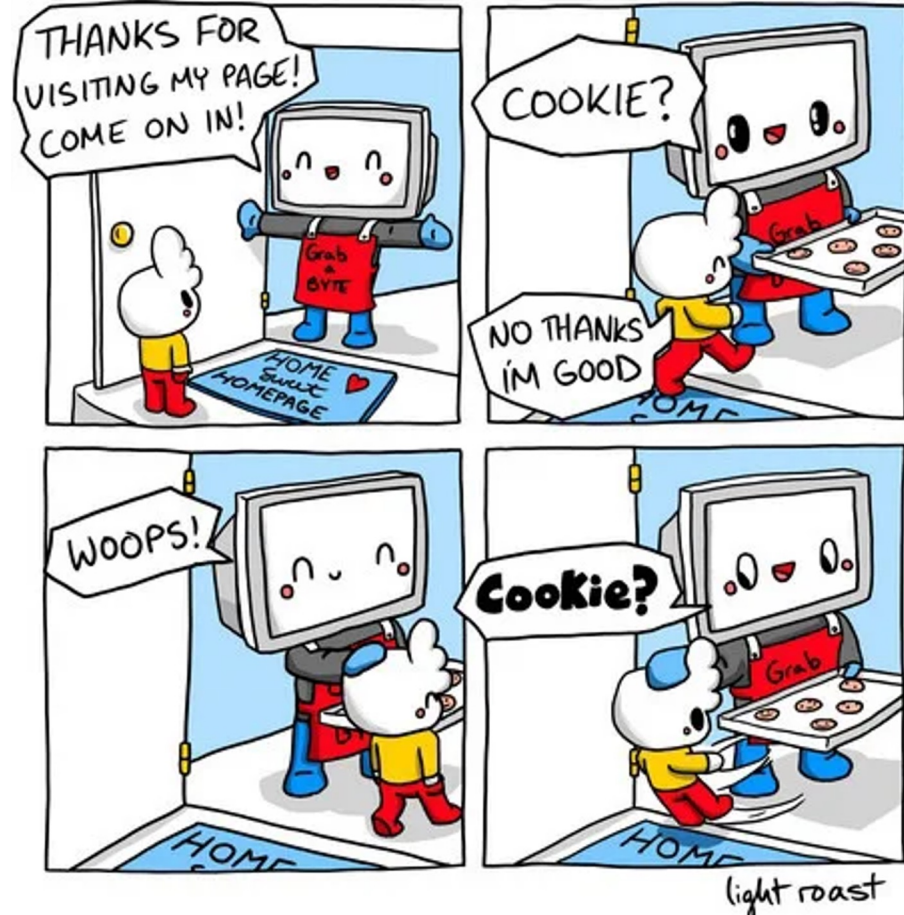


# Web Technologies

## Web programming (II)




HTTP cookies

Web sessions



Prof. Sabin Corneliu Buraga – [profs.info.uaic.ro/sabin.buraga/](http://profs.info.uaic.ro/sabin.buraga/)

Assoc. Prof. Andrei Panu – [profs.info.uaic.ro/andrei.panu/](http://profs.info.uaic.ro/andrei.panu/)

-  how can data sent from the server, by the Web application, be stored at the browser's level
-  how can we identify successive requests sent by the same instance of the client
-  alternatives to Web cookies and sessions



Is there a way to – temporarily – store on front-end  
(browser) data transmitted by the back-end of a  
Web application?

# cookies

A script running on a Web server can put data on the client-computer via the user's Web browser

subsequently, the navigator will return that data to the same script available on the same server

# cookies

A (quasi-)persistent way to store data on the machine of a Web client in order to be further accessed by a program running on a server

[developer.mozilla.org/docs/Web/HTTP/Cookies](https://developer.mozilla.org/docs/Web/HTTP/Cookies)

also consult [Cookiepedia: cookiepedia.co.uk](http://Cookiepedia: cookiepedia.co.uk)

# cookies: usages

Storing user preferences

typical examples:

options regarding interaction – visual theme  
(e.g., chromatics, graphic arrangement), language  
preferences, geographical location, interests on shopping

...

# cookies: usages

Automatic form completion

using values previously entered by the user  
in certain fields

# cookies: usages

Monitoring the access to a Web resource

aspect of interest:

**Web analytics**

collecting information about clients

(hardware platform, browser, screen resolution, etc.)

# cookies: usages

Monitoring the access to a Web resource

aspect of interest:

**user tracking**

monitoring the user's behavior

▶ **Do Not Track** initiative

[www.eff.org/issues/do-not-track](http://www.eff.org/issues/do-not-track)

# cookies: usages

Storing authentication info

e.g., keeping data about the user account  
in the e-commerce context

# cookies: usages

Transaction status

example: current state of the virtual shopping cart  
provided by an e-shop application

# cookies: usages

Web session management

# cookies: types

## Persistent cookies

not destroyed when the Web browser closes

kept into a file – on the client-side

lifetime set by the cookie creator

# cookies: types

Non-persistent (volatile) cookies

disappear when the browser is closed

# cookies

a cookie can be considered as a variable

its value is transferred via HTTP  
between the Web server (back-end application)  
and the client (browser)

the size of a cookie cannot exceed 4KB

# cookies

A cookie can be considered as a variable

name=value

the value is an URL encoded string

example:

Europe/Bucharest → Europe%2FBucharest

# cookies

Data about a cookie is received by the browser  
a list of cookies for each server (domain)

# cookies

Data about a cookie is received by the browser  
a list of cookies for each server (domain)

1 <sup>st</sup> party cookies	created by the Web application, belonging to the visited primary domain– (non)persistent [essential or not]
3 <sup>rd</sup> party cookies	created by external applications–(non)persistent [can cause security problems – privacy]
session cookies	specific to the current Web session – nonpersistent [usually, essential]

# cookies

A cookie is sent to a client  
by using the **Set-Cookie**  
header field of an HTTP response message

# cookies

```
Set-Cookie: name=value; expires=date; path=path;  
            domain=Internet-domain; secure
```

# cookies

```
Set-Cookie: name=value; expires=date; path=path;  
            domain=Internet-domain; secure
```

**expires** – indicates the date and time when the cookie will expire (the Web client should destroy expired cookies)

# cookies

```
Set-Cookie: name=value; expires=date; path=path;  
domain=Internet-domain; secure
```

**domain** – signifies the symbolic name of the Web server that generated the cookie

# cookies

```
Set-Cookie: name=value; expires=date; path=path;  
domain=Internet-domain; secure
```

`path` – specifies a subset of URLs  
from the cookie's domain

distinguishes multiple applications  
existing on the same server

# cookies

```
Set-Cookie: name=value; expires=date; path=path;  
            domain=Internet-domain; secure
```

**secure** – indicates that the cookie will be sent back to the server only if the communication channel is secure (via HTTPS)

# cookies

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
_docs-events	06d73ff6-89e7-4c23-bb37-efcd84ac4d45	docs.github.com	/	Thu, 25 Feb 2027 06:48:28 GMT	48	false	true	Strict	Wed, 25 Feb 2026 06:48:28
_octo	GH1.1.425965642.1771491077	.github.com	/	Fri, 19 Feb 2027 08:53:05 GMT	31	false	true	Lax	Wed, 25 Feb 2026 06:48:27
cpu_bucket	xlg	.github.com	/	Session	13	false	true	Lax	Wed, 25 Feb 2026 06:48:27
GHCC	Required:1-Analytics:0-SocialMedia:0-Advertising:0	.github.com	/	Session	54	false	true		Wed, 25 Feb 2026 06:48:28
logged_in	no	.github.com	/	Fri, 19 Feb 2027 08:53:05 GMT	11	true	true	Lax	Wed, 25 Feb 2026 06:48:27
preferred_color_mode	light	.github.com	/	Session	25	false	true	Lax	Wed, 25 Feb 2026 06:48:27
tz	Europe%2FBucharest	.github.com	/	Session	20	false	true	Lax	Wed, 25 Feb 2026 06:48:27

```
▼ tz:"Europe%2FBucharest"  
  Created:"Wed, 25 Feb 2026 06:37:24 GMT"  
  Domain:".github.com"  
  Expires / Max-Age:"Session"  
  HostOnly:false  
  HttpOnly:false  
  Last Accessed:"Wed, 25 Feb 2026 06:48:27 GMT"  
  Path:"/"  
  SameSite:"Lax"  
  Secure:true  
  Size:20  
  Updated:"Wed, 25 Feb 2026 06:37:24 GMT"
```

inspecting cookies stored by the Web browser  
for each domain

# cookies

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
_docs-events	06d73ff6-89e7-4c23-bb37-efcd84ac4d45	docs.github.com	/	Thu, 25 Feb 2027 06:48:28 GMT	48	false	true	Strict	Wed, 25 Feb 2026 06:48:28
_octo	GH1.1.425965642.1771491077	.github.com	/	Fri, 19 Feb 2027 08:53:05 GMT	31	false	true	Lax	Wed, 25 Feb 2026 06:48:27
cpu_bucket	xlg	.github.com	/	Session	13	false	true	Lax	Wed, 25 Feb 2026 06:48:27
GHCC	Required:1-Analytics:0-SocialMedia:0-Advertising:0	.github.com	/	Session	54	false	true		Wed, 25 Feb 2026 06:48:28
logged_in	no	.github.com	/	Fri, 19 Feb 2027 08:53:05 GMT	11	true	true	Lax	Wed, 25 Feb 2026 06:48:27
preferred_color_mode	light	.github.com	/	Session	25	false	true	Lax	Wed, 25 Feb 2026 06:48:27
tz	Europe%2FBucharest	.github.com	/	Session	20	false	true	Lax	Wed, 25 Feb 2026 06:48:27

inspecting cookies stored by the Web browser  
for each domain

**httpOnly: true** – indicates that the value of a cookie can be obtained only from a data transfer through HTTP



the cookie cannot be accessed by a program executed on client side (browser) – [www.owasp.org/index.php/HttpOnly](http://www.owasp.org/index.php/HttpOnly)

vp	1450x160	.emag.ro	/	Tue, 10 Mar 2026 12:39:07 GMT	10	false	false	
token1	%222506852746793280054-03a856cb-a709-4b8d-...	.emag.ro	/	Wed, 18 Jun 2025 12:40:18 GMT	92	false	false	
sr	2112x1148	.emag.ro	/	Tue, 10 Mar 2026 12:39:07 GMT	11	false	false	
site_version_11	not_mobile	.emag.ro	/	Fri, 27 Mar 2026 06:53:19 GMT	25	false	false	
ltuid	2506845546157962524	.emag.ro	/	Tue, 14 Apr 2026 12:39:06 GMT	24	false	false	
loginTooltipShown	1	.emag.ro	/	Tue, 10 Mar 2026 12:39:07 GMT	18	false	false	
listingResetView	1	www.emag.ro	/	Tue, 10 Mar 2026 13:05:52 GMT	17	false	false	
listingPerPage	60	www.emag.ro	/	Fri, 19 Feb 2027 09:09:25 GMT	16	false	false	
listingDisplayId	2	www.emag.ro	/	Fri, 19 Feb 2027 09:08:01 GMT	17	false	false	
EMAGVISITOR	a%3A1%3A%7Bs%3A7%3A%22user_id%22%3Bi%3...	.emag.ro	/	Fri, 26 Mar 2027 09:08:03 GMT	75	false	false	
EMAGUUID	1741610346-7010970270-21083.024	.emag.ro	/	Tue, 14 Apr 2026 12:39:06 GMT	39	false	false	
EMAGROSESSID	9584b2ce41e987ab11e62a0dc7b8ae61	.emag.ro	/	Wed, 25 Feb 2026 18:53:19 GMT	44	true	true	Lax
eab_allocation		.emag.ro	/	Wed, 25 Feb 2026 06:54:19 GMT	14	false	false	

inspecting cookies stored by the Web browser  
for each domain

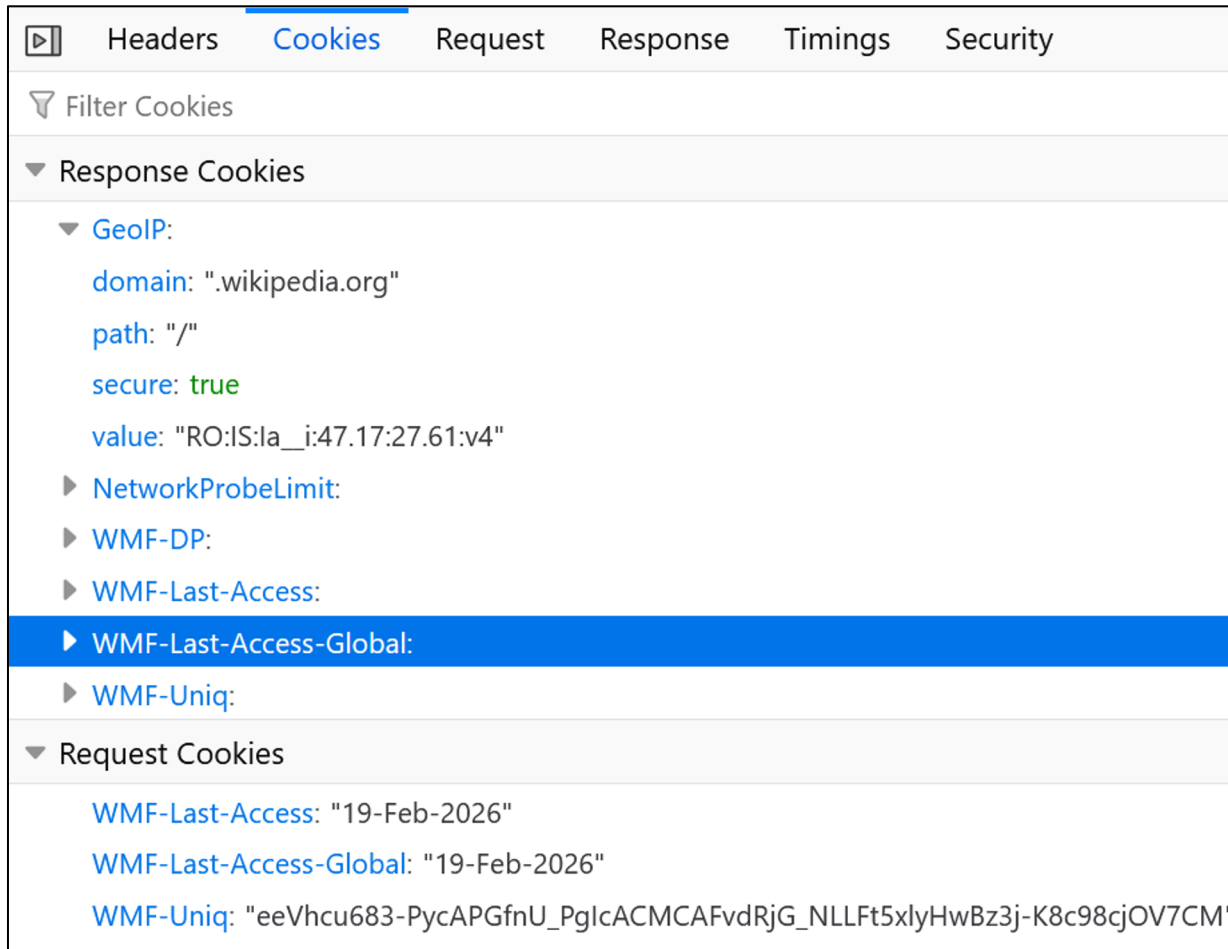
**sameSite** – specifies whether the cookie’s value will be passed across requests between different websites (cross-site requests)  
possible values: Strict / Lax / None

# cookies

A cookie is transmitted back from the client to the Web server only if it satisfies all validity conditions

domain, (virtual) path, expiration time, and communication channel security are matching

# inspecting the cookies stored by the browser for each domain specific to a Web application



cookie values sent back to the web server  
on the next HTTP request

# cookies

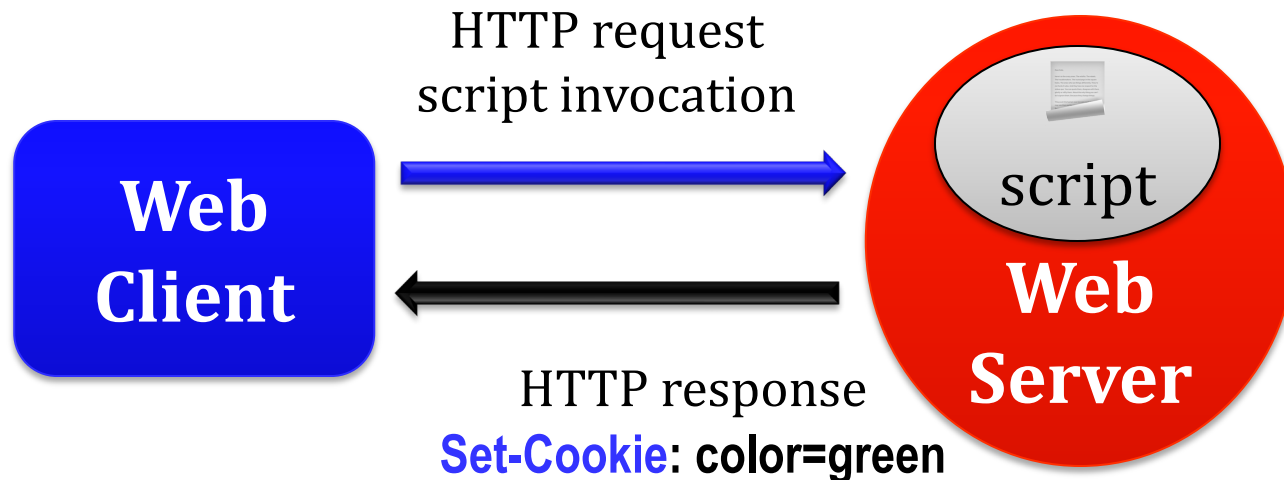
the server will receive, in the header of an HTTP request message, the following:

**Cookie: name1=value1; name2=value2...**

the list of cookies which satisfy the validity conditions

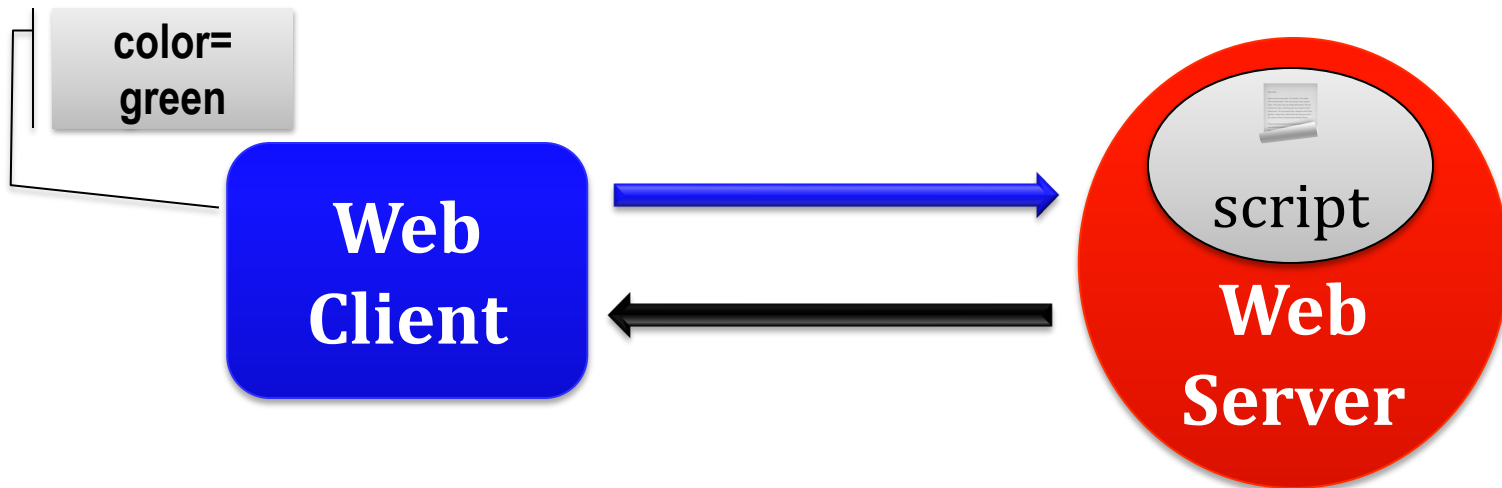
# cookies

A script invocation leads to returning a representation + placing various cookies



# cookies

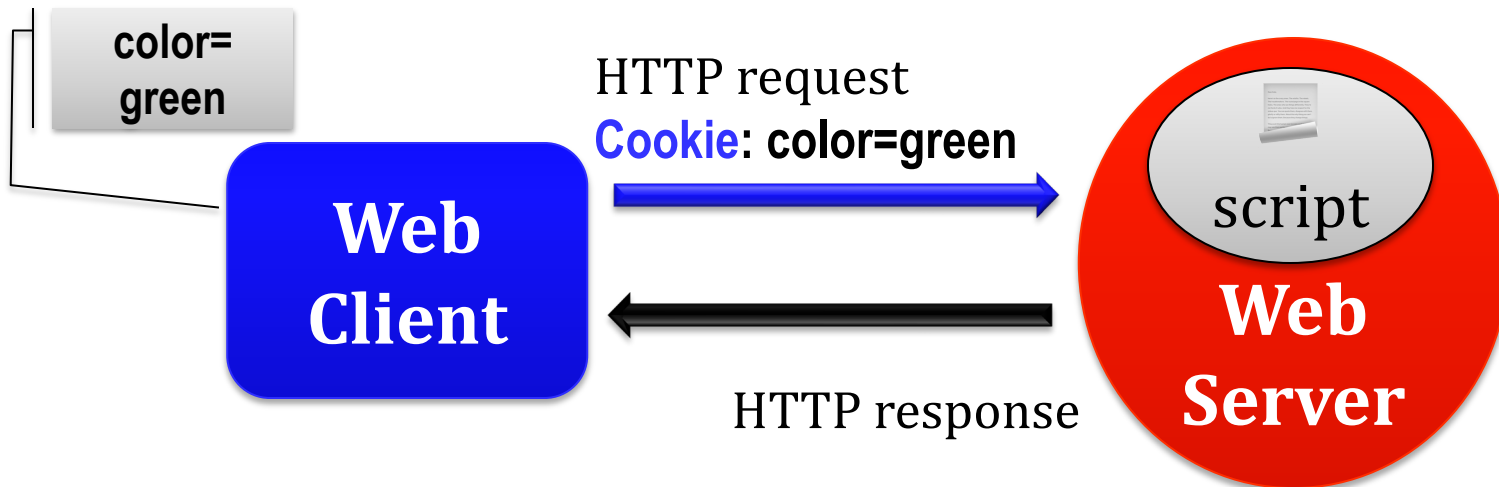
Cookies – persistent or not –  
are processed and stored by the browser



persistent cookies are stored in files or databases (SQLite)

# cookies

Next access to the script is made by transmitting the cookies to the server according to the validity conditions



# cookies: consulting

Cookies reside in the header field  
of an HTTP message

**HTTP\_COOKIE**

# cookies: expiration

To remove a cookie,  
the value and time are canceled  
eventually, the other attributes of the cookie

# cookies

Other information of interest is available in  
RFC 6265

*HTTP State Management Mechanism*

[tools.ietf.org/html/rfc6265](https://tools.ietf.org/html/rfc6265)

(instead of a) break





How can we identify successive requests  
expressed by the same client instance?



HTTP is a **stateless** protocol

HTTP is a **stateless** protocol

does not offer information if certain successive requests are received from the same client (from the same instance of a Web browser)

# necessity

Preserving certain data for a sequence of related HTTP messages (requests/responses)

examples:

shopping cart status

multi-step Web forms

content pagination

user authentication state

etc.

# sessions

Each visitor of a website will have associated a unique identifier – **session ID (SID)**

usually, stored by a cookie

(having an implicit name specific to the Web application server/framework – e.g., ASP.NET\_SessionId, JSESSIONID, PHPSESSID, session-id, \_wp\_session)

# sessions

Each visitor of a website will have associated a unique identifier – **session ID (SID)**

usually, stored by a cookie  
(e.g., ASP.NET\_SessionId, JSESSIONID,  
PHPSESSID, session-id, \_wp\_session)

alternative – not recommended 

SID propagated via an URL

# sessions

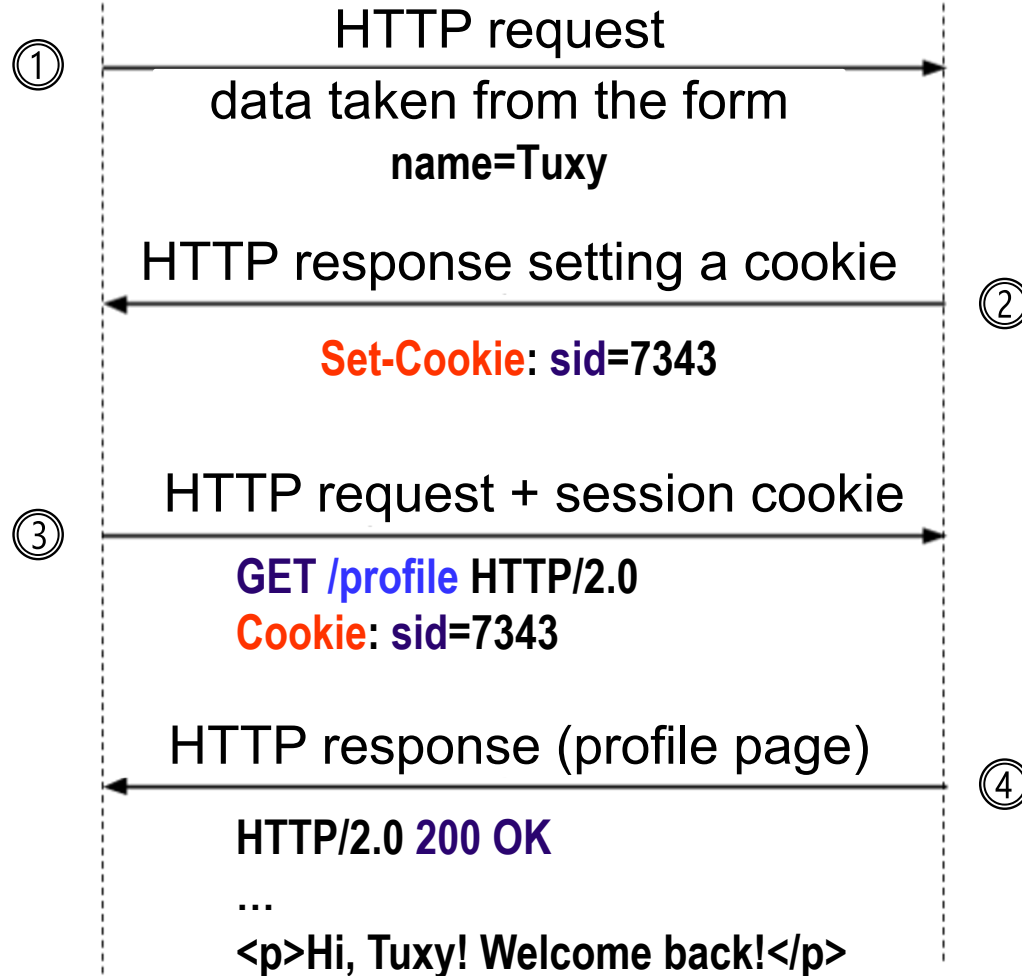
Each visitor of a website will have associated a unique identifier – **session ID (SID)**

in this way, consecutive visits (requests) made by the same user can be identified



Web client (browser)

Web server  
(daemon)



establishing a Web session using a cookie

# sessions

Various variables can be attached to a session

their values will be kept (stored) between consecutive  
– e.g., related – requests from the same instance  
of a Web client (browser)

# sessions

A session could be implicitly (automatically) or explicitly (manually, by programmer) registered, depending on the Web application server or the default configuration

# sessions

A session could be implicitly (automatically) or explicitly (manually, by programmer) registered, depending on the Web application server or the default configuration

Web session info is persistently stored on the server by using non-relational database systems – e.g., DynamoDB, Memcached, Redis,... – or, in most cases, files

**POST /?\_task=login HTTP/1.1**

**Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,\*/\*;q=0.8**

**Accept-Encoding: gzip, deflate, br, zstd**

**Accept-Language: en-US,en;q=0.9**

**Connection: keep-alive**

**Cookie: language=en\_US**

**Cookie: roundcube\_sessid=3hg...71m6**

**DNT: 1**

**Host: webmail.info.uaic.ro**

**Referer: https://webmail.info.uaic.ro/**

**Upgrade-Insecure-Requests: 1**

**User-Agent: Mozilla/5.0 ... Gecko/20100101 Firefox/148.0**

user authentication by using the POST method  
(already existing cookies are transmitted)

**HTTP/1.1 302 Found**

**Cache-Control:** private, no-cache, no-store, must-revalidate...

**Connection:** Keep-Alive

**Content-Length:** 0

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

**Date:** Wed, 25 Feb 2026 07:23:36 GMT

**Expires:** Wed, 25 Feb 2026 07:23:36 GMT

**Keep-Alive:** timeout=5, max=100

**Last-Modified:** Wed, 25 Feb 2026 07:23:36 GMT

**Location:** [/?\\_task=mail&\\_token=sWFnOZ...W5le67yz](/?_task=mail&_token=sWFnOZ...W5le67yz)

**Server:** Apache

**Set-Cookie:** **roundcube\_sessid=vn4...2uv2; path=/; secure; HttpOnly**  
**roundcube\_sessauth=S9...2c71; path=/; secure; HttpOnly**

**<!DOCTYPE html>**

...

redirection  
after  
authentication

HTTP response

a Web session-related cookie is set

# sessions: programming

In the case of CGI, session management must be entirely implemented by the programmer

there is no standard way for Web session management



What alternatives are there  
to Web cookies and sessions?

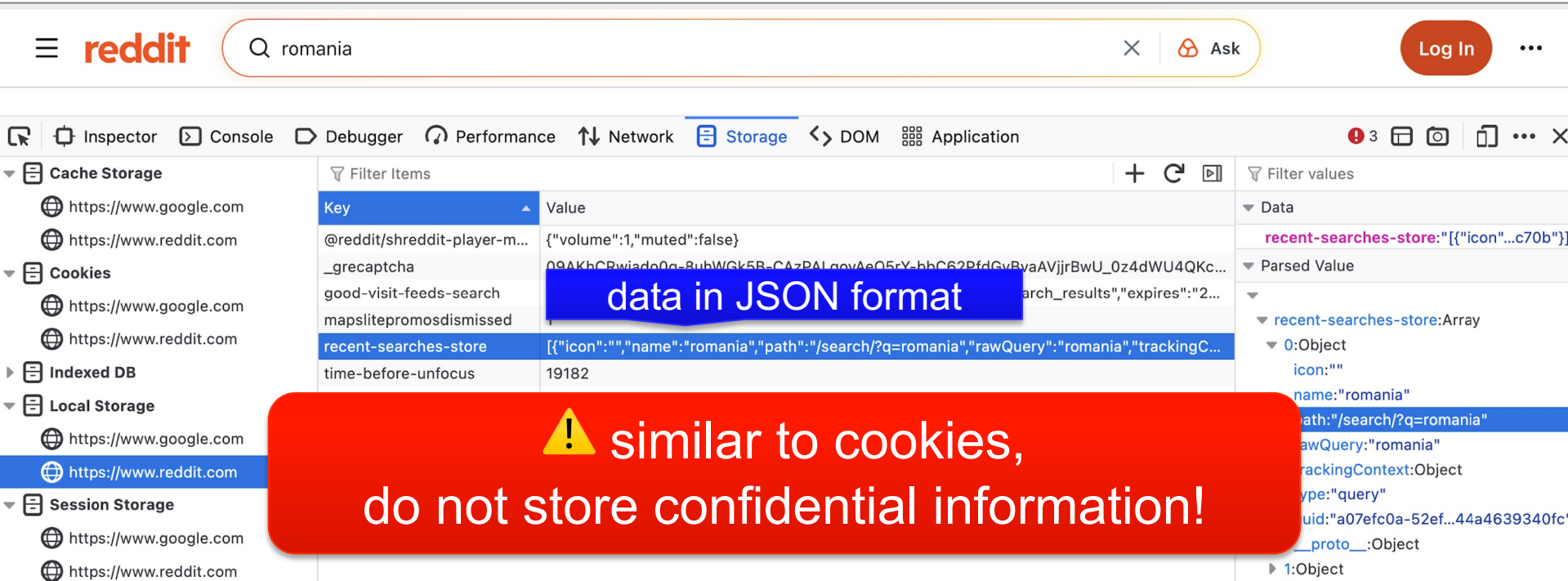
# alternatives

## Web Storage

browser-level storage for lists of  
key—value pairs  
via **sessionStorage** and **localStorage** attributes

see *HTML Living Standard* (25 Feb. 2026) specification  
[html.spec.whatwg.org/multipage/webstorage.html](http://html.spec.whatwg.org/multipage/webstorage.html)

# alternatives



The screenshot shows the browser's developer tools with the Storage tab selected. The 'recent-searches-store' item is highlighted, showing its key and value in JSON format. A red callout box with a warning icon and text is overlaid on the screenshot.

Key	Value
@reddit/shreddit-player-m...	{"volume":1,"muted":false}
_grecaptcha	09AKhCBwiado0g-8ubWGk5B-CAzPAL...devAeQ5rY-bbC62PfdGvBvaAVjrrBwU_0z4dWU4QKc...
good-visit-feeds-search	...
mapslitepromosdismissed	...
recent-searches-store	[{"icon":"","name":"romania","path":"/search/?q=romania","rawQuery":"romania","trackingC...}
time-before-unfocus	19182

**data in JSON format**

**! similar to cookies, do not store confidential information!**

[developer.mozilla.org/Web/API/Web\\_Storage\\_API](https://developer.mozilla.org/Web/API/Web_Storage_API)

# alternative

## Indexed DB

object-level data storage via an asynchronous API  
provided by the Web browser

**Indexed Database API 2.0** – W3C Recommendation, 2018

**Indexed Database API 3.0** – work in progress, August 2025

[www.w3.org/TR/IndexedDB/](http://www.w3.org/TR/IndexedDB/)

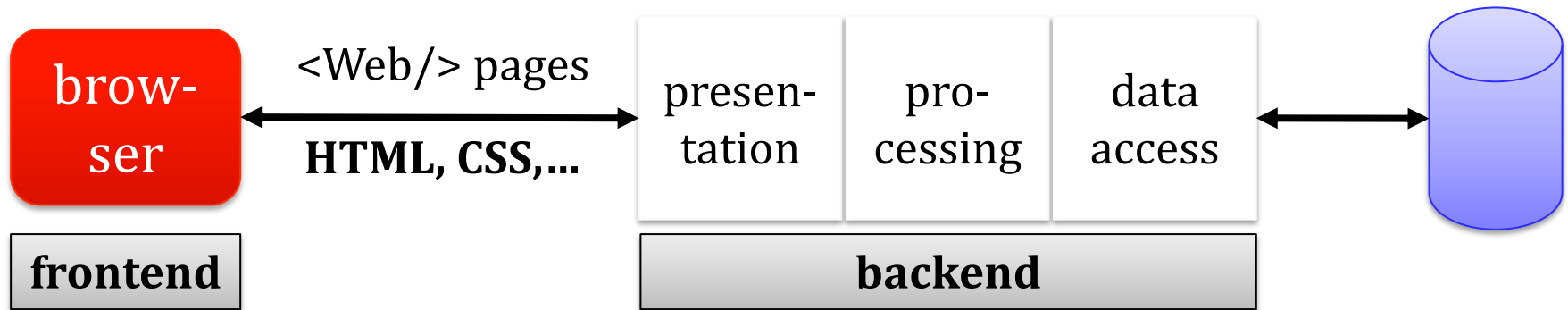
# summary



Web cookies and sessions  
many thanks to Ciprian Amariei, MSc.

dumb client

fat server



next episode: **Web programming**

Web application servers, Web application architecture