

# Web programiranje

***ITAcademy***

# Kolačići (Cookies)

- Web browser ima osobinu da na lokalnom disku sačuva određenu informaciju karakterističnu za neki domen
- Na primer, ako postoji domen koji se zove: ***mojdomen.com***, browser može sačuvati neku promenljivu **x** koja će imati vrednost **10** i povezati je sa domenom **mojdomen.com**.
- Da bi ovo bilo ostvareno, browser koristi **kolačiće**.

# Kolačići (Cookies)

- Kako browser zna da treba da napravi kolačić?
  - Server (serverska skripta) obaveštava browser da treba da napravi kolačić.
  - Ovo obaveštenje stiže kroz jedno od zaglavlja HTTP odgovora.
  - Zaglavlje u kome se browseru daje direktiva da napravi kolačić izgleda ovako:

## **Set-Cookie: x=10**

- Kada browser dobije ovo zaglavlje, on kreira kolačić koji je isključivo vezan za domen sa koga je stiglo zaglavlje

# Kolačići (Cookies)

- Kada je browser napravio kolačić za određeni domen, svaki put, prilikom otvaranja neke strane sa tog domena, browser automatski šalje i kreirani kolačić
- Browser šalje kolačić takođe kroz HTTP zaglavlje
- HTTP zaglavlje u kome se nalazi kolačić izgleda ovako:

**Cookie:** x=10

- Browser uvek šalje kolačiće samo domenu koji ih je napravio, **ali ne smemo se nikada u potpunosti pouzdati u ovu činjenicu. Kolačići su česta meta bezbednosnih napada, jer sadrže i informacije koje jedinstveno identifikuju korisnike.**

# Kolačići (Cookies) i PHP

- Zbog čega su nam kolačići važni za PHP i web aplikaciju uopšte?
- Pre svega, zato što omogućavaju da očuvamo neku vrednost između requesta.
- Na primer: ako napravimo neku promenljivu u php-u i damo joj vrednost

**\$x = 10;**

- Ni jedna druga strana neće biti svesna postojanja te promenljive
  - Promenljiva će prilikom svakog sledećeg otvaranja strane, uvek imati vrednost 10. Jer će se uvek generisati iznova.
- Ako bi promenljivu smestili u kolačić, bila bi dostupna svim stranama, jer browser svaki put šalje kolačić, kada god otvori neku stranu sa domena za koji je kolačić registrovan

# Kolačići (Cookies) i PHP

- Postoji mnogo načina da napravimo kolačić u php-u
  - Prvi je da ručno unesemo odgovarajuće zaglavlje. Na primer:  
***header("Set-Cookie: y=25");***
  - Drugi način je da upotrebimo php funkciju koja služi specijalno za tu namenu:  
***setcookie("x",10);***
- U oba slučaja, biće generisano zaglavlje Set-Cookie, na osnovu koga će browser znati da treba da napravi kolačić

# Kolačići (Cookies) i PHP – rok trajanja

- Kako je moguće da na nekom sajtu ubacimo neki artikal u korpu, a zatim, nakon nekog vremena posetimo taj isti sajt, i na njemu nas sačeka korpa sa ubačenim artiklom? “Krivac” za ovo su kolačići.
- Ali kolačići podrazumevano nemaju mogućnost da sačuvaju informaciju toliko dugo. Kolačići podrazumevano čuvaju informaciju sve dok je browser aktivan. Kada isključimo browser, obrisaćemo i podrazumevane kolačiće
- Da bi kolačići “opstali” i nakon restarta browsera, moramo naznačiti i njihov **rok trajanja**.

# Kolačići (Cookies) i PHP – rok trajanja

- Rok trajanja kolačića predstavlja se parametrom **expires**:

**Set-Cookie: x=10; expires=Tue, 04-Mar-2014 10:51:37 GMT**

- Ovaj parametar mora biti u UTC formatu, pa je zbog toga, ako hoćemo da definišemo vreme trajanja kolačića, najbolje upotrebiti php funkciju, jer je ona u stanju da timestamp format (kojim lako manipuliramo), pretvori u formu koja odgovara onoj koja se očekuje u kolačiću
- Sledeći kolačić će isteći minut nakon kreiranja  
**setcookie("x",10,time()+60);**



# Kolačići (Cookies) i PHP – preuzimanje

- Da bi preuzeli kolačić u php-u, možemo opet koristiti različite varijante
- Obzirom da je kolačić transportovan putem zaglavlja, možemo ga pročitati čitanjem zaglavlja:

```
$hdrs = getallheaders();
```

```
$cookies = $hdrs['Cookie'];
```


- Problem kod ovakvog pristupa je što ćemo dobiti sve kolačiće za određeni domen, i to u sirovom formatu. Na primer:

```
x=10; y=15; y=20
```

- PHP prilikom dobijanja svakog zahteva proverava zaglavlje sa kolačićima i raspakuje ih u čitljivu formu (asocijativni niz), a zatim ih smešta u dva niza: **`$_COOKIE`** i **`$_REQUEST`** (može biti isključeno u php.ini fajlu)

# Kolačići (Cookies) i PHP – preuzimanje

- Prilikom upravljanja kolačićima mora se obratiti pažnja na minimum tri važne stvari:
- Kolačići moraju biti definisani pre bilo kakvog izlaza (kao i sve ostalo što uključuje modifikaciju zaglavlja). Ukoliko pokušamo da napravimo kolačić, a prethodno je izvršeno neko emitovanje na izlaz, dobijamo sledeću grešku:

 <b>Warning:</b> Cannot modify header information - headers already sent by (output started at D:\programi\wamp\www\index.php:1) in D:\programi\wamp\www\index.php on line 2				
Call Stack				
#	Time	Memory	Function	Location
1	0.0003	326096	{main}()	..\index.php:0
2	0.0003	326320	<a href="#">setcookie ()</a>	..\index.php:2

Greška može biti uzrokovana samo jednim karakterom:

## Ispravan kod

```
<?php  
setcookie("my_cookie", "HEY!");
```

## Prazan prostor na početku uzrokuje grešku

```
<?php  
setcookie("my_cookie", "HEY!");
```

# Kolačići (Cookies) i PHP – preuzimanje

2. Kolačić ne možemo koristiti odmah nakon kreiranja. On će biti unet u superglobalnu promenljivu tek nakon sledećeg zahteva.

Ovo nije ispravno:

```
<?php
    setcookie("username", "Peter");
    echo "Hello " . $_COOKIE['username'];
```

3. Kolačići se lako čitaju, i u njima **nikada** ne čuvamo **ništa važno** a pri tom lako razumljivo.


# Primer: prepoznavanje jezika pomoću kolačića

- U primeru, biće proveren jezički kolačić. Na osnovu ovog kolačića, korisniku će biti prikazana pozdravna poruka na odgovarajućem jeziku.
- Korisnik će moći da promeni jezik pomoću hiperlinka
- Lista jezičkih paketa biće hard kodirana u nizu.

# Primer: prepoznavanje jezika pomoću kolačića

<http://pastie.org/8856359>

```
<?php
    $messages = array(
        "en"=>"Welcome",
        "sr"=>"Dobro dosli"
    );
    $language = isset($_COOKIE['lng'])?$_COOKIE['lng']:"en";
    if(isset($_GET['lng'])){
        $language = $_GET['lng'];
        setcookie("lng",$language);
    }
    echo $messages[$language];
?><br />
<a href="?lng=en">english</a> &nbsp;&nbsp;&nbsp;<a href="?lng=sr">serbian</a>
```

← → ↻  localhost/?lng=en

Welcome  
[english](#) [serbian](#)