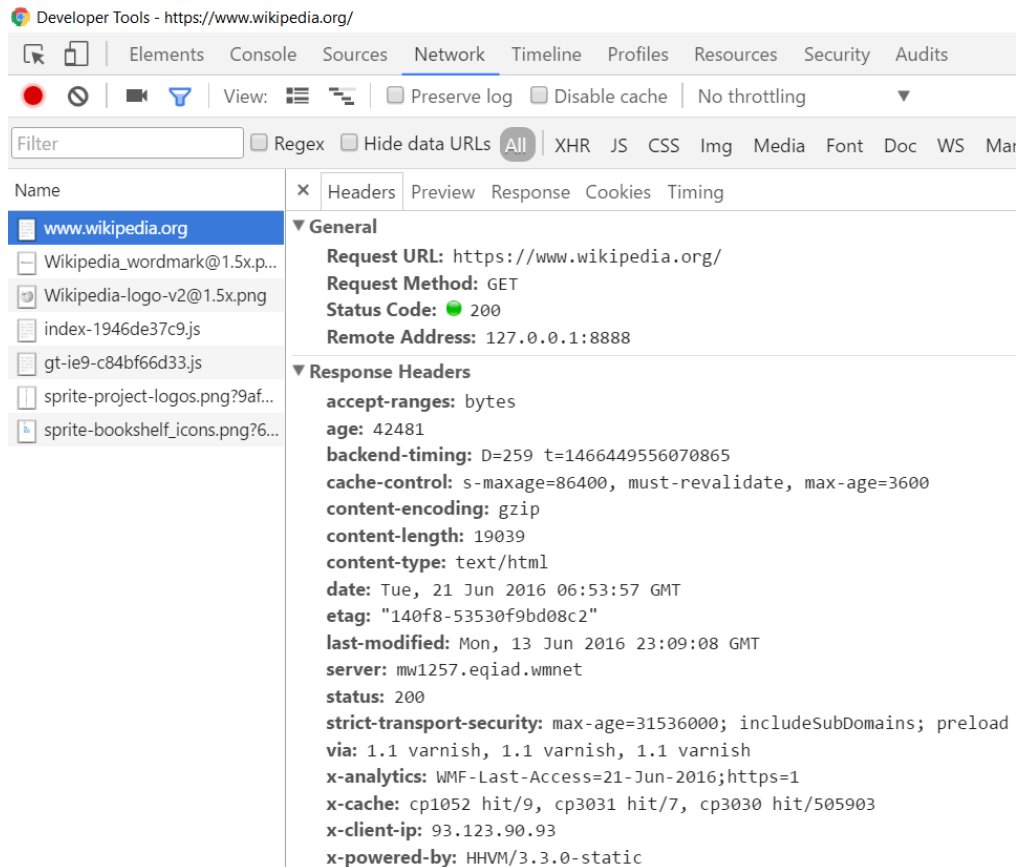
Празният ред обозначава  
края на хедъра на отговора



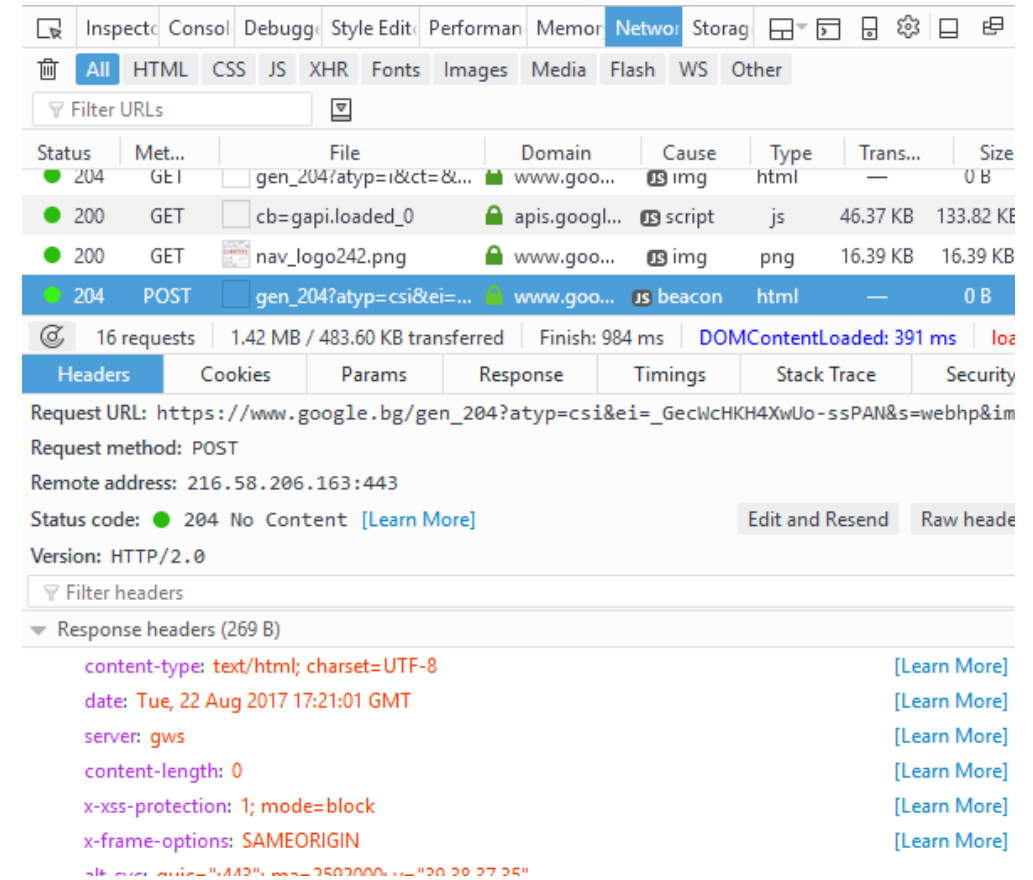


Инструменти за  
разработчици

# Browser Dev Tools



[Chrome Developer Tools](#)

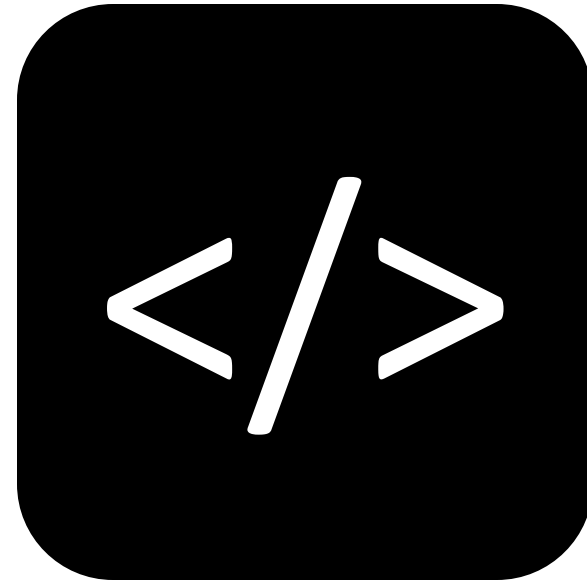


[Mozilla Developer Tools](#)

# Добавки към браузъра



[Postman](#) - Chrome



[Rested](#) - Firefox

# Desktop инструменты



[Fiddler](#)



[Postman](#)

```
<form method="post">  
  Name: <input type="text" name="name"/> <br/>  
  Age: <input type="text" name="age"/> <br/>  
  <input type="submit" />  
</form>
```

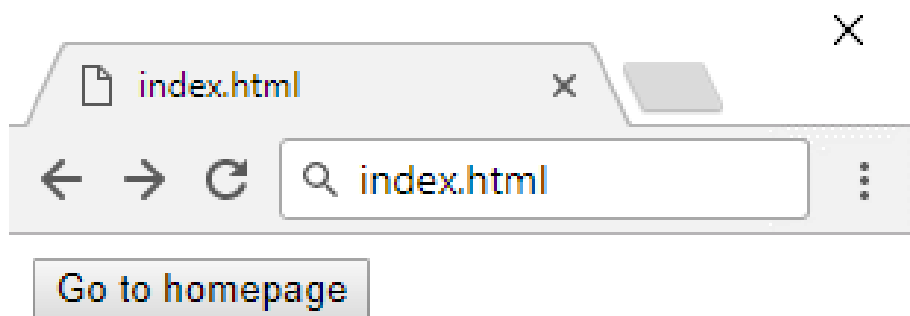
# HTML формуляри

# HTML формуляри - атрибутът "Action"

- Определя къде да изпратите данните на формуляра:

```
<form action="home.html">  
  <input type="submit" value="Go to homepage"/>  
</form>
```

Релативен URL към текущия файл



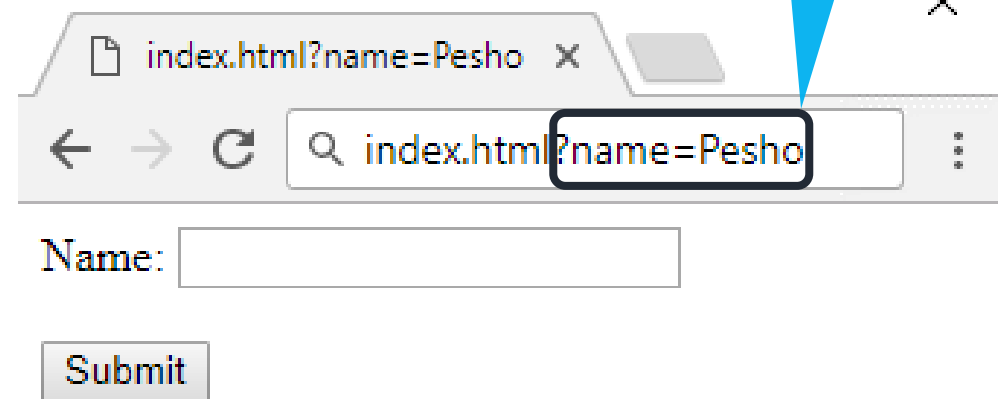
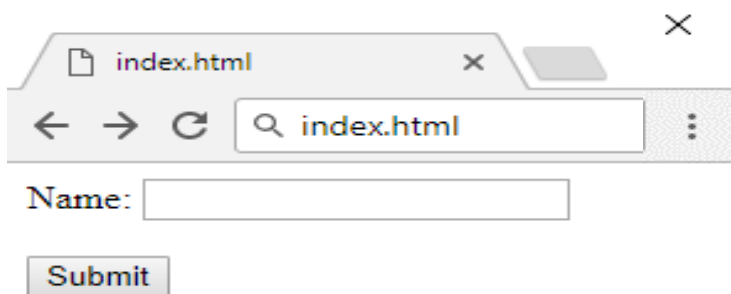
**Welcome to Our Home Page :)**

# HTML формуляри – атрибутът "Method"

- Указва HTTP метода, който да се използва при изпращане на данни от формуляра

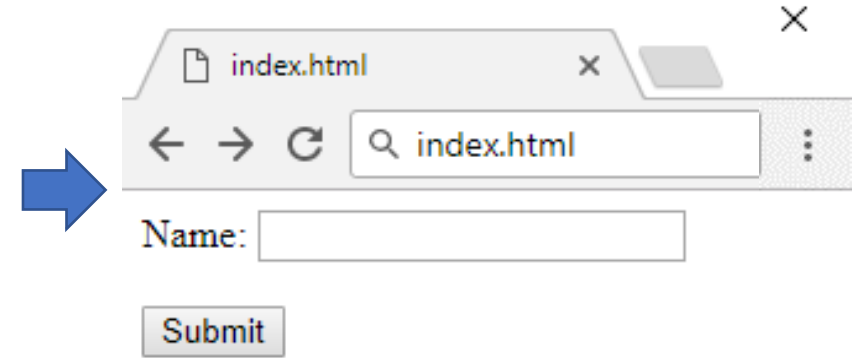
```
<form method="get">  
  Name: <input type="text" name="name">  
  <br /><br />  
  <input type="submit" value="Submit">  
</form>
```

Данните  
са в URL  
адреса



# HTML формуляри – атрибутът "Method" (2)

```
<form method="post">  
  Name: <input type="text" name="name">  
  <br /><br />  
  <input type="submit" value="Submit">  
</form>
```



Хедър на  
заявката

```
POST http://localhost/index.html HTTP/1.1  
Host: localhost  
Content-Type: application/x-www-form-urlencoded  
Content-Length: 10
```

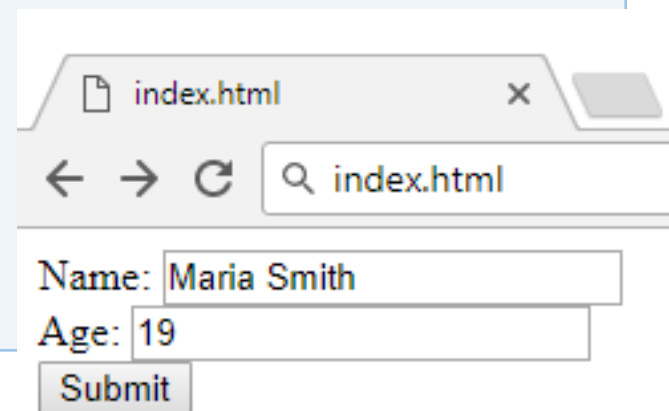
name=Pesho

HTTP тялото за заявки съхранява  
данните от формуляра за заявка и  
данните за отговор



# Данни за формуляр, кодирани в URL адрес

```
<form method="post">  
  Name: <input type="text" name="name"/> <br/>  
  Age: <input type="text" name="age"/> <br/>  
  <input type="submit" />  
</form>
```



A screenshot of a web browser window. The tab is titled 'index.html'. The address bar shows 'index.html'. The form contains two text input fields: 'Name: Maria Smith' and 'Age: 19'. Below the fields is a 'Submit' button.



```
POST http://localhost/cgi-bin/index.cgi HTTP/1.1  
Host: localhost  
Content-Type: application/x-www-form-urlencoded  
Content-Length: 23
```

`name=Maria+Smith&age=19`

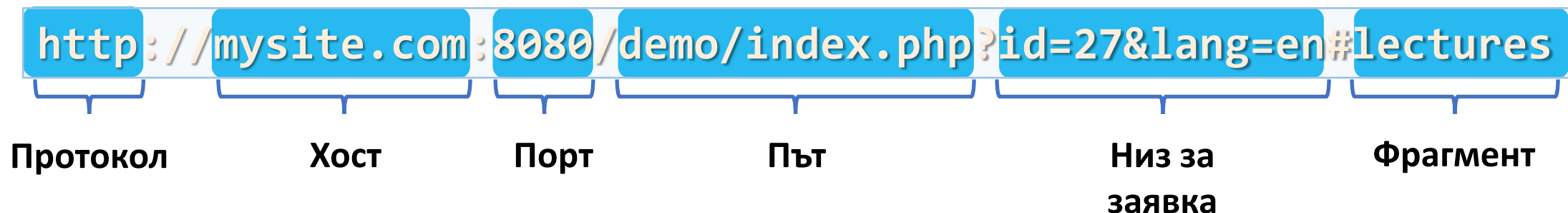
Качването на файлове  
не се поддържа



# URL

Uniform Resource Locator

# Uniform Resource Locator (URL)



- URL адресът е форматиран низ, състоящ се от:
  - Протокол за комуникация (**http**, **ftp**, **https**...) – HTTP в повечето случаи
  - Хост или IP адрес (**www.mon.bg**, **gmail.com**, **127.0.0.1**, **web**)
  - Порт (стандартният порт е 80) – число в обхвата [0...65535]
  - Път (**/forum**, **/path/index.php**)
  - Низ за заявка(**?id=27&lang=en**)
  - Фрагмент (**#lectures**) – used on the client to navigate to some section

# Низове за заявки в C#

- Съдържа данни, които не са част от структурата на пътя
- Често използван при търсения и динамични страници
- Частта от URL след въпросителен знак (?)
- Множество параметри са разделени от разделители

```
http://example.com/path/to/page?name=ferret&color=purple
```

# Кодиране на URL

- URL адресите са кодирани съгласно RFC 1738:

- Нерезервирани URL символи - нямат специално значение

[0-9a-zA-Z]

- Запазени знаци за URL - могат да имат специално значение в URL адреса

! \* ' ( ) ; : @ & = + \$ / , ? # [ ]

- Запазените символи се избягват чрез процентно кодиране

%[character hex code]

- Пространството е кодирано като "+" или "% 20"

# Кодиране на URL - Примери

- Всички останали знаци се избягват от %

Знак	Кодиране на URL
space	%20
щ	%D1%89
"	%22
#	%23
\$	%24
%	%25
&	%26

Всеки знак се преобразува в стойността му ASCII, представена като шестнадесетични цифри

Иван-愛-mon



%D0%98%D0%B2%D0%B0%D0%BD-%E7%88%B1-mon

# Валидни и Неваидни URL Адреси - Примери

- Валидни URL адреси:

`http://www.google.bg/search?sourceid=navclient&ie=UTF-8&rlz=1T4GGLL_enBG369BG369&q=http+get+vs+post`

`http://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80%D0%BD%D0%B0_%D0%B0%D0%BA%D0%B0%D0%B4%D0%B5%D0%BC%D0%B8%D1%8F`

- Неваидни URL адреси:

`http://www.google.bg/search?&q=C# .NET 4.0`

Би трябвало да е:  
C%23+.NET+4.0

`http://www.google.bg/search?&q=бира`

Би трябвало да е: %D0%B1  
%D0%B8%D1%80%D0%B0



MIME u munoBe meguu



# Какво е MIME?

- Multi-Purpose Internet Mail Extensions

- Интернет стандарт за кодиране на ресурси
- Първоначално разработен за прикачени файлове по имейл
- Използва се в много интернет протоколи като HTTP и SMTP

# Концепции на MIME

- **Content-Type** - тип медия на съдържанието на съобщението
  - `text/html`, `image/gif`, `application/pdf`
- **Content-Disposition** - определя стила на представяне
  - `attachment; filename=logo.jpg`
- Многочастни съобщения - множество ресурси в един документ

# Общи MIME типове медии

MIME Тип / Подтип	Описание
application/json	JSON данни
image/png	PNG снимка
image/gif	GIF снимка
text/html	HTML
text/plain	Text
text/xml	XML
video/mp4	MP4 видео
application/pdf	PDF документ



# HTTP заявки

Какво е HTTP заявка?

# HTTP съобщение за заявка

- Съобщение за заявка, изпратено от клиент, се състои от:
  - HTTP ред за заявка
    - Метод на заявка (GET / POST / PUT / DELETE / ...)
    - URI (URL)
    - Версия на протокола
  - HTTP хедър
    - Допълнителни параметри
  - Тяло на HTTP заявка – незагължителни данни, напр. публикувани формулярни полета

**<method> <resource> HTTP/<version>**

**<headers>**

*(empty line)*

**<body>**

# GET метод за заявка - Пример

```
<form method="get">  
  Name: <input type="text" name="name" />  
  Age: <input type="text" name="age" />  
  <input type="submit" />  
</form>
```

HTTP ред на заявката

GET /HTTP/1.1

Host: localhost

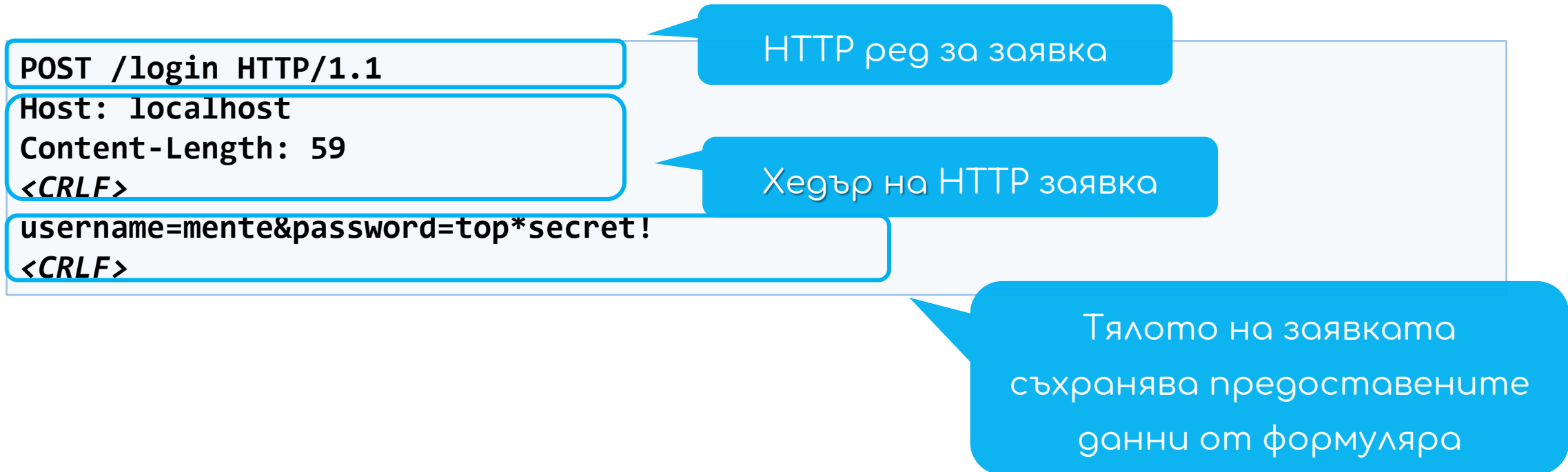
<CRLF>

Хедър на HTTP заявка

Тялото на заявката е празно

# POST метод за заявка - Пример

- Методът POST прехвърля данни в HTTP тялото
- POST може да изпраща текстови и двоични данни, напр. качване на файлове





# HTTP отговор

Какво е HTTP отговор?



# HTTP съобщение за отговор

- Съобщението за отговор, изпратено от HTTP сървър, се състои от:
  - HTTP ред на състоянието на отговора
    - Версия на протокола
    - Код на състоянието
    - Текст на състоянието
  - Хедър
    - Предоставя метаданни за върнатия ресурс
  - Тяло на отговора
    - Съдържанието на HTTP отговора (данни)

```
HTTP/<version> <status code> <status text>  
<headers>  
<CRLF>  
<response body – the requested resource>
```

# HTTP отговор - Пример

- Пример за HTTP отговор от веб сървъра:

HTTP/1.1 200 OK

HTTP рег на състоянието на отговора

Date: Fri, 17 Jul 2010 16:09:18 GMT+2  
Server: Apache/2.2.14 (Linux)  
Accept-Ranges: bytes  
Content-Length: 84  
Content-Type: text/html

Хедър на HTTP  
отговора

<CRLF>

<html>  
 <head><title>Test</title></head>  
 <body>Test HTML page.</body>  
</html>

Тяло на HTTP  
отговора

# HTTP кодове за отговор

- HTTP класове кодове за отговор
  - 1xx: информационен (e.g., **"100 Continue"**)
  - 2xx: успешен (e.g., **"200 OK"**, **"201 Created"**)
  - 3xx: пренасочване (e.g., **"304 Not Modified"**, **"301 Moved Permanently"**, **"302 Found"**)
  - 4xx: клиентска грешка (e.g., **"400 Bad Request"**, **"404 Not Found"**, **"401 Unauthorized"**, **"409 Conflict"**)
  - 5xx: грешка в сървъра
    - (e.g., **"500 Internal Server Error"**, **"503 Service Unavailable"**)

# HTTP отговор - Пример

- Пример за HTTP отговор с резултат от грешка:

HTTP/1.1 404 Not Found

Date: Fri, 17 Nov 2014 16:09:18 GMT+2

Server: Apache/2.2.14 (Linux)

Connection: close

Content-Type: text/html

<CRLE>

<HTML><HEAD><TITLE>404 Not Found</TITLE></HEAD>

<BODY>

<H1>Not Found</H1>

The requested URL /img/logo.gif was not found on this server.<P>

<HR><ADDRESS>Apache/2.2.14 Server at Port 80</ADDRESS>

</BODY></HTML>

HTTP ред на състоянието на  
отговора

Хедър на HTTP отговора

HTTP  
тяло на  
отговора

# Пренасочване на браузъра

- HTTP GET заявява преместване на URL адрес:

```
GET / HTTP/1.1  
Host: http://mon.org  
User-Agent: Gecko/20100115 Firefox/3.6  
<CRLF>
```

- Следният HTTP отговор (301 Moved Permanently) казва на браузъра да пуска друг URL адрес:

```
HTTP/1.1 301 Moved Permanently  
Location: http://mon.bg  
...
```

# Content-Type u Disposition

- С хедъра Content-Type в отговора сървърът посочва как трябва да се обработва изхода
- Примери:

UTF-8 кодирана HTML страница. Ще се покаже в браузъра.

**Content-Type: text/html; charset=utf-8**

**Content-Type: application/pdf**

**Content-Disposition: attachment; filename="Report-April-2016.pdf"**

Това ще изтегли PDF файл с името  
Report-April-2016.pdf

# Content-Disposition. Пример [1/2]

**Content-Type: text/plain**

**Content-Length: 19**

**Content-Disposition: inline filename=example.txt**



This is inline view

# Content-Disposition. Пример [2/2]

**Content-Type: text/plain**

**Content-Length: 19**

**Content-Disposition: attachment; filename=example.txt**



Opening example.txt

You have chosen to open:



**example.txt**

which is: TXT file (19 bytes)

from: http://localhost





Национална програма  
"Обучение за ИТ умения и кариера"  
<https://it-kariera.mon.bg>

Министерството на  
образованието и науката  
<https://www.mon.bg>



Документът е разработен за нуждите на Национална програма "Обучение за ИТ умения и кариера" на Министерството на образованието и науката (МОН) и се разпространява под свободен лиценз CC-BY-NC-SA (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).