

PRIMERI PROGRAMIRANJA INTERNET APLIKACIJA

ULFETA MAROVAC

DRŽAVNI UNIVERZITET U NOVOM PAZARU
2022

Naslov:

PRIMERI PROGRAMIRANJA INTERNET APLIKACIJA

Autor: Doc. dr Ulfeta Marovac

Mesto i godina izdanja: Novi Pazar, 2022

Izdavač:

Državni univerzitet u Novom Pazaru

Vuka Karadžića br. 9

36300 Novi Pazar

<http://www.np.ac.rs>

Recenzenti:

Prof. dr Bratislav Mirić

Prof. dr Edin Dolićanin

Doc. dr Irfan Fetahović

ISBN: 978-86-81506-14-1

Štamparija: SZR "Fami grafika"

Mesto štampanja: Novi Pazar

Tiraž: 20

CIP - Katalogizacija u publikaciji
Narodna biblioteka Srbije, Beograd

004.42:004.738.5(075.8)(076)

004.438PHP(075.8)(076)

MAROVAC, Ulfeta, 1980-

Primeri programiranja internet aplikacija / Ulfeta Marovac. - Novi Pazar : Državni univerzitet, 2022 (Novi Pazar : Fami grafika). - 129 str. : ilustr. ; 26 cm

Tiraž 20. - Bibliografija: str. 129.

ISBN 978-86-81506-14-1

a) Web prezentacije -- Programiranje -- Vežbe b) Programski jezik "PHP" -- Vežbe
COBISS.SR-ID 69280009

*PRIMERI PROGRAMIRANJA INTERNET
APLIKACIJA*

praktikum

Mojim studentima

PREDGOVOR

Veći deo ovog praktikuma čine alati i metode koje se obrađuju u okviru predmeta Programiranje internet aplikacija na studijskom programu Softversko inženjerstvo, Državnog univerziteta u Novom Pazaru. Cilj praktikuma je da objedini različite alate i metode programiranja internet aplikacija i time omogući njihovo bolje razumevanje i osposobljavanje za programiranje složenih internet strana. U praktikumu će prvo biti prikazan primer realizacije statičke veb stranice. Ipak, glavni akcenat je na dinamičkim internet stranama koje omogućavaju interakciju veb korisnika, sa posebnom pažnjom na povezivanje veb strane sa bazom podataka na serverskoj strani.

Praktikum se sastoji iz četiri poglavlja. U uvodnom delu su opisane osnove interneta kao i protokola koje on koristi kako bi se lakše razumelo i samo programiranje internet aplikacija. Drugo poglavlje posvećeno je dizajniranju statičkih veb strana čija je uloga isključivo prikazivanje informacija. U trećem poglavlju prikazani su primeri programiranja dinamičkih aplikacija koje se izvršavaju na klijentskoj strani. Četvrto poglavlje predstavlja dinamičke aplikacije na serverskoj strani, i rad sa bazama podataka.

Dugujem veliku zahvalnost kolegama sa Departmana za tehničko-tehnološke nauke Državnog univerziteta u Novom Pazaru koje su deleći svoje iskustvo pomogle da ovaj praktikum izgleda bolje. Posebno zahvaljujem dr Aldini Avdić koja je kao asistent na predmetu za koji je praktikum pisan učestvovala praktično u realizaciji svih sadržaja predmeta. Primeri koji su navedeni inspirisani su seminarskim radovima studenata tako da jedan deo zasluge ide mojim studentima. Za sve primedbe, čiji je cilj otklanjanje propusta u praktikumu autor se unapred zahvaljuje. Primedbe i sva dodatna pitanja mogu se slati na mejl autora umarovac@np.ac.rs.

Autor,
U Novom Pazaru, februar 2022

SADRŽAJ

I.	UVOD	1
II.	STATIČKE INTERNET PREZENTACIJE	11
	<i>Primer 1. Pregledni HTML dokument sa upotrebom CSS-a.....</i>	<i>11</i>
	<i>Primer 2. Tabele u HTML-u</i>	<i>16</i>
	<i>Primer 3. Stilizovana tabela</i>	<i>19</i>
III.	DINAMIČKE INTERNET APLIKACIJE NA KLIJENTSKOJ STRANI	21
	<i>Primer 4. Forme i obrada korisničkih zahteva na klijentskoj strani</i>	<i>21</i>
	<i>Primer 5. DOM objekti</i>	<i>29</i>
IV.	DINAMIČKE INTERNET APLIKACIJE NA SERVERSKOJ STRANA.....	33
	<i>Primer 6. Opseg delovanja promenljivih u PHP-u</i>	<i>33</i>
	<i>Primer 7. Superglobalne promenljive.....</i>	<i>36</i>
	<i>Primer 8. Prosleđivanje argumenta pomoću GET metode.....</i>	<i>40</i>
	<i>Primer 9. Prosleđivanje argumenta pomoću POST metode.....</i>	<i>44</i>
	<i>Primer 10. Nizovi u PHP-u</i>	<i>54</i>
	<i>Primer 11. Asocijativni nizovi, sortiranje i druge funkcije za rad sa nizovima...</i>	<i>58</i>
	<i>Primer 12. Znakovni nizovi u PHP-u</i>	<i>63</i>
	<i>Primer 13. Regularni izrazi</i>	<i>66</i>
	<i>Primer 14. Super globalna promenljiva \$_FILES</i>	<i>69</i>
	<i>Primer 15. Rad sa datotekama</i>	<i>77</i>
	<i>Primer 16. Rad sa kolačićima.....</i>	<i>79</i>
	<i>Primer 17. Rad sa sesijama</i>	<i>87</i>
	<i>Primer 18. Kreiranje baze podataka, korisnika i korisničkih privilegija.....</i>	<i>93</i>
	<i>Primer 19. Povezivanje i rad sa bazom podataka</i>	<i>95</i>
	<i>Primer 20. Slanje elektronske pošte.....</i>	<i>117</i>
	LITERATURA	129

I. Uvod

Još od memeksa [1] koga je 1945. godine opisao Vanevar Buš (*Vannevar Bush*) kao uređaj u kome pojedinac čuva svoje knjige, zapise i komunikaciju a koji je realizovan tako da im se brzo i fleksibilno može pristupiti, razvijala se ideja o potrebi proširenja ljudskog sećanja. Ako se posmatra svet danas, može se reći da je postao simbioza između čoveka i računara i da se pola ljudskog pamćenja, ali i logičkog zaključivanja prebacilo u digitalni svet. Internet je za čoveka danas izvor informacija, mesto socijalnog umrežavanja, sredstvo za komunikaciju, kao i poslovni prostor.

Internet sa strukturnog stanovišta je svetska računarska mreža koja se sastoji od oko milijardu računara domaćina (*eng. host computer*) raspoređenih po celom svetu koji se nalaze u DNS¹ (*Domain Name System*) sistemu. Da bi se regulisalo povezivanje heterogene mrežne opreme i softvera uvedena je slojevita arhitektura mreže koja je opisana referentnim modelima ISO OSI (*International Organization for Standardization Open Systems Interconnection*) i TCP/IP (*Transmission Control Protocol/Internet Protocol*). Ovim standardima rešeni su problemi vezani za komunikacione interfejse, formate za razmenu podataka i komunikacione protokole. Protokol definiše format i redosled poruka koje se razmenjuju između dva ili više komunicirajućih entiteta, kao i akcije koje se preduzimaju nakon slanja i prijema poruke ili nekog drugog događaja [2]. Za internet se ne može reći da je jedna mreža već skup različitih mreža koji koriste neke zajedničke protokole i usluge. TCP i IP protokoli predstavljaju najznačajnije protokole interneta. Za standardizaciju interneta zadužena je Tehnička radna grupa za internet (IETF - *Internet Engineering Task Force*) koja standard dokumentuje RFC (*Request for Comments*) dokumentima. Ovim dokumentima su opisani protokoli interneta, ona sadrže niz tehničkih uputstava i jako su detaljna. Slojevi i glavni protokoli koji se koriste na internetu prikazani su tabeli 1.

Značaj interneta se ogleda u uslugama koje pruža kroz razne mrežne aplikacije. Protokoli aplikativnog sloja omogućavaju da se ove različite usluge prema korisnicima izvršavaju. Najpopularnije usluge interneta su: rad na udaljenom računaru, transfer datoteka, elektronska pošta, veb, razmena poruka, video konferencije i druge.

¹ Sve skraćenice su pisane u origanlu na engleskom jeziku

Tabela 1. Slojevi i protokoli na internetu

Sloj	Protokoli
Aplikativni	<i>FTP, HTTP, Telnet, SSL, DNS</i>
Transportni	<i>TCP, UDP</i>
Mrežni	<i>IP, ICMP, IGMP, ARP, RARP</i>
Fizički	<i>IRDA, DSL, 10BASE-T, RS-232</i>

Dve aplikacije koje se nalaze na različitim stranama komuniciraju kroz procese. Međusobna komunikacija mrežnih procesa odvija se kroz internet utičnice (*engl. network socket*) slanjem poruka. Protokoli aplikativnog sloja određuju tipove poruka koje se razmenjuju, strukturu i značenje njihovih polja, kao i kako i kada ih proces šalje i odgovara na njih. U tabeli 2. je prikazano nekoliko najviše korišćenih mrežnih aplikacija i aplikativnih protokola koje one koriste.

Tabela 2. Internet usluge i protokoli aplikativnog sloja

Mrežne aplikacije	Protokoli
Veb	<i>HTTP</i>
Elektronska pošta	<i>SMTP, POP3 i IMAP</i>
Udaljeni pristup računaru	<i>Telnet</i>
Transfer datoteka	<i>FTP</i>

Dve najpopularnije aplikacije interneta koje su izmenile rad i život ljudi jesu veb i elektronska pošta pa će one dalje biti detaljnije opisane.

VEB

Jedana od najpoznatijih i najšire korišćenih usluga interneta je veb. Ranih 1990-tih godina Tim Berners Li (*Tim Berners Lee*) izumeo je svetsku komunikacionu mrežu WWW (*World Wide Web*) poznatu i kao veb (*engl. web*) koja je promenila pogled na internet. Veb je sistem međusobno povezanih dokumenata poznatih kao veb strane koje mogu da sadrže tekst, slike, video snimke i dr. Jedna od osnovnih osobina veba je da funkcioniše na zahtev i da korisnik dobija samo one informacije koje želi. Svaki pojedinac može uz male zahteve da ponudi neke informacije putem veba. Za veb su vezani sledeći pojmovi:

- Hiperveza, hiperlink ili samo link - osobina pojedinih reči, slika ili delova u tekstu koji su dati u elektronskom obliku da predstavljaju posrednu vezu ka dodatnim informacijama koje se nalaze u drugom delu teksta ili drugom dokumentu. Tekst koji je povezan naziva se ankor tekst.
- Hipertekst, hipermedija - obuhvata tekst, slike, zvuk, animacije koje sadrže hiperveze.

Veb je ogroman informacioni sistem koji se sastoji od softverskih aplikacija ili procesa koji razmenjuju informacije i izvršavaju se u korist korisnika ili drugih aplikacija.

Komponente Veb-a su:

Identifikatori - Jedinstvena specifikacija koja identifikuje objekte u sistemu: URI (*Uniform Resource Identifier*).

Formati - Skup specifikacija formata podataka projektovanih za razmenu podataka između komponenti sistema, npr. HTML (*HyperText Markup Language*), XML (*Extensible Markup Language*) šema, itd.

Protokoli - Mali i neisključivi skup specifikacija protokola za razmenu informacija između komponenti sistema, na primer HTTP (*Hypertext Transfer Protocol*) [3].

Osnovni protokol veba je HTTP i predstavlja protokol za prenos hiperteksta.

Stranica veba može biti:

Statička - ako je predstavljena dokumentom koji ne menja svoj sadržaj tokom prikazivanja

Dinamička - ako se sadržaj dokumenta generiše na zahtev pomoću programa ili je i sam taj dokument program.

Veb stranice se prikazuju u veb čitaču (*engl. web browser*) koji omogućava korisniku da reaguje na njegove akcije (na primer pritiskanje dela teksta koji je hiperveza). Da bi se pristupilo određenoj stranici na internetu potrebno je da se stranica identifikuje i za to služi URL koji je deo URI-a koji jedinstveno identifikuje svaki resurs i ime na vebu. URL obuhvata mrežni protokol za preuzimanje odgovarajućeg mrežnog resursa (http, https, ftp, itd.). URL ima tri dela: protokol, DNS ime računara na kome se nalazi veb stranica i putanja do stranice. Na primer:

<http://www.dunp.np.ac.rs/tehnickenauke/softversko-inzenjerstvo/>

1. protokol: http

2. DNS: www.dunp.np.ac.rs

3. putanja: tehnickenauke/softversko-inzenjerstvo/.

Kada korisnik na serverskoj strani pritiskom na neki hiperlink ili pomoću veb čitača zatraži resurs sa određenim URL-om, čitač uspostavlja vezu sa kompjuterom čiji je DNS dat u URL-u i šalje mu zahtev za odgovarajućom datotekom. Savremeni veb server ne vrši samo usluge vraćanja tražene datoteke već razrešava ime zahtevane veb stranice, utvrđuje ovlašćenja klijenta, pretražuje keš, preuzima zahtevane stranice i pokreće programe, kompletira odgovor, kao na primer upis MIME(*Multipurpose Internet Mail Extensions*) tipa, vraća odgovor i upisuje stavke u dnevnik servera [4].

Protokolom HTTP se definiše način na koji veb klijenti traže veb strane od veb servera, kao i način na koji veb serveri šalju tražene veb strane klijentima. Ovo je jednostavan protokol koji ne sadrži stanja, to jest svaki par zahtev klijenta i odgovor servera se odvija nezavisno od drugih zahteva i odgovora. Razlikuju se dva tipa poruka: zahtev klijenta ka serveru i odgovor servera klijentu. Obe poruke se sastoje od: početne linije, jedne ili više linija zaglavlja, praznog reda i tela poruke. Poruke su napisane u standardnom ASCII (*American Standard Code for Information Interchange*) tekstu.

Poruke zahteva klijenta prema serveru u početnoj liniji sadrže tri dela:

- HTTP metod
- URL
- Verziju HTTP-a.

Prikazan je HTTP zahtev koji se dobija pozivom stranice <http://www.dunp.np.ac.rs>.

HTTP zahtev

GET /tehnickenauke/ HTTP/1.1

Host: www.dunp.np.ac.rs

Connection: keep-alive

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)

AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.54

Safari/537.36

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9

Accept-Encoding: gzip, deflate

Accept-Language: sr-RS,sr;q=0.9,en-US;q=0.8,en;q=0.7

Cookie: _ga=GA1.3.476011270.1571081355;

_gid=GA1.3.580202229.1635283574; pll_language=lt

HTTP metode mogu biti: GET, HEAD, POST, PUT, DELETE, CONNECT, OPTIONS i TRACE. Veliki broj HTTP zahteva koristi metodu GET kojom se od servera traži resurs čiji je URL prosleđen. Zahtev metoda GET nema telo, a eventualni parametri su zapisani u okviru putanje.

Primer: GET / tehnickenauke?upit=softversko/ HTTP/1.1

Upotreba GET metode za prosleđivanje podataka serveru data je u primeru 8. Prikaz URL-a i HTTP-a zahteva za primer 8. su prikazani u produžetku.

URL zahteva

<http://localhost/login.php?korisnickoIme=Paja&lozinka=Pata>

HTTP zahtev

```
GET /login.php?korisnickoIme=Paja&lozinka=Pata HTTP/1.1
Host: localhost
Connection: keep-alive
....
```

Napredniji metod koji ima dvojaku funkciju - podnošenje (novog) sadržaja (koji se prenosi kroz telo zahteva) i čitanje sadržaja (slično kao u slučaju metoda GET) je POST metod.

Primer:
POST / tehnickenauke / HTTP/1.1
....
upit=softversko

Primer prosleđivanja podataka iz forme serveru pomoću POST metode dat je u primeru 9, a HTTP zahtev za ovaj primer je priložen u produžetku.

HTTP zahtev

POST /primeriphp/PHP/primer%209/anketar.php HTTP/1.1

Host: localhost

...

pitanje1: Ne

pitanje2: Srednje

pitanje3: 3

pitanje4[]: JS

pitanje4[]: PHP

pitanje4[]: C

komentar:

popuni_anketu: Submit

Osnovna razlika između GET i POST metode je to što su svi parametri koji su prosleđeni GET metodom vidljivi u URL-u, dok to nije slučaj sa POST metodom. Međutim, tu se krije jedna suštinska razlika, a to je da su GET pozivi idempotentni, to jest keširaju se. Zbog keširanja više GET poziva sa istim parametrima će uvek, se vraćati isti odgovor. Stoga, GET pozivi se trebaju koristiti samo u situacijama kada se vrednost rezultata ne menja ako se ne menjaju parametri. POST pozivi nisu idempotentni, jer njima nije dozvoljeno keširanje, pa su stalno u kontaktu sa serverom.

Početna linija poruke koja sadrži odgovor servera sadrži :

- Verziju HTTP protokola
- Statusni kod
- Opis

HTTP odgovor

HTTP/1.1 200 OK

Server: nginx

Date: Tue, 26 Oct 2021 21:28:43 GMT

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

Transfer-Encoding: chunked

X-Powered-By: PHP/7.0.33

Vary: Accept-Encoding, Cookie

Link: <http://www.dunp.np.ac.rs/wp-json/>; rel="https://api.w.org/",
<http://www.dunp.np.ac.rs/?p=6226>; rel=shortlink

X-TEC-API-VERSION: v1

X-TEC-API-ROOT: http://www.dunp.np.ac.rs/wp-json/tribe/events/v1/
X-TEC-API-ORIGIN: http://www.dunp.np.ac.rs
X-Proxy-Cache: MISS
Set-Cookie: uid=w/xusWF4cwu6/PAaBE8qAg==; expires=Thu, 25-Nov-21 21:28:43 GMT; domain=\$host; path=/
P3P: policyref="/w3c/p3p.xml", CP="CUR ADM OUR NOR STA NID"
Content-Encoding: gzip

Format HTTP poruke

Request URL: http://www.dunp.np.ac.rs/tehnickenauke/
Request Method: GET
Status Code: 200 OK
Remote Address: 195.252.110.177:80
Referrer Policy: strict-origin-when-cross-origin

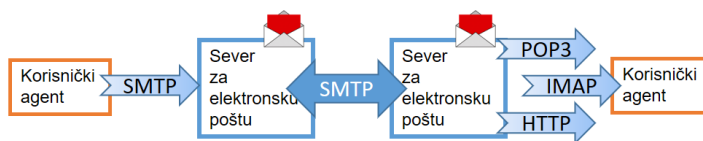
Statusnim kodom se opisuje odgovor servera, u tabeli 3 su dati neki statusni kodovi definisani u RFC 2616.

Tabela 3. Neki statusni kodovi HTTP odgovora

Kod	Opis originalni	Prevod opisa
100	Continue	Nastavi
200	OK	Sve je u redu
300	Multiple Choices	Višestruki izbor
301	Moved Permanently	Trajno pomeren
400	Bad Request	Pogrešan zahtev
404	Not Found	Nije pronađen resurs
500	Internal Server Error	Greška na serveru
505	HTTP Version Not Supported	Nije podržana verzija HTTP-a

ELEKTRONSKA POŠTA

Može se reći da je elektronska pošta jedan od najranije korišćenih internet usluga. Prve programe za razmenu poruka između računara (sendmail i readmail) napisao je Rej Tomlinson (*Raymond Samuel Tomlinson*). On je uveo format adrese username@hostname i upotrebio @ da razdvoji ime korisnika od domena servera elektronske pošte. U slanju jedne elektronske poruke učestvuju četiri računara: dva server računara i dva klijenta. Sandučići se nalaze na internet serverima, slanje i čitanje poruke se obavlja preko korisničkih agenata na strani klijenta. Protokoli koji se koriste u okviru elektronske pošte su SMTP (*Simple Mail Transfer Protocol*), POP3 (*Post Office Protocol version 3*) i IMAP (*Internet Message Access Protocol*). Na slici 1. su prikazani protokoli za slanje i čitanje elektronske pošte. SMTP protokol je standardni protokol za slanje elektronske pošte. Za razliku od HTTP protokola kod koga komunikaciju započinje onaj koji nešto traži, kod SMTP protokola komunikaciju započinje pošiljalac. Ovaj protokol se koristi za slanje poruke od klijenta pošiljaoca do servera elektronske pošte pošiljaoca kao i za slanje poruke između servera za elektronsku poštu da bi bila smeštena u primaćevo sanduće.



Slika 1. Protokoli za slanje i čitanje elektronske pošte

SMTP protokol ima klijentsku stranu koja se izvršava na pošiljačevom serveru za elektronsku poštu i serversku koja se izvršava na primaćevom serveru za elektronsku poštu. Elektronska poruka se sastoji od zaglavlja i tela. Klijent SMTP u zaglavlje dodaje informaciju o pošiljaocu (from), primaocu (to) i predmetu pošte (subject). Na osnovu standarda RFC 822, elektronske poruke sadrže samo ASCII znakove pa se za potrebe multimedijalnih poruka dodaju dodatna polja definisana MIME protokolom (RFC 2045) unutar elektronske poruke. Da bi mogle da se preduzmu odgovarajuće akcije, dekodiranja i upravljanja porukom dodatna polja su: verzija protokola (MIME-Version), način kodiranja (Content-Transfer-Encoding) i tip sadržaja (Content-Type). Na vrh poruke prijemni SMTP server umeće poruku serveru koji je primio (Delivered-To), o serveru koji je poslao i o vremenu primanja poruke (Received).

Prva poruka: Tekstualna poruka Content-Type: text/plain; charset="UTF-8"

pošiljac: apljaskovic@np.ac.rs primalac: umarovac@np.ac.rs

Delivered-To: umarovac@np.ac.rs

Received: by 2002:a17:90a:2c0f:0:0:0:0 with SMTP id m15csp1955082pjd;
Fri, 22 Oct 2021 09:43:26 -0700 (PDT)

.....

MIME-Version: 1.0

From: "Aldina Avdić" <apljaskovic@np.ac.rs>

Date: Fri, 22 Oct 2021 18:43:13 +0200

Message-ID:

<CAFpL5APKbDAz1K6QU3Hhe0+FWAWfmVAa36JLWjT2N-QqeDoo5A@mail.gmail.com>

Subject: prva poruka

To: Ulfeta Marovac <umarovac@np.ac.rs>

Content-Type: multipart/alternative;
boundary="000000000000bcb9d005cef3b5c8"

--000000000000bcb9d005cef3b5c8

Content-Type: text/plain; charset="UTF-8"

Puno pozdrava!!!

--000000000000bcb9d005cef3b5c8

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

<div dir="auto">Puno pozdrava!!!</div>

--000000000000bcb9d005cef3b5c8—

Druga poruka: Tekstualna poruka sa slikom Content-Type: text/plain;
charset="UTF-8", Content-Type: image/jpeg; pošiljac: apljaskovic@np.ac.rs
primalac: umarovac@np.ac.rs

Delivered-To: umarovac@np.ac.rs

Received: by 2002:a17:90a:2c0f:0:0:0:0 with SMTP id m15csp1956105pjd;
Fri, 22 Oct 2021 09:44:37 -0700 (PDT)

....

MIME-Version: 1.0

From: "Aldina Avdić" <apljaskovic@np.ac.rs>

Date: Fri, 22 Oct 2021 18:44:24 +0200

Message-ID: <CAFpL5ANMQwMkv3gWaJsEB=-
+v6_8ztKLP9Hhs4vrX0NUx-O6pw@mail.gmail.com>

Subject: Druga poruka

To: Ulfeta Marovac <umarovac@np.ac.rs>

Content-Type: multipart/mixed;
boundary="000000000000fb020f05cef3b9a9"

--000000000000fb020f05cef3b9a9

Content-Type: multipart/alternative;
boundary="000000000000fb020e05cef3b9a7"

--000000000000fb020e05cef3b9a7

Content-Type: text/plain; charset="UTF-8"

--000000000000fb020e05cef3b9a7

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

<div dir="auto"></div>

--000000000000fb020e05cef3b9a7--

--000000000000fb020f05cef3b9a9

Content-Type: image/jpeg; name="IMG-
102ad6d55d1c26a506c1980f3e61de59-V.jpg"

Content-Disposition: attachment;

filename="IMG-102ad6d55d1c26a506c1980f3e61de59-V.jpg"

Content-Transfer-Encoding: base64

Content-ID: <17ca8e396748a485f6b1>

X-Attachment-Id: 17ca8e396748a485f6b1

--000000000000fb020f05cef3b9a9--

Protokol SMTP šalje poruke, a za pristup elektronskim porukama zaduženi su POP3, IMAP i HTTP. Sve više se za čitanje i slanje elektronske pošte koristi Web, pa samim tim i HTTP protokol.

II. Statičke internet prezentacije

Statička veb strana služi isključivo za prikazivanje informacija. Statičke veb strane su međusobno povezane hiperlinkovima. Sadržaj veb stranice opisan je pomoću jezika HTML dok se njenim izgledom može upravljati upotrebom CSS-a (*Cascading Style Sheets*) [4]. HTML je jezik za označavanje koji je namenjen veb prezentacijama. Njega interpretira veb pregledač kako bi znao da organizuje informacije. HTML dokument je sačinjen od elemenata: ceo dokument, pasus, tabela, slika,... Elementi se označavaju oznakama (etiketama ili tagovima), čime se obeležava početak i kraj elementa. CSS je prekretnica u veb dizajnu zato što dozvoljava programerima da kontrolišu stil i izgled više veb stranica odjednom. CSS kaskadno stilizuje veb stranice što znači da kada se jedan stil primeni na jedan element, mnoga svojstva se kaskadno prenose i na svu decu elemente unutar tog elementa. Upotrebom CSS-a obezbeđuje se povezivanje kontrolnih elemenata (bilo da su XML ili HTML) sa određenom vrstom prikaza na raspoloživim medijima.

U ovom poglavlju dati su primeri statičkih veb stranica opisanih pomoću HTML i CSS dokumenata. Za svaki prikazani primer dat je kod kao i izgled stranice nakon pozivanja odgovarajućeg koda.

Primer 1. Pregledni HTML dokument sa upotrebom CSS-a

Ovim primerom je prikazana struktura html dokumenta kao i osnovna formatiranja pomoću stilskih listova. Prikaz rezultata: slika 2.

primer1.html

```
<!-- Ovo je komentar u HTML-u -->
<!-- DOCTYPE-om se označava se verzija koja se koristi
-- DOCTYPE html označava HTML dokument verzije 5 -->
<!DOCTYPE html>

<!-- Cela strana je unutar jednog elementa html -->
<html lang="en">
```

<!-- html element se sastoji iz: zaglavlja (head) i tela (body) -->

`<head>`

<!-- Meta elementi:

-- kodne strane korišćene za zapis teksta,

-- autor stranice,

-- opis,

-- ključne reči i slično. -->

`<meta charset="UTF-8">`

`<meta name="autor" content="Ulfeta Marovac">`

`<meta name="description" content="Pregledni HTML dokument sa upotrebom CSS-a">`

`<meta name="keywords" content="HTML, CSS">`

<!-- Naslov stranice koji je vidljiv na kartici pregledača

-- u kojoj je stranica otvorena -->

`<title> Primer 1: HTML, CSS </title>`

<!-- Elementom link se uključuju dodatne datoteke,

-- atributi:

rel - opis datoteke

href - putanja do datoteke.

type - tip datoteke u formatu MIME type -->

<!-- Uključivanje CSS stilova iz odvojene datoteke -->

`<link rel="stylesheet" href="style.css" type="text/css">`

<!-- Uključivanje ikonice (favicon) koja se prikazuje u kartici pregledača

-->

`<link rel="icon" href="sun.png" type="image/png">`

<!-- Element style sa datim CSS svojstvima za element h1 -->

`<style>`

`h1 {`

`color: blue;`

`}`

`</style>`

`</head>`

<!-- Telo prvog primera -->

`<body>`

```
<!-- Sadržaj veb stranice se može podeliti na sekcije -->
<section>
```

```
<!-- Sekcije se dele na članke – aricle -->
<article>
```

```
<!-- CSS svojstva se mogu navoditi i u okviru style atributa
-- i imaju najviši prioritet -->
```

```
<h1 style="color: red">Ovo je naslov tipa h1 kome je promenjena boja</h1>
<h1>Još jedan h1 naslov koji je formatiran stilom zadatim u zaglavlju</h1>
<h2>Podnaslov h2 čije je formatiranje dato u fajlu style.css</h2>
<h3>Podpodpodnaslov h3 sa podrazumevanim formatiranjem </h3>
```

```
<!-- Elementima se može dodeliti JEDINSTVENI identifikator -->
<h3 id="moj_podnaslov">Podnaslov h3 sa specijalnim formatiranjem</h3>
```

```
<!-- Elementima se može dodeliti klasa kojoj pripadaju,
-- više elemenata može pripadati istoj klasi -->
<h4 class="zeleni_podnaslovi">Zeleni podnaslov h4 </h4>
<h5 class="zeleni_podnaslovi">Zeleni podnaslov h5 </h5>
<h6 class="zeleni_podnaslovi">Zeleni podnaslov h6 </h6>
</article>
```

```
<!-- Veb strana se može podeliti na manje logičke celine:
-- main - centralni sadržaj veb strane
-- header - zaglavlje veb strane
-- footer - podnožje veb strane
-- nav - grupisane veze ka veb stranama u okviru istog veb sajta
-- aside - sporedni deo sadržaja veb strane -->
<aside>
<article id=numeracija>
```

```
<!-- Element p prikazuje tekst u jednom redu -->
<p>Novi red se ispisuje pomoću elementa <br> br </p>
<pre>Element pre podržava
tekst sa belinama </pre>
```

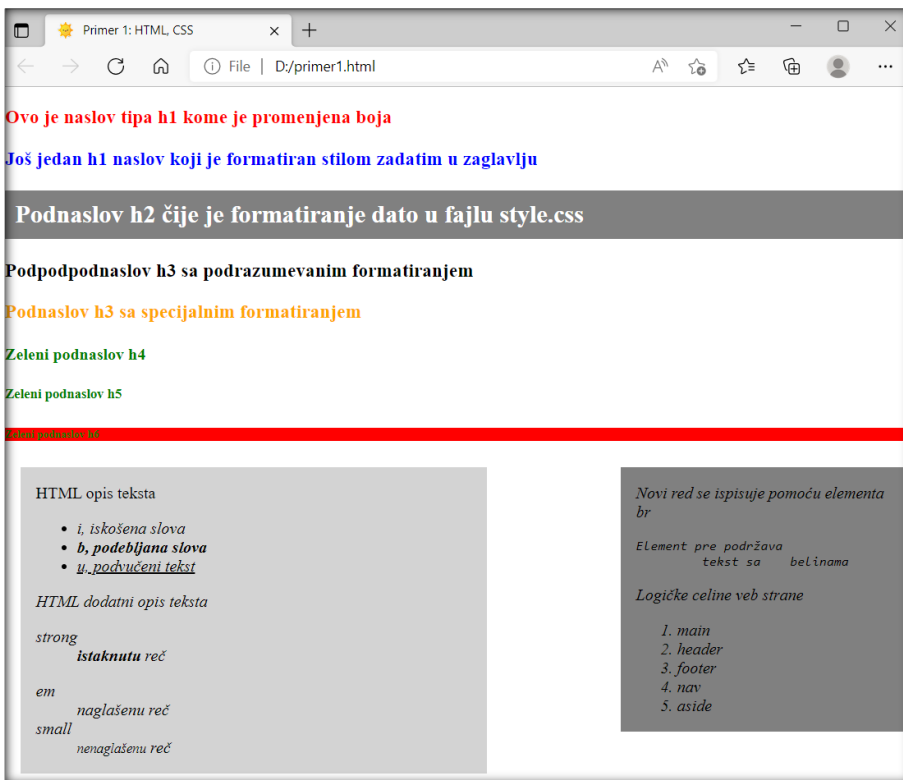
```
<!-- Numerisane liste označavaju se elementom ol.
-- Atribut title-prelaskom kursora preko teksta
-- prikazuje se sadržaj atributa title -->
<p title="A ovo je title">Logičke celine veb strane </p>
<ol>
```

```
<li>main</li>
  <li>header</li>
  <li>footer </li>
  <li>nav</li>
  <li>aside</li>
</ol>
</article>
</aside>
<main>
<article>
  <!-- Nenumerisane liste se označavaju elementom ul
  -- dok se stavke liste označavaju elementom li -->
  <p> HTML opis teksta </p>
  <ul>
    <li> <i> i, iskošena slova </i></li>
    <li> <b> b, podebljana slova </b> </li>
    <li> <u> u, podvučeni tekst </u></li>
  </ul>

  <!-- Opisne liste se označavaju elementom dl
  -- stavke liste se označavaju elementima dt (naziv pojma)
  -- i dd (opis pojma) -->
  <p> HTML dodatni opis teksta </p>
  <dl>
    <dt>strong</dt>
    <dd><strong>istaknutu</strong> reč</p></dd>
    <dt>em </dt>
    <dd><em> naglašenu</em> reč</dd>
    <dt>small</dt>
    <dd><small> nenaglašenu</small> reč</dd>
  </dl>
</article>
</main>
</section>
</body>
</html>
```

style.css

```
/* Ovo je komentar u CSS-u */
/* CSS se sastoji od pravila
  * svako pravilo je oblika selektor opis
  * selektor - ime_elementa, #identifikator, .klasa
  * opis - niz deklaracija oblika svojstvo: vrednost
*/
h2{
    background-color: grey;
    color: white;
    padding: 10px;
}
#moj_podnaslov {
    color: orange;
}
.zeleni_podnaslovi {
    color: green;
}
/* Moguće je selektovati samo elemente
  * određenog tipa koji pripadaju nekoj klasi
*/
h6.zeleni_podnaslovi {
    background-color: red;
}
aside {
    width: 30%;
    padding-left: 15px;
    margin-left: 15px;
    float: right;
    font-style: italic;
    background-color: gray;
}
main {
    width: 50%;
    padding-left: 15px;
    margin-left: 15px;
    float: left;
    background-color: lightgray;
}
```



Slika 2: HTML dokument sa upotrebom CSS-a (primer1. html, style.css)

Primer 2. Tabele u HTML-u

Tabelama se u HTML-u prikazuju podaci čija je priroda pogodna za ovakav prikaz, ali se i bolje organizuje prostor. Ovim primerom se pokazuje način za tabelarno prikazivanje sadržaja u HTML-u. Prikaz rezultata: slika 3.

primer2.html

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Tabele u HTML-u">
  <meta name="keywords" content="HTML, CSS, Tabele">
  <title>Primer 2: Tabele</title>
```

```

<link rel="icon" href="star.png" type="image/png">
</head>
<body>
  <h1> Tabele</h1>

  <!-- Tabele se označavaju elementom table kao niz vrsta
    -- vrsta se predstavlja elementom tr i može sadržati polja td.
    -- naslovne ćelije predstavljaju se elementom th
    -- zaglavlje table odvajaju se elementima thead i tbody
    -- tabele se podrazumevano prikazuju bez okvira
    -- okvir se može dodati tabeli postavljanjem atributa border
    -- za spajanje nekoliko susednih kolona, koristi se atribut colspan
    -- za spajanje nekoliko susednih vrsta, koristi se atribut rowspan
    -- naslov se dodaje pomoću elementa caption
  -->
  <table border="1">
    <caption>Specijalni simboli</caption>
    <thead>
      <tr>
        <th rowspan="2" colspan="2">Spojene dve kolone i dve vrste</th>
      </tr>
      <tr> <!-- Prazan red--></tr>
    </thead>
    <tbody>
      <tr>
        <th>

```

<!-- Tabela kao sadržaj ćelije-->

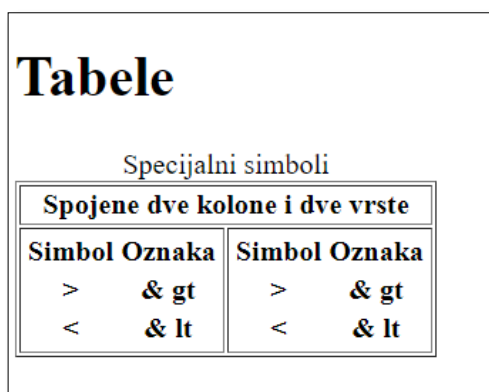
```

<table>
  <tr>
    <th>Simbol</th> <th>Oznaka </th>
  </tr>
  <tr>
    <td> &gt; </td> <td> & gt</td>
  </tr>
  <tr>
    <td> &lt; </td> <td> & lt</td>
  </tr>
</table>
</th>
<th>

```

<!-- Tabela kao sadržaj ćelije-->

```
<table>
  <tr>
    <th>Simbol</th> <th>Oznaka </th>
  </tr>
  <tr>
    <td> &gt; </td> <td> & gt</td>
  </tr>
  <tr>
    <td> &lt; </td> <td> & lt</td>
  </tr>
</table>
</th>
</tr>
</body>
</table>
</body>
</html>
```



The screenshot shows a web page with the title "Tabele". Below the title is a link "Specijalni simboli". The main content is a table with two columns and two rows. The first row is a header for the columns, and the second row contains the symbols and their HTML entities.

Specijalni simboli			
Spojene dve kolone i dve vrste			
Simbol Oznaka		Simbol Oznaka	
>	& gt	>	& gt
<	& lt	<	& lt

Slika 3. Tabele u HTML-u (*primer2.html*)

Primer 3. Stilizovana tabela

Pomoću CSS-a tabela i njene ćelije mogu se prilagoditi veličini stranice i dodatno formatirati. U ovom primeru je prikazana stilizovana tabela kao i njena upotreba da bi se bolje organizovao prostor veb stranice. Prikaz rezultata: slika 4.

primer3.html

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Stilizovana tabela">
  <meta name="keywords" content="HTML, CSS, Tabele">
  <title>Primer 3: Tabele</title>
  <link rel="icon" href="star.png" type="image/png">
  <style>

/* Poništava se formatiranje linkova
* kad se pređe kursorom miša pozadina postaje boje FF6600
* tabele se prostiru po maksimalnoj širini i dužini stranice
* navigacija - zauzima 25% strane, i ima žutu pozadinu
* sadržaj - leva i desna margina su 6px
* desno - formatira element na sirinu 150 px i maksimalne visine*/
a:link{text-decoration:none;}
a:hover{background-color:#FF6600;}

table{
height:100%;
width:100%;
}
#navigacija{
width:25%;
background-color:yellow;
}
#sadrzaj{
margin-left: 6px;
margin-right: 6px;
}
#desno{
```

```

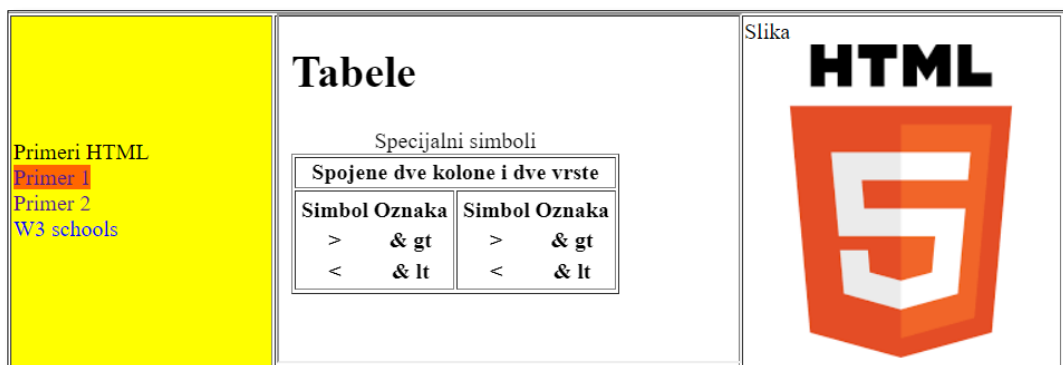
width: 150px;height: 100%;
}
</style>
</head>
<body>
  <table border ="1">
    <tr>

      <!-- Prikazivanje hiperlinkova do apsolutno adresiranih fajlova
      -- i jednog relativno adresiranog -->
      <td id="navigacija">Primeri HTML</a><br/>
        <a href="primer1.html">Primer 1</a><br/>
        <a href="primer2.html">Primer 2</a><br/>
        <a href="https://www.w3schools.com/">W3 schools </a><br/>
      </td>
      <td id="sadrzaj">

        <!-- Prikazivanje drugog HTML dokumenta u okviru-->
        <iframe width="100%" height="100%" src="primer2.html">
          </iframe>
        </td>

        <!-- Prikazivanje slike-->
        <td id="desno">Slika</img></td>
      </tr>
    </table>
  </body>
</html>

```



Slika 4. Stilizovana tabela (*primer3.html*)

III. Dinamičke internet aplikacije na klijentskoj strani

Da bi unapredili veb strane uključivanjem obrade podataka koje korisnik unosi koriste se tehnologije za dinamičku realizaciju veb stranica kao što je JavaScript. Java Script je kompaktan i objektno zasnovan interpretirani jezik namenjen za razvoj internet aplikacija na klijentskoj strani [4]. Narednim primerima biće prikazana obrada formi i rad sa DOM elementima upotrebom JavaScript-a. Prikaz programa kao i rezultati nakon različitih aktivnosti korisnika su prikazani na slikama koje su date posle koda.

Primer 4. Forme i obrada korisničkih zahteva na klijentskoj strani

Da bi se prikupile informacije od posetioca veb prezentacije koriste se forme. Podaci prikupljeni pomoću forme mogu se obrađivati na serverskoj ili na klijentskoj strani. Sledećim primerom će biti prikazana obrada informacija koje korisnik unese na klijentskoj strani. Forma prikazana na slici 5. predstavlja formu za unos podataka o knjizi i čitaocu. Podaci o čitaocu čine jednu celinu (fieldset) i obrađuju se pomoću JavaScript funkcije pritiskom na dugme “Unesi“. JS funkcija proverava da li su svi podaci uneti i ako jesu ispisuje poruku o podacima u paragrafu na kraju forme i menja boju pozadine na boju koju je čitalac izabrao. Drugi deo forme čine podaci o knjizi i sadrži dva dugmeta. Dugme “Pošalji” izvršava formu odgovarajućom metodom (GET) i poziva definisanu akcija koja je u ovom primeru izostavljena. Drugo dugme, “Poništi” postavlja elemente forme na početna podešavanja. Prikaz rezultata: slika 6, slika 7, slika 8. i slika 9.

primer4.html

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Forme, js">
  <meta name="keywords" content="form, js">
  <title>Primer 4:Forme i JScript</title>
  <link rel="icon" href="star.png" type="image/png">
  <!-- Element style sa datim CSS svojstvima za element
       desnog i levog dela prezentacije -->
  <style>
    .sivi_levo{
```

```
background-color: grey;
width: 40%;
float: left;
}
.desno{
width: 50%;
float: right;
}
</style>
```

```
</head>
```

```
<body>
```

```
<!-- Formulari služe za unos korisničkih podataka
-- atributi:
-- action - strana kojoj se prosleđuju podaci za obradu
-- method - metoda kojom se obrađuju prosleđeni podaci
-- može biti GET i POST -->
<form action="" method="GET">
```

```
<!-- Grupa polja koja čine celinu mogu se objediniti sa fieldset
-- ovom fieldset-u je pridružena klasa za formatiranje
-- sivi_levo grupi
-->
<fieldset class="sivi_levo" id="prvi">
```

```
<!-- Elementom legend zadaje se naslov odgovarajućem fieldset-u
-->
<legend style="color: red; font-weight: bold" >Čitalac</legend>
```

```
<!-- Labelama se daju oznake odgovarajućim elementima forme
-- atributom for labela se veže sa odgovarajući element
-- koji opisuje
-- Poljima za unos omogućava se unos informacija korisnika
-- atributi:
type - tip polja
placeholder - tekst kojim se opisuje željeni unos
maxlength - maksimalna dozvoljena dužina teksta
name - naziv polja pomoću koga se može pristupiti u JScript-u
value - vrednost polja -->
<label for="ime">Ime</label>
<input type="text" id="ime" name="ime" placeholder="Unesite ime">
```



```

<br><br>
<label for="prezime">Prezime</label>
<input type="text" id="prezime" name="prezime">
<br><br>
<label for="email">Email:</label>
<input type="email" id="email" name="email">
<br><br>
<label for="lozinka">Lozinka</label>
<input type="password" name="password" id="lozinka"
maxlength="5">
<br><br>
<label for="pozadina">Boja pozadine</label>
<input type="color" name="pozadina" id="pozadina" >
<br><br>
<label for="datum">Datum:</label>
<input type="date" id="datum" name="datum">
<br><br>
<label for="vreme">Vreme registracije</label>
<input type="time" name="vreme" id="vreme">
<br><br>

<!-- Dugme koje nakon pritiskanja izvršava JS funkciju Unesi() -->
<input type="button" value="Unesi podatke" id="unesi"
onclick="Unesi()"/>
<p> Uneli ste sledeće podatke: </p>

<!-- Paragraf “punos“ unutar koga će JS funkcija Unesi()
-- umetnuti tekst -->
<p id="punos"> </p>

<!-- JScript obrađuje vrednosti unete u elementima forme
-- Kreira sadržaj paragrafa punos
-- i postavlja izabranu boju u formi za pozadinu filedset-a -->
<script>
function Unesi()
{

// Komentar u JScript-u
/* Promenljive definisane sa var
* document.getElementById-pristup elementu html dokumenta
* pomoću id atributa
* value vrednost elementa*/

```

```

var ime=document.getElementById("ime").value;
var prezime=document.getElementById("prezime").value;
var email=document.getElementById("email").value;
var lozinka=document.getElementById("lozinka").value;
var datum=document.getElementById("datum").value;
var vreme=document.getElementById("vreme").value;
var pozadina=document.getElementById("pozadina").value;

/* Ukoliko postoji nepopunjeno polje prikazuje se okvir
   * sa upozorenjem(alert), inače prikazuje poruku*/
if(ime=="" || prezime=="" || email=="" || lozinka=="" || datum==""
   || vreme=="" || pozadina=="")
{
    alert("Popunite sva polja!");
}
else{
    punos.innerHTML="Hvala puno "+ime+" "+prezime+"!!!";
    document.getElementById("prvi").style.backgroundColor =pozadina;
}
}
</script>
</fieldset>

```

```

<!-- Sledeći fieldset sadrži različite tipove polja za unos
-- za izbor: radiobutton, checkbox
-- atributom checked postavlja se početno stanje odgovarajućeg
-- polja na
-- "izabrano";
-- dodatno:
-- file - za izbor fajla,
-- range - slajder za odabir numeričke vrednosti za dati opseg
-- number - izbor dozvoljene numeričke vrednosti
-- select - polja sa padajućom listom za izbor
-- textarea - polje za unos teksta
-- submit - dugme za izvršavanje forme i prosleđivanje parametara
-- (GET i POST metodom)
-- reset - dugme koje postavlja formu na inicijalnu vrednost -->
<fieldset class="desno" style="background-image: url('knjiga2.jpg');" >
<br><br>
<label for="cbP1">Da li želite da se vaš komentar arhivira ?</label>
<input type="checkbox" checked value="1" name="pitanje1" id="cbP1">
<br><br>

```

```
<label>Izabrite tip knjige</label>
<br><br>
```

```
<!-- Ekskluzivna selekcija
```

```
-- da bi više "radio" polja bila u istoj grupi moraju imati isto ime
```

```
-- (atribut name : name="tip knjige") -->
```

```
<label for="k1">Dečije knjige</label>
<input type="radio" checked name="tip_knjige" value="1" id="k1"><br>
<label for="k2">Edukativne knjige</label>
<input type="radio" name="tip_knjige" value="2" id="k2"><br>
<label for="k3">Romani</label>
<input type="radio" name="tip_knjige" value="3" id="k3"><br>
<br><br>
<label for="opis">Opis</label>
<select name="opis" id="opis">
  <option value="za">zanimljiva</option>
  <option selected value="do">dosadna </option>
  <option value="uz">uzbudljiva</option>
</select>
<br><br>
<label for="obim">Broj strana</label>
<input type="range" min="10" max="2000" step="50" name="obim"
id="obim">
<br><br>
<label for="ocena">Ocena</label>
<input type="number" name="broj" id="broj" min="1" max="10"
step="1">
```

```
<!-- Oznaka za tematski prelom -->
```

```
<hr>
<label for="fajl">Fajl</label>
<input type="file" id="fajl">
<br><br>
<label for="komentar">Komentar</label>
<textarea name="komentar" id="komentar" placeholder="Napišite
komentar..." cols="30" rows="10">
</textarea>
<br><br>
```

```
<!-- input type="submit" - definiše dugme za slanje koje šalje sve
-- vrednosti forme rukovodiocu, onome ko je definisan sa action
-- atributom
```

-- *input type="reset" - definiše dugme za resetovanje koje vraća sve*
-- *vrednosti forme na početne vrednosti. -->*

```
<input type="submit" value="Pošalji">  
<input type="reset" value="Poništi">  
</fieldset>  
</form>  
</body>  
</html>
```

Čitalac

Ime

Prezime

Email:

Lozinka

Boja pozadine

Datum:

Vreme registracije

Uneli ste sledeće podatke:

Da li želite da se vaš komentar arhivira ? ☒

Izabrite tip knjige

Dečije knjige ☒

Edukativne knjige ☐

Romani ☐

Opis

Broj strana

Ocena

Fajl No file chosen

Komentar

Slika 5. HTML forma za unos korisničkih podataka (*primer4.html*)

Čitalac

Ime

Prezime

Email:

Lozinka

Boja pozadine

Datum:

Vreme registracije

Uneli ste sledeće podatke:

This page says
Popunite sva polja!

Dečije knjige ☒
Edukativne knjige ☐
Romani ☐

Opis

Broj strana

Ocena

Fajl No file chosen

Slika 6 . Izgled strane kada nisu podaci u potpunosti popunjeni

Čitalac

Ime

Prezime

Email:

Lozinka

Boja pozadine

Datum:

Vreme registracije

Uneli ste sledeće podatke:
Hvala puno Mini Mouse!!!

Da li želite da se vaš komentar arhivira ? ☒

Izabrite tip knjige

Dečije knjige ☒
Edukativne knjige ☐
Romani ☐

Opis

Broj strana

Ocena

Fajl bajke.docx

Komentar

Slika 7 . Izgled strane nakon pritiskanja dugmeta “Unesi podatke”

Čitalac

Ime

Prezime

Email:

Lozinka

Boja pozadine

Datum:

Vreme registracije

Uneli ste sledeće podatke:
Hvala puno Mini Mouse!!!

Da li želite da se vaš komentar arhivira ? ☒

Izabrite tip knjige

Dečije knjige ☒
Edukativne knjige ☐
Romani ☐

Opis

Broj strana

Ocena

Fajl No file chosen

Komentar

Slika 8 . Izgled strane nakon pritiskanja dugmeta “Poništi”

Čitalac

Ime

Prezime

Email:

Lozinka

Boja pozadine

Datum:

Vreme registracije

Uneli ste sledeće podatke:

Da li želite da se vaš komentar arhivira ? ☒

Izabrite tip knjige

Dečije knjige ☒
Edukativne knjige ☐
Romani ☐

Opis

Broj strana

Ocena

Fajl No file chosen

Komentar

Slika 9 . Izgled strane nakon pritiskanja dugmeta “Pošalji”

Primetimo da se nakon izvršenja forme (pritiskom na dugme „Pošalji“) u pregledaču pojavljuje poziv *primer4.html* programa sa prosleđenim parametrima iz forme:

file:///D:/primer4.html?ime=Mina&prezime=Mouse&email=mmina%40gmail.com&password=cvrs&pozadina=%234e2eb8&datum=2022-02-23&vreme=00%3A15&pitanje1=1&tip_knjige=1&opis=za&obim=1310&broj=5&komentar=Divna+bajka%21%21%21.

Primer 5. DOM objekti

Upravljanje elementima veb prezentacije vrši se pomoću DOM-a (Document Object Model). Svakom elementu web prezentacije, pomoću DOM-a, može se pristupiti JavaScript-om i dalje sa njim postupati kao sa objektom. Sledeći program prikazuje formu (slika 10) koja ispisuje osobine izabranog DOM objekta: *window*, *navigator*, *history*, *screen*, *location*, *document*. U desnom delu forme nalazi se polje za izbor odgovarajućeg elementa i dugme kojim se izvršava JS funkcija za ispis osobina odgovarajućeg DOM elementa u tekstualnom polju koje se nalazi u desnom delu forme.

primer5.html

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="DOM objekti">
  <meta name="keywords" content="DOM, js">
  <title> Primer 5:Prozor i objekti dokumenta </title>
  <link rel="icon" href="star.png" type="image/png">
  <style>
    .desno{
      background-color: grey;
      width: 40%;
      float: right;
    }
    .levo{
      width: 50%;
      float: left;
    }
  </style>
```

```

</style>
</head>
<body>

<!--Forma za izbor elementa čije osobine treba da se prikažu-->
<form>

    <!--Grupa elemenata forme je formatirana kao klasa levo
    -- ovde se nalazi polje za izbor DOM objekta čije osobine
    -- treba da se prikažu kao i dugme za pokretanje JS funkcije -->
    <fieldset class="levo" >
        <br><br>
        <label for="element">Izaberite element čije osobine želite da
prikažete</label>
        <select name="element" id="element">
            <option value="document"> document – aktivni dokument </option>
            <option selected value="window"> window – aktivni prozor
            </option>
            <option value="navigator"> navigator – pregledač </option>
            <option value="history"> history – istorija pregledanja </option>
            <option value="screen"> screen – podaci o ekranu </option>
            <option value="location"> location – rad sa URL-om </option>
        </select>
        <br><br>
        <input type="button" value="Prikazi osobine" id="prikaz"
onclick="Prikaz()"/>

    <!-- JScript obrađuje selektovanu vrednost iz liste
    -- prikazuje elemente objekta -->
    <script>
function Prikaz()
{
    /* Sadržaju elemenata koji ima id „element“ pristupa se pomoću
    * getElementById, a zatim se na osnovu izabranog polja
    * postavlja odgovarajući DOM objekat u promenljivu obj
    */
    switch(document.getElementById("element").value){
        case "document": obj=document;break;
        case "window": obj=window;break;
        case "navigator": obj=navigator;break;
        case "history": obj=history;break;
    }
}
    </script>

```



```
case "screen": obj=screen;break;
case "location": obj=location;break;
}
var result = ""
```

```
/* Sadržaju elemenata nabrojivog tipa može se pristupiti pomoću
* for petlje : for (var imePromenljive in imeSkupa)
* promenljiva result sadrži naziv i ime objekta
*/
```

```
for (var i in obj) {
result += obj + "." + i + " = " + obj[i] + "\n"
}
```

```
/* Sadržaju HTML elementa pristupa se preko innerHTML
* Opis svih elemenata odgovarajućeg DOM objekta biće
* prikazani u tekstualnom polju koje ima id "objekat"
*/
```

```
document.getElementById("objekat").innerHTML=result;
}
</script>
```

```
<!-- Grupa elemenata forme za prikaz osobina izabranog DOM objekta
-- osobine će biti prikazane u tekst polju "objekat" -->
```

```
</fieldset>
```

```
<fieldset class="desno" >
```

```
<legend style="color: red; font-weight: bold" >DOM objekat</legend>
```

```
<label for="objekat">Osobine objekta</label>
```

```
<textarea name="objekat" id="objekat" cols="42" rows=100%>
```

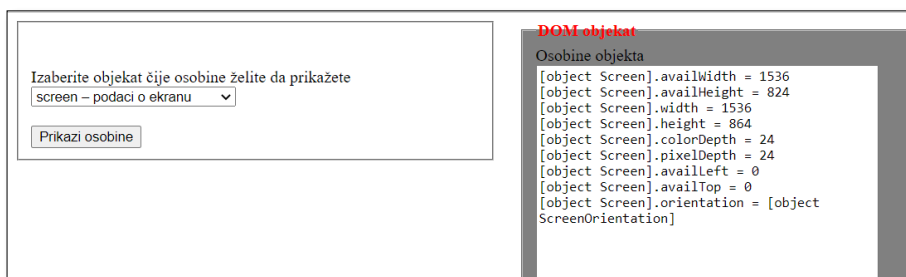
```
</textarea>
```

```
</fieldset>
```

```
</form>
```

```
</body>
```

```
</html>
```



Slika 10. Prikaz osobina izabranog DOM objekta (*primer6.html*)

IV. Dinamičke internet aplikacije na serverskoj strana

PHP (*Hypertext Preprocessor*) je jednostavan, a moćan programski jezik kojim se omogućava izgradnja dinamičkih veb stranica [6]. Obrada koja se definiše pomoću PHP-a se odvija na serverskoj strani, i tipično uključuje obradu i ažuriranje baza podataka. U okviru ovog praktikuma prikazana je upotreba MySQL baze podataka. PHP i MySQL su rešenja otvorenog koda što ih čini popularnim alatima za izradu veb strana.

Primer 6. Opseg delovanja promenljivih u PHP-u

Promenljive u PHP-u mogu biti lokane, globalne i statičke. Kroz sledeći primer prezentovan je opseg delovanja promenljivih i pristup različitim tipovima promenljivih. Vrednost i vidljivost promenljivih iz različitih delova programa može se videti na slici 11.

primer6.php

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Promenljive u PHP-u">
  <meta name="keywords" content="PHP, GLOBALS">
  <title>Primer 6: Promenljive u PHP-u</title>
</head>
<body>
```

```
<!-- PHP kod se može ubaciti u HTML kao skript ali se koristi i sintaksa
-- <?php PHP_kod ?> -->
<?php
```

```
/*
 * Ovo je jedan komentar u više linija unutar PHP programa
 * Promenljive u PHP-u započinju znakom $ i mogu biti:
 * lokalne, globalne i statičke
 * Superglobalne promenljive:
 * $GLOBALS - služi za pristup globalnim varijablama */
```

#Globalna promenljiva x

\$x = 1;

\$a='A';

// Ispis iz PHP programa

echo "<p>Vrednost promenljive globalne promenljive x: \$x</p>";

// Promenljiva x nije vidljiva unutar funkcije

function moja_funkcija() {

//Globalnoj promenljivoj se može pristupiti i pomoću global

global \$a;

/ Statische promenljive pripadaju funkcijama*

** i ne brišu se posle njihovog izvršavanja*

**/*

static \$z=0;

/ Ovaj deo koda će se izvršiti samo u prvom pozivu f-je*

** u svakom sledećem pozivu će z imati veću vrednost*

**/*

if(\$z== 0){

/ Globalne promenljive koje nisu definisane sa global u funkciji*

** neće biti vidljive u funkciji i zato se štampa upozorenje*

**/*

echo "<p>Vrednost promenljive x unutar funkcije: \$x</p>";

echo "<p>Vrednost promenljive a unutar funkcije: \$a</p>";

// Ako se definiše nova promenljiva ona je lokalnog karaktera

\$x="Lokalna promenljiva x";

echo "<p>Vrednost promenljive x definisane unutar funkcije:\$x</p>";

//Globalnoj promenljivoj pristupa se preko \$GLOBALS

echo "<p>Vrednost globalne promenljive x prikazane pomoću\n\$GLOBALS:";

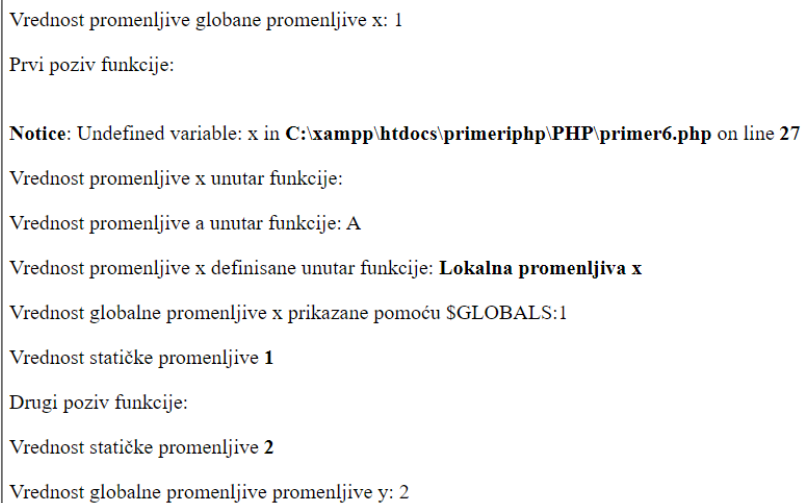
echo \$GLOBALS['x'];

echo "</p>";

// Globalna promenljiva definisana unutar funkcije

```
$GLOBALS['y']=2;
}
$z++;
echo "<p>Vrednost statičke promenljive <b> $z </b> </p>";
}

//Poziv f-je
echo "<p>Prvi poziv funkcije:</p>";
moja_funkcija();
echo "<p>Drugi poziv funkcije:</p>";
moja_funkcija();
echo "<p>Vrednost globalne promenljive promenljive y: $y</p>";
?>
</body>
</html>
```



Vrednost promenljive globalne promenljive x: 1

Prvi poziv funkcije:

Notice: Undefined variable: x in C:\xampp\htdocs\primeriphp\PHP\primer6.php on line 27

Vrednost promenljive x unutar funkcije:

Vrednost promenljive a unutar funkcije: A

Vrednost promenljive x definisane unutar funkcije: **Lokalna promenljiva x**

Vrednost globalne promenljive x prikazane pomoću \$GLOBALS:1

Vrednost statičke promenljive **1**

Drugi poziv funkcije:

Vrednost statičke promenljive **2**

Vrednost globalne promenljive promenljive y: 2

Slika 11. Prikaz promenljivih iz funkcije i glavnog programa (*primer6.php*)

Primer 7. Superglobalne promenljive

Superglobalne promenljive su interfejsi PHP programa sa spoljnom okolinom preko kojih razmenjuju informacije. Superglobane promenljive su: `$GLOBALS`, `$_SERVER`, `$REQUEST`, `$POST`, `$GET`, `$FILES`, `$ENV`, `$_COOKIE` i `$_SESSION`. U prethodnom primeru je prikazana upotreba `$GLOBALS` promenljive. Sledeći program prikazuje upotrebu promenljive `$_SERVER` za prilagođavanje navigacije stranici koja ju je pozvala. Program je realizovan kroz skripte *navigacija.php*, *footer.php* i *index.php*. Skripta *index.php* uključuje skripte *navigacija.php* na početku i *footer.php* na kraju skripte. Pozivom *navigacija.php* skripte iz *index.php* u meniju se označava zelenom bojom da je aktivna početna strana (slika 12).

navigacija.php

```
<!-- navigacija.php skripta koja sadži zaglavlje html fajla
-- koje će biti uključeno u različite skripte koje ga
-- pozivaju index.php, anketa.php, rezultati.php, login.php...-->
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Superglobana promenljiva
$_SERVER">
  <meta name="keywords" content="PHP, SERVER">
  <title>Primer 7: $_SERVER</title>
<style>
body {
  margin: 0;
  font-family: Arial, Helvetica, sans-serif;
}
main {
  margin: 30px;
}
.topnav {
  overflow: hidden;
  background-color: #333;
}

.topnav a {
  float: left;
```

```

color: #f2f2f2;
text-align: center;
padding: 14px 16px;
text-decoration: none;
font-size: 17px;
}

```

```

.topnav a:hover {
    background-color: #ddd;
    color: black;
}

```

```

.topnav a.active {
    background-color: #04AA6D;
    color: white;
}

```

```
</style>
```

```
</head>
```

```
<body>
```

```
<?php
```

// Preuzimanje informacija o fajlu iz koga je pozvana navigacija

```
$putanja = $_SERVER["SCRIPT_NAME"];
```

// Poslednja pozicija znaka '/'

```
$pozicija=strrpos($_SERVER["SCRIPT_NAME"], "/");
```

```
$naziv_fajla=substr($_SERVER["SCRIPT_NAME"], $pozicija + 1);
```

```
?>
```

<!--Podela delova HTML koda se vrši pomoću oznake div

-- za svaki deo je pripremljeno odgovarajuće formatiranje -->

```
<div class="topnav">
```

<!-- Umetnuti PHP kod služi da postavi klasu aktive za onaj element

-- navigacije koji upućuje na stranicu iz koje je pozvana navigacija-->

```
<a class=" <?php if ($naziv_fajla === "index.php") echo 'active'; ?>"
href="index.php"> Početna</a>
```

```
<a class=" <?php if ($naziv_fajla === "anketa.php") echo 'active'; ?>"
href="anketa.php"> Anketa</a>
```

```
<a class=" <?php if ($naziv_fajla === "rezultati.php") echo 'active'; ?>"
href="rezultati.php">Rezultati ankete</a>
```

```
<a class=" <?php if ($naziv_fajla === "login.php") echo 'active'; ?>"
href="login.php"> Prijavi se </a>
</div>
```

footer.php

<!-- footer.php – zajedničko podnožje svake stranice-->

```
<footer>
  <hr>
  <div>
    <p><a href="#">Nazad na vrh</a></p>
    <p>© 2021 <a href="mailto:umarovac@np.ac.rs"> Kontakt </a> </p>
  </div>
</footer>
</body>
</html>
```

index.php

*<!-- index.php početna strana koja se podrazumevano poziva ako
-- nije navedena druga konkretna skripta.
-- Naredbom include uključuje se sadržaj iz programa navigacija.php-->*

```
<?php include "navigacija.php" ?>
```

*<!-- Glavni deo prikazuje sadržaj glavne strane koji odnosi na različite
-- tipove programskih jezika koji se koriste za pravljenje dinamičkih veb
-- stranica na serverskoj strani -->*

```
<main >
  <div>
    <div >
      <h1>Ulogujte se i popunite anketu.</h1>
      <p><a href="login.php" >Ulogujte se</a></p>
    </div>
  </div>
  <div >
    <div >
      <div >
```



```
<h2>Server-Side script </h2>
<p> Skup instrukcija koje interpretira server i na osnovu njih generiše
HTML kod. </p>
<p> Rezultujući HTML kod se šalje klijentu kao deo HTTP
odgovora.</p>
</div>
<div >
<h3>PHP</h3>
<p> PHP- se ugrađuje u HTML kod i interpretira na strani servera od
strane PHP interpretatora</p>
</div>
<div >
<h3>ASP</h3>
<p>ASP je Microsoft-ova tehnologija koja omogućava slanje dinamičkih
sadržaja klijentu kao što su XHTML, ActiveX kontrole, klijentske skripte i
Java apleti.</p>
</div>
<div >
<p>...</p>
</div>
</div>
</div>
</main>
```

```
<!-- Naredba require uključuje sadržaj iz programa isto kao include
-- ali u slučaju greške izbacuje poruku i prekida program -->
<?php require "footer.php" ?>
```



Slika 12. Prikaz programa kada se pozove *index.php* aktivno polje je “Početna”

Primer 8. Prosleđivanje argumenta pomoću GET metode

Superglobalna promenljiva `$_GET` čuva podatke koji se prosleđuju serveru putem HTTP zahteva sa GET metodom. Imena podataka koji se prosleđuju su data parametrom *name*, a vrednost se preuzima iz odgovarajućeg polja forme. U narednom primeru (slika 13) prikazana je skripta *login.php* koja za izvršenje forme koristi metodu GET i prosleđuje podatke ponovo skripti *login.php*. Validacija podataka se obavlja pomoću JavaScript funkcije i ukoliko podaci nisu ispravno uneti forma se ne izvršava (slika 14). Nakon uspešnog unosa podataka ispisuju se podaci o trenutno ulogovanom korisniku (slika 15). Za ovaj zadatak se koriste skripte *navigacija.php* i *footer.php* (primer 7).

login.php

```
<!-- Uključivanje skripte navigacija.php iz primera 7. -->
<?php include "navigacija.php" ?>
```

```

<main>
<?php

// Promenljiva koja pamti stanje da li su podaci uspešno prosleđeni
$poslato = false;

/* Superglobalna promenljiva $_GET je asocijativni niz
 * indeksi niza su vrednosti parametra name polja koja su prosleđena
 * i čije vrednosti čuvaju. */
if(isset($_GET['korisnickoIme']) && isset($_GET['lozinka']))
{
    $poslato = true;
    $korisnik=$_GET['korisnickoIme'];
}
?>
<div>
    <h1>Prijava</h1><br>

    <!-- method="GET" govori pregledaču da vrednosti koje korisnik upiše
    -- u formular prosleđuje HTTP metodom GET kroz URL
    -- pritiskom na submit dugme poziva se program zadat kao action sa
    -- prosleđenim argumentima tako na primer ako u polje korisničko ime
    -- upišete Paja a u polje lozinka Pata nakon submита dobijate
    -- login.php?korisnickoIme=Paja&lozinka=Pata-->
    <form action="login.php" method="GET" onsubmit="return provera()">
        <div >
            <label>Korisničko ime</label>
            <input type="text" name="korisnickoIme" id="username" autofocus>
        </div>
        <div >
            <label>Lozinka</label>
            <input type="password" name="lozinka" id="password" >
        </div>

        <!-- Ukoliko je skripta pozvana sa prosleđenim podacima ispisuju se
        -- podaci koje je korisnik uneo-->
        <?php
            if ($poslato)
                echo "<p>Trenutno je poslat zahtev za logovanje korisnika
        <b>$korisnik</b> </p>";
            else
                echo "<p>Trenutno nema poslatih zahteva za logovanje</p>";

```

?>

*<!--Ukoliko je korisnik zaboravio lozinku poziva se skripta
-- za resetovanje koja će biti predstavljena u primeru 20.
-->*

`Zaboravio sam lozinku

<button type="submit" >Prijavi se</button>
</form>`

*<!-- Za validaciju polja koristi se JS
-- funkcija proverava da li su polja username i password popunjena
-- i daje obaveštenje o nepopunjenom polju -->*

```
<script>
function provera()
{
    var username = document.getElementById('username').value.trim();
    var password = document.getElementById('password').value.trim();

    if(username.length == 0)
    {
        alert("Nije uneto korisničko ime!");
        return false;
    }

    if(password.length == 0)
    {
        alert("Nije uneta lozinka!");
        return false;
    }

    return true;
}
</script>
<br>Nemate nalog? <a href="registracija.php">Registrujte se</a><br>
</div>
</main>
```

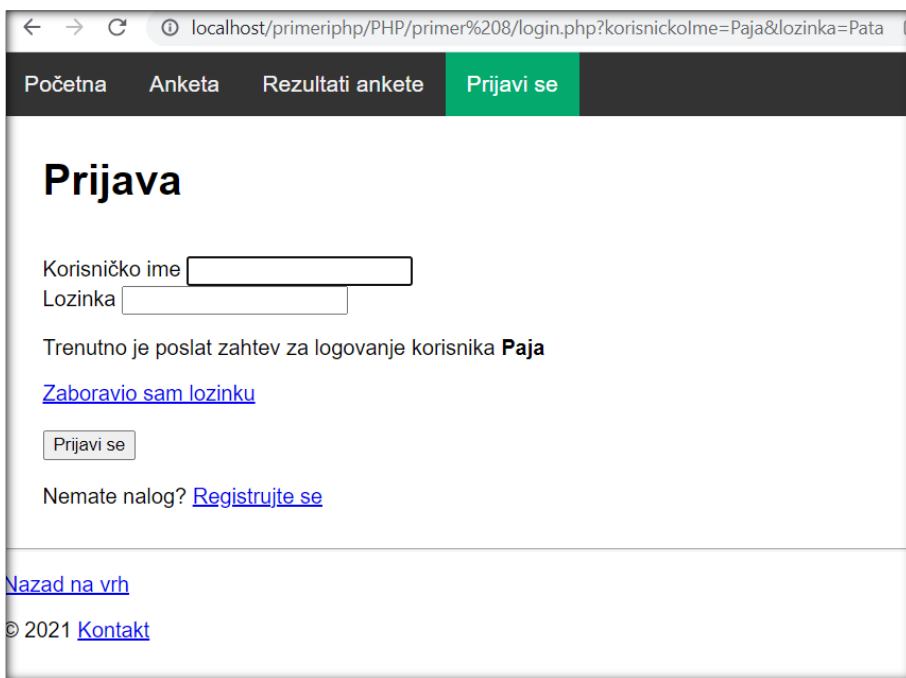
<!-- Uključivanje skripte footer.php iz primera 7. -->
`<?php include "footer.php" ?>`

The screenshot shows a web application with a dark navigation bar at the top containing links: Početna, Anketa, Rezultati ankete, and Prijavi se (highlighted in green). The main content area is titled 'Prijava'. It contains two input fields for 'Korisničko ime' and 'Lozinka'. Below the fields, it states 'Trenutno nema poslatih zahteva za logovanje' and provides a link 'Zaboravio sam lozinku'. A 'Prijavi se' button is present. At the bottom, it asks 'Nemate nalog?' and provides a link 'Registrujte se'. A footer section includes a link 'Nazad na vrh' and copyright information '© 2021 Kontakt'.

Slika 13. Početni prikaz formulara za logovanje, nisu prosleđeni podaci za logovanje (*login.php*)

This screenshot shows the same login page as Slika 13, but with an error message displayed in a white box with a blue border. The message reads: 'localhost наводи: Nije uneto korisničko ime!'. A blue button labeled 'Потврди' is located to the right of the message. The input fields for 'Korisničko ime' and 'Lozinka' are still present, with the password field showing masked characters (dots). The rest of the page content, including the navigation bar and footer, remains the same.

Slika 14. Upozorenje ukoliko podaci nisu prosleđeni



Slika 15. Prosledeni podaci GET metodom gde je URL :
<http://localhost/login.php?korisnickoIme=Paja&lozinka=Pata>

Primer 9. Prosleđivanje argumenta pomoću POST metode

Superglobalna promenljiva `$_POST` služi da se smeste vrednosti promenljivih prosleđenih HTTP metodom POST koja za razliku od GET metode podatke šalje unutar HTTP zahteva kao dodatne informacije. U narednom programu prikazana je skripta *anketa.php* koja je jedna od stavki navigacije u primeru 7. Ova skripta prikazuje formu za anketiranje korisnika kako bi se utvrdilo njihovo interesovanje za programiranje (slika 16). Validacija ankete delom vrši na klijentskoj strani (slika 17), a delom se šalje post metodom na serversku stranu gde se vrše provere da li je korisnik nasumično unosio podatke ili je dosledno unosio vrednosti pomoću skripte *anketaval.php* (slika 18, slika 19, slika 20. i slika 21).

anketa.php

```
<?php include "navigacija.php" ?>
<main>
  <div align=center>
    <h1>Anketa</h1>
  </div>
```

```

<!-- Forma sa pitanjima za anketiranje
-- validacija forme biće izvršena delimično uz pomoć JS funkcije
-- validacija()
-- akcija nakon izvršenja forme je PHP skripta anketaval.php koja
-- proverava da li je korisnik ispravno popunio anketu i da li ima
-- nelogičnosti forma je prosledjena POST metodom -->
<form action="anketaval.php" method="POST" name="anketa"
id="anketa" onsubmit="return validacija()">

```

```

<!-- prvo pitanje -->

```

```

<div>
<h3><b>1. Da li volite programiranje?</b></h3>
<input type="radio" name="pitanje1" value="Da" > Da<br>
<input type="radio" name="pitanje1" value="Ne" > Ne<br>

```

```

<!-- Polje p1Err sadrži obaveštenje ukoliko nije validno popunjeno
-- pitanje 1. -->

```

```

<i style="color:Tomato;" id="p1Err" ><b></b></i>
</div>
<hr>

```

```

<!-- Drugo pitanje -->

```

```

<div>
<h3><b>2. Koliko vremena provodite za računarom?</b></h3>
<select name="pitanje2" id="lista" >
  <option disabled selected value>-Odaberite jedan odgovor-</option>
  <option value="1">Manje od dva sata nedeljno</option>
  <option value="2">Manje od dva sata dnevno</option>
  <option value="3">Od dva do četiri sata dnevno</option>
  <option value="4">Od četiri do osam sati dnevno</option>
  <option value="5">Preko osam sati dnevno</option>
</select><br>

```

```

<!-- Polje p2Err sadrži obaveštenje ukoliko nije validno popunjeno
-- pitanje 2. -->

```

```

<i style="color:Tomato;" id="p2Err"></i>
</div>
<hr>

```

```

<!-- Treće pitanje -->

```

```

<div>
<h3><b>3. Ocenite vaše programersko iskustvo?</b></h3>

```

```
<select name="pitanje3" >
  <option disabled selected value>-Odaberite jedan odgovor-</option>
  <option value="1">1</option>
  <option value="2">2</option>
  <option value="3">3</option>
  <option value="4">4</option>
  <option value="5">5</option>
</select><br>
```

<!-- Polje p3Err sadrži obaveštenje ukoliko nije validno popunjeno -- pitanje 3. -->

```
<i style="color:Tomato;" id="p3Err"></i>
</div>
<hr >
```

<!-- Četvrto pitanje -->

```
<div >
  <h3><b>4.Koje programske jezike koristite?</b></h3>
  <p><small>(moguće je više odgovora)</small></p>
  <input type="checkbox" name="pitanje4[]" value="Py" > Python
  <input type="checkbox" name="pitanje4[]" value="JS" > JavaScript
  <input type="checkbox" name="pitanje4[]" value="PHP" > PHP
  <input type="checkbox" name="pitanje4[]" value="C" > C
  <input type="checkbox" name="pitanje4[]" value="Java" > Java
  <input type="checkbox" name="pitanje4[]" value="C++" > C++
  <input type="checkbox" name="pitanje4[]" value="C#" > C#<br>
  <input type="checkbox" name="pitanje4[]" value="x" > Nijedan od
navedenih<br>
</div>
```

<!-- Polje p4Err sadrži obaveštenje ukoliko nije validno popunjeno -- pitanje 4. -->

```
<i style="color:Tomato;" id="p4Err"></i>
<hr>
```

<!-- Komentar -->

```
<div >
  <h3><b>Komentar</b></h3><br>
  <textarea name="komentar" rows="4" cols="50" placeholder="Opišite
Vaše programersko interesovanje."></textarea>
  <p><small> Komentari koji su upisani samo velikim ili samo malim
slovima biće označeni kao nevalidni </small></p>
```



```
</div>
```

```
<!-- Polje komErr sadrži obaveštenje ukoliko nije validno popunjen  
-- komentar. -->
```

```
<i style="color:Tomato;" id="komErr"></i>
```

```
<hr>
```

```
<!-- Dugme potvrdi -->
```

```
<div >
```

```
<button type="submit" name="posalji_anketu" value="Submit">Pošalji
```

```
</button>
```

```
</div>
```

```
</form>
```

```
<script>
```

```
// Funkcija za štampanje greške ispod odgovarajućeg polja za unos u formi
```

```
function stampajGresku(elemId, hintMsg)
```

```
{
```

```
/* U polje forme sa odgovarajućim id-jem biće ubačena poruka*/
```

```
document.getElementById(elemId).innerHTML = hintMsg;
```

```
}
```

```
//Funkcija za proveru da li su uneti svi odgovori
```

```
function validacija(){
```

```
//Preuzimaju se vrednosti iz elemenata forme za svako pitanje
```

```
var p1=document.anketa.pitanje1.value;
```

```
var p2=document.anketa.pitanje2.value;
```

```
var p3=document.anketa.pitanje3.value;
```

```
var kom=document.anketa.komentar.value;
```

```
//Postavlja se greška za svako pitanje na netačno
```

```
var p1Err=p2Err=p3Err=p4Err=komErr=false;
```

```
//Ako nije izabran odgovor označi kao greška i stampaj upozorenje
```

```
if(p1==""){
```

```
stampajGresku("p1Err","Odaberite jedan odgovor!");
```

```
p1Err=true;
```

```
}
```

```
if(p2==""){
```

```
stampajGresku("p2Err","Odaberite jedan odgovor!");
```

```
p2Err=true;
}
if(p3==""){
    stampajGresku("p3Err","Odaberite jedan odgovor!");
    p3Err=true;
}
```

/ Pitanje 4. čuva odgovore kao niz pa se prebrojava*

** Koliko je odgovora označeno*

** Ukoliko nije ni jedan postavlja se poruka o grešci */*

```
var count=0;
var nBox = document.getElementsByName('pitanje4[]');
for (i=0; i<nBox.length; i++)
{
    if (nBox[i].checked == true){
        count++;
        break;
    }
}
if (count == 0)
{stampajGresku("p4Err","Morate odabrati makar jedan odgovor!");
p4Err=true;
}
```

// Komentar mora da ima bar dva karaktera

```
if(kom.length<2){
    stampajGresku("komErr", "Morate uneti makar dva karaktera.");
    komErr = true;
}
```

//Ako postoji greška neće se izvršavati forma

```
if((p1Err || p2Err || p3Err || p4Err || komErr )==true){return false;}
else
    return true;
}
</script>
</main>
<?php include "footer.php" ?>
```

 anketaval.php

```

<?php include "navigacija.php" ?>
<main>
<?php

// Promenljive u kojima se čuvaju prosledene vrednosti
$p1 = $p2= $p3 = "";
$p4[]="";
$komentar="";

/* Funkcija za proveru validnosti komentara da li je ispisan
 * samo velikim odnosno malim slovima*/
function proveru_validnost($k)
{

/* Za proveru validnosti korišćeni su regularni izrazi:
 * \p{xx} - izdvaja karaktere sa svojstvom xx
 * \P{xx} - izdvaja karaktere koji nemaju svojstvo xx
 * L - oznaka za slovne karaktere
 * /u - upozorenje za nevalidne UTF-8 karaktere
 * preg_replace - vrši zamenu svakog karaktera koji odgovara uzorku
 * sa navedenim karakterom u našem slučaju praznim */
$komentar_samo_slova=preg_replace('/\PL/u', '', $k);

/* Provera da li je poruka ispisana samo velikim ili malim slovima
 * vrši se pomoću funkcija za proveru tipa karaktera:
 * ctype_upper - proveru da li su karakteri veliki
 * ctype_lower - proveru da li su karakteri mali */
if(ctype_upper($komentar_samo_slova) ||
ctype_lower($komentar_samo_slova))
{
    return 0;
}
else
{
    return 1;
}
}

//Postavlja se promenljiva o validno popunjenoj anketi na tačno
$validno_popunjeno=true;

```

```

/* Ako je forma izvršena preuzimaju se vrednosti odgovarajućih
 * promenljivih iz superglobalne promenljive $POST */
if(isset($_POST['posalji_anketu']))
{
    $p1 = $_POST["pitanje1"];
    $p2 = $_POST["pitanje2"];
    $p3 = $_POST["pitanje3"];
    $p4= implode(",",$_POST["pitanje4"]);
    $komentar = $_POST["komentar"];
    $p4_length= count($_POST["pitanje4"]);

    /* Ako polja nisu prazna proverava se njihova validnost i
     * da li su nasumično popunjena */
    if(!empty($p1) && !empty($p2) && !empty($p3) && !empty($p4) &&
        !empty($komentar) )
    {

        // Funkcijom proveri_komentar ispituje se da li je komentar validan
        $validno_popunjeno=proveri_validnost($komentar);

        /* Proverava se da li je korisnik na četvrto pitanje
         * izabrao poslednju opciju "Nijedan od navedenih"
         * i još neki programski jezik ( kontradiktorno->NEVALIDNO)*/
        if($p4_length>1 and in_array("x", $_POST["pitanje4"]))
            $validno_popunjeno=false;

        /* Proverava se da li je korisnik odgovorio da zna solidno
         * programiranje (ocene 3 i više), a pri tom ne zna
         * makar dva od navedenih jezika i
         * nije za računarom makar 2h do 4h sata dnevno
         * (kontradiktorno->NEVALIDNO)*/
        if(intval($p3)-intval($p2)>2 or (in_array("x", $_POST["pitanje4"]) and
            intval($p3)>1 ) or intval($p3)-2>$p4_length ){
            $validno_popunjeno=false; }

        /* Ispisuje se poruka da li je uspešno popunjena anketa */
        if($validno_popunjeno)
            echo"<h4>Uspešno ste popunili anketu</h4>";
        else
            echo"<h4>Anketa nije uspešno popunjena</h4>";
        }
    }
}

```

```
?>
</main>
<?php include "footer.php" ?>
```

Anketa

1. Da li volite programiranje?

☐ Da
☐ Ne

2. Koliko vremena provodite za računarom?

3. Ocenite vaše programersko iskustvo?

4. Koje programske jezike koristite?

(moguće je više odgovora)

☐ Python ☐ JavaScript ☐ PHP ☐ C ☐ Java ☐ C++ ☐ C# ☐ Nijedan od navedenih

Komentar

Komentari koji su upisani samo velikim ili samo malim slovima biće označeni kao nevalidni

Slika 16. Prikaz izgleda ankete pre unosa podataka (*anketa.php*)

Anketa

1. Da li volite programiranje?

☐ Da
☐ Ne

Odaberite jedan odgovor!

2. Koliko vremena provodite za računarom?

Odaberite jedan odgovor!

3. Ocenite vaše programersko iskustvo?

Odaberite jedan odgovor!

4. Koje programske jezike koristite?

(moguće je više odgovora)

☐ Python ☐ JavaScript ☐ PHP ☐ C ☐ Java ☐ C++ ☐ C# ☐ Nijedan od navedenih

Morate odabrati makar jedan odgovor!

Komentar

Komentari koji su upisani samo velikim ili samo malim slovima biće označeni kao nevalidni

Morate uneti makar dva karaktera.

Slika 17. Validacija na klijentskoj strani pomoću funkcije `proveri_validnost()`

Anketa

1. Da li volite programiranje?

☒ Da
☐ Ne

2. Koliko vremena provodite za računarom?

3. Ocenite vaše programersko iskustvo?

4. Koje programske jezike koristite?

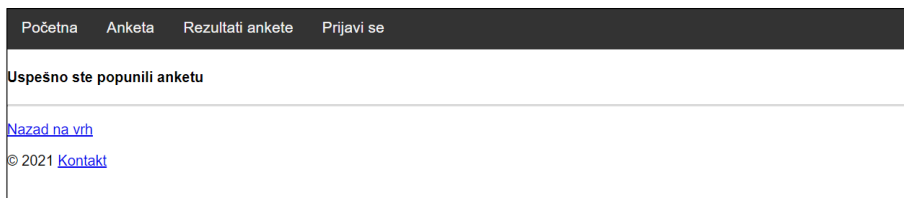
(moguće je više odgovora)

☐ Python ☒ JavaScript ☒ PHP ☐ C ☒ Java ☒ C++ ☒ C# ☐ Nijedan od navedenih

Komentar

Komentari koji su upisani samo velikim ili samo malim slovima biće označeni kao nevalidni

Slika 18. Validno uneti podaci u anketu



Početna Anketa Rezultati ankete Prijavi se

Uspešno ste popunili anketu

[Nazad na vrh](#)

© 2021 [Kontakt](#)

Slika 19. Odgovor serverske strane nakon validno unetih podataka (*anketaval.php*)

Primetimo da navigacija kada se poziva iz skripte *anketaval.php* nema ni jedno polje aktivno (označeno zelenom bojom) jer za skriptu *anketaval.php* nije navedeno koje polje treba naznačiti.



Anketa

1. Da li volite programiranje?

☐ Da
☒ Ne

2. Koliko vremena provodite za računarom?

Od dva do četiri sata dnevno

3. Ocenite vaše programersko iskustvo?

3

4. Koje programske jezike koristite?

(moguće je više odgovora)

☐ Python ☐ JavaScript ☐ PHP ☐ C ☐ Java ☐ C++ ☐ C# ☒ Nijedan od navedenih

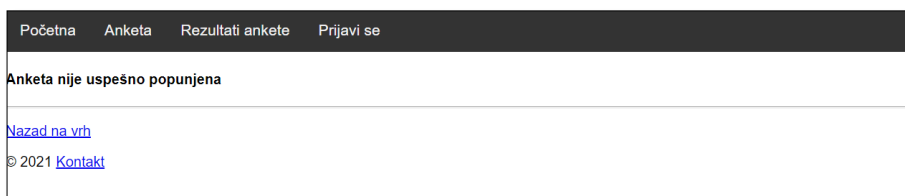
Komentar

Nisam obiljan

Komentari koji su upisani samo velikim ili samo malim slovima biće označeni kao nevalidni

Pošalji

Slika 20. Nevalidno uneti podaci u anketu (*anketa.php*)



Slika 21. Odgovor servera nakon nevalidno unetih podataka (*anketaval.php*)

Primer 10. Nizovi u PHP-u

Nizovi u PHP su složeni objekti. To su uređene kolekcije parova ključ/vrednost. Za rad sa nizovima obezbeđene su brojne funkcije i operatori. Sledećim primerom je prikazan rad sa nizovima u PHP-u i rezultati ovog programa prikazani su na slici 22.

primer10.php

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Nizovi u PHP-u">
  <meta name="keywords" content="PHP, ARRAY">
  <title>Primer 10: PHP nizovi</title>
  <style>
    main{
      margin: 30px;
      font-size: 17px;
    }
    table{
      height:100%;
      width:100%;
      margin: 15px;
    }
    tr {
      border-bottom: 3px solid;
    }
    th,td {
      vertical-align: top;
      text-align: left;
      width: 30% ;
    }
  </style>
</head>
<body>
```



```

    th {
        height: 20% ;
    }
</style>
</head>
<body>
    <main>
    <?php

```

/ Prikaz rezultata funkcija za rad sa nizovima se štampa u*

** HTML tabeli */*

```
echo "<table><tr><th>";
```

//Prvi red tabele zaglavlja

```
echo '</br></br><b style="color: red;"> Kreiranje niza i funkcija count
    </b></br></br>';
```

```
echo "</th><th>";
```

```
echo '</br></br><b style="color: red;"> Petlja za rad sa kolekcijama -
    foreach</b></br></br>';
```

```
echo "</th><th>";
```

```
echo '</br></br><b style="color: red;"> Funkcije explode i
each</b></br></br>';
```

```
echo "</th></tr>";
```

//Drugi red rezultati odgovarajućih funkcija navedenih u zaglavlju

```
echo "<tr><td>";
```

/ Kreiranje niza programskih jezika*

** count - daje broj elemenata niza*

** štampaju se svi elementi niza*

**/*

```
$p_jezici=array("C","C#","C++","Java","PHP","JavaScript");
```

```
$count=count($p_jezici);
```

```
echo "<i>Broj elemenata u nizu \$p_jezici je $count </i></br></br>";
```

```
for($i=0;$i<$count;$i++){
```

```
    echo "p_jezici[$i]=$p_jezici[$i]</br>";
```

```
}
```

```
echo "</td><td>";
```

/ Petlja foreach dozvoljava vam da prolazite kroz elemente niza*

** jednostavno */*

```
foreach($p_jezici as $jezik){
```

```
        echo "$jezik<br/>";
    }
    echo "</td><td>";

    /* Funkcijom explode rastavlja se string na elemente niza
    * pomoću zadatog operatora
    * each() iz niza $p_jezici izdvaja tekući element i interni pokazivač
    * pomera na naredni element */
    $jezici="C,C#,C++,Java,PHP,JavaScript";
    $p_jezici=explode(",",$jezici);
    echo "<i>Funkcijom explode od stringa </br>'C,C#,C++,Java,PHP,
    JavaScript ' </br> dobijen je niz elemenata: </i></br>";
    while($jezik=each($p_jezici)){
        echo $jezik['key'];
        echo "->";
        echo $jezik['value'];
        echo "<br/>";
    }
    echo "</br></td></tr>";
```

```
//Dugi red zaglavlja
echo "<tr><th>";
echo '<b style="color: red;"> Funkcije reset,list</b>';
echo "</th><th>";
echo '<b style="color: red;"> Da li se u nizu nalazi "C"? </br> funkcija
in_array
    </b>';
echo "</th><th>";
echo ' <b style="color: red;"> Niz svih indeksa elemenata
array_keys,array_search
    </b>';
echo "</br></th></tr><tr><td>";
```

//Rezultati funkcija navedeni u drugom redu zaglavlja

```
/* Postavlja se pokazivač na početak niza - reset
* koristi se funkcija list za smeštanje elemenata niza
* u listu elemenata $r_br, $jezik
* prikazuju se svi elementi redom
* na kraju vraća se pokazivač na početak niza - reset
*/
reset($p_jezici);
```

```

while(list($r_br, $jezik) = each($p_jezici)){
    echo "p_jezici[$r_br]=$jezik<br/>";
}
reset($p_jezici);
echo "</td><td>";

```

```

/* Funkcijom explode rastavlja se string na elemente niza pomoću
* zadatog operatora
* niz je kolekcija elemenata koji sadrže elemente key i value
* prikazuju se elementi niza
* in_array daje TRUE ako je vrednost datog elementa u datom nizu
* i FALSE u obrnutom slučaju
* proverava se da li se string "C" nalazi u nizu*/
echo "<i>Niz sadrži u vrednosti koje se pojavljuju više puta </i></br>";
$jezici="C,C,C#,C++,Java,C#,PHP,PHP,C,JavaScript";
$p_jezici_x=explode(",",$jezici);
while($jezik=each($p_jezici_x)){
    echo $jezik['key'];
    echo "->";
    echo $jezik['value'];
    echo "<br/>";
}
if (in_array("C",$p_jezici_x))
    echo "<i>String 'C' se nalazi u nizu \$p_jezici</i>";
else
    echo "<i></br>String 'C' se ne nalazi u nizu \$p_jezici</i>";
echo "</td><td>";

```

```

/* array_keys – daje niz svih indeksa elemenata,
* odnosno niz indeksa zadate vrednosti
* array_search(element,imeNiza) – daje vrednost prvog indeksa zadate
* vrednosti*/
echo '<i>Array_keys za element "C" </i></b></br>';
$keys=array_keys ($p_jezici_x,"C");
while($k=each($keys)){
    echo $k['key'];
    echo "->";
    echo $k['value'];
    echo "</br>";
}
echo '<i>Vrednost prvog indeksa element "C"</i> </br>';

```

```
echo array_search("C",$p_jezici_x) ;
echo "</td></tr></table>";
?>
</main>
</body>
```

Kreiranje niza i funkcija count	Petlja za rad sa kolekcijama-foreach	Funkcije explode i each
<p>Broj elemenata u nizu \$p_jezici je 6</p> <pre>p_jezici[0]=C p_jezici[1]=C# p_jezici[2]=C++ p_jezici[3]=Java p_jezici[4]=PHP p_jezici[5]=JavaScript</pre>	<pre>C C# C++ Java PHP JavaScript</pre>	<p>Funkcijom explode od stringa 'C,C#,C++,Java,PHP,JavaScript' dobijen je niz elemenata:</p> <pre>0->C 1->C# 2->C++ 3->Java 4->PHP 5->JavaScript</pre>
<p>Funkcije reset,list</p> <pre>p_jezici[0]=C p_jezici[1]=C# p_jezici[2]=C++ p_jezici[3]=Java p_jezici[4]=PHP p_jezici[5]=JavaScript</pre>	<p>Da li se u nizu nalazi "C"? funkcija in_array</p> <p>Niz sadrži u vrednosti koje se pojavljuju više puta</p> <pre>0->C 1->C 2->C# 3->C++ 4->Java 5->C# 6->PHP 7->PHP 8->C 9->JavaScript</pre> <p>String 'C' se nalazi u nizu \$p_jezici</p>	<p>Niz svih indeksa elemenata array_keys,array_search</p> <p>Array_keys za element "C"</p> <pre>0->0 1->1 2->8</pre> <p>Vrednost prvog indeksa element "C"</p> <pre>0</pre>

Slika 22. Rad sa nizovima (primer10.php)

Primer 11. Asocijativni nizovi, sortiranje i druge funkcije za rad sa nizovima

PHP podržava i nizove u kojima su vrednosti indeksa tipa string, takozvani asocijativni nizovi. Narednim primerom prikazan je rad sa ovakvim nizovima kao i neke funkcije za sortiranje i uređivanje rasporeda elemenata u nizu (slika 23).

primer11.php

```
<!DOCTYPE html>
<?php
```

// Funkcija shuffle za asocijativne nizove

```
function shuffle_assoc($list) {
```

```
    //Proverava da li je list niz, ako nije vraća ga kao rezultat f-je  
    if (!is_array($list)) return $list;
```

```
    //Izdvaža ključeve niza u poseban niz  
    $keys = array_keys($list);
```

```
    //Pomoću shuffle f-je vrši se nasumično raspoređivanje elemenata u nizu  
    shuffle($keys);  
    $random = array();
```

```
    //Pravi novi asocijativni niz sa nasumičnim rasporedom  
    foreach ($keys as $key)  
        $random[$key] = $list[$key];  
    return $random;  
}
```

```
?>
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <meta name="autor" content="Ulfeta Marovac">
```

```
    <meta name="description" content="Nizovi u PHP-u">
```

```
    <meta name="keywords" content="PHP, ARRAY">
```

```
    <title> Primer 11: PHP sortiranje nizova </title>
```

```
    <style>
```

```
        main{
```

```
            margin: 30px;
```

```
            font-size: 17px;
```

```
        }
```

```
        table{
```

```
            height:100%;
```

```
            width:100%;
```

```
            margin: 15px;
```

```
        }
```

```
        tr {
```

```
            border-bottom: 3px solid;
```

```
        }
```

```
        th,td {
```

```
            vertical-align: top;
```

```
            text-align: left;
```

```

        width: 30% ;
    }
    th {
        height: 20% ;
    }
</style>
</head>

```

```

<body style="font-size:130%;">
<main>

```

<!-- Tabela za prikaz rada sa nizovima-->

```

<table>
<tr>
<th><b>Asocijativni niz</b></th>
<th><b>Sortiranje niza</b></th>
</tr>

```

```

<?php
echo "<tr><td>";

```

//Kreiranje i štampanje asocijativnih nizova

/ Kreiranje:*

** niza programskih jezika sa string indeksima*
** niza sa indeksima za programske jezike*
** niza programskih jezika sa numeričkim indeksima */*

```

$p_jezici=array(          "klijentska"=>"JS",          "staticke"=>"HTML",
"serverska"=>"PHP");
$p_jezici_key=array("klijentska","staticke","serverska");
$p_jezici_val=array("JS","HTML","PHP");

```

/ Petljom foreach prolazi se kroz niz gde i svaki element se preslikava u*

** kljuc(\$x)=>vrednost(\$x_value).*
** Kada se pomoću echo štampa string koji sadrži navodnike*
** koriste se apostrofi za početak i kraj odgovarajućeg stringa */*

```

echo '</br></br><b style="color: red;">Štampanje niza (p_jezici)
kljuc=>vrednost </b></br><hr>';
foreach($p_jezici as $x => $x_value) {
    echo "Key=" . $x . ", Value=" . $x_value;

```

```

    echo "<br>";
}

// Funkcija print_r štampa niz sa svim njegovim osobinama
echo ' </br></br><b style="color: red;" >Štampanje sa
print_r:</b></br>';
print_r($p_jezici);

/* Prolaz kroz niz sa numeričkim indeksima gde se svaki element preslikava
 * u ključ($x)=>vrednost($x_value)
 * print_r štampa niz*/
echo ' </br></br><b style="color: red;">Štampanje niza sa numeričkim
indeksom (p_jezici_val) ključ=>vrednost </b></br><hr>';
foreach($p_jezici_val as $x => $x_value) {
    echo "Key=" . $x . ", Value=" . $x_value;
    echo "<br>";
}
echo ' </br></br><b style="color: red;">Štampanje sa print_r niza
(p_jezici_key)</b></br><hr>';
print_r($p_jezici_key);
echo "</td><td>";

/* Sortiranje nizova po ključu i vrednosti
 * sortira u rastućem redosledu parove ključ vrednost */
echo ' </br></br><b style="color: red;" >asort</b></br><hr>';
asort($p_jezici, SORT_STRING);
print_r($p_jezici);

// Sortira u opadajućem redosledu parove ključ vrednost
echo ' </br></br><b style="color: red;">arsort</b></br><hr>';
arsort($p_jezici, SORT_STRING);
print_r($p_jezici);

// Sortira u rastućem redosledu po ključu
echo ' </br></br><b style="color: red;">ksort</b></br><hr>';
ksort($p_jezici, SORT_STRING);
print_r($p_jezici);

// Sortira u opadajućem redosledu po ključu
echo ' </br></br><b style="color: red;">krsort</b></br><hr>';
krsort($p_jezici, SORT_STRING);

```

```
print_r($p_jezici);
```

// Obrnuti redosled elemenata

```
echo '</br></br><b style="color: red;">array_reverse</b></br><hr>';  
print_r(array_reverse($p_jezici));  
echo "</td></tr><tr>";
```

// Funkcije niza

```
echo "<th></br></br><b>Funkcije niza</b></th><tr><td>";
```

// Zamena uloge indeksima i vrednostima

```
echo '</br></br><b style="color: red;">array_flip asocijativnog  
niza</b></br><hr>';  
print_r(array_flip($p_jezici));
```

```
echo '</br></br><b style="color: red;">array_flip niza sa numerickim  
indeksima</b></br><hr>';  
print_r(array_flip($p_jezici_val));
```

/ Funkcija koja kreira nov niz u kome vrednosti prvog*

** niza postaju indeksi drugog niza */*

```
echo '</br></br></br><b style="color: red;">array_combine nizova  
p_jezici_key, p_jezici_val </b></br><hr>';  
print_r(array_combine($p_jezici_key,$p_jezici_val));
```

/ Nasumično menja redosled elemenata u nizu */*

```
echo '</br></br><b style="color: red;">shuffle za asocijativni  
niz</b></br><hr>';  
print_r(shuffle_assoc($p_jezici));
```

```
echo "</td></tr><table>";
```

```
?>
```

```
</main>
```

```
</body>
```

Asocijativni niz	Sortiranje niza
Štampanje niza (p_jezici) ključ=>vrednost	asort
Key=klijentska, Value=JS Key=statičke, Value=HTML Key=serverska, Value=PHP	Array ([statičke] => HTML [klijentska] => JS [serverska] => PHP)
	arsort
Štampanje sa print_r: Array ([klijentska] => JS [statičke] => HTML [serverska] => PHP)	Array ([serverska] => PHP [klijentska] => JS [statičke] => HTML)
	ksort
Štampanje niza sa numeričkim indeksom (p_jezici_val) ključ=>vrednost	Array ([klijentska] => JS [serverska] => PHP [statičke] => HTML)
Key=0, Value=JS Key=1, Value=HTML Key=2, Value=PHP	krsort
	Array ([statičke] => HTML [serverska] => PHP [klijentska] => JS)
Štampanje sa print_r niza (p_jezici_key)	array_reverse
Array ([0] => klijentska [1] => statičke [2] => serverska)	Array ([klijentska] => JS [serverska] => PHP [statičke] => HTML)
Funkcije niza	
array_flip asocijativnog niza	
Array ([HTML] => statičke [PHP] => serverska [JS] => klijentska)	
array_flip niza sa numeričkim indeksima	
Array ([JS] => 0 [HTML] => 1 [PHP] => 2)	
< b style="color: red;">array_combine nizova p_jezici_key, p_jezici_val	
Array ([klijentska] => JS [statičke] => HTML [serverska] => PHP)	
shuffle za asocijativni niz	
Array ([serverska] => PHP [statičke] => HTML [klijentska] => JS)	

Slika 23. Kreiranje, prikazivanje, sortiranje i funkcije sa nizovima sa numeričkim i asocijativnim indeksima (*primer11.php*)

Primer 12. Znakovni nizovi u PHP-u

Znakovni nizovi u veb programiranju su jako česti pa se u PHP-u koriste čak u dva oblika kao nizovi karaktera između ' ili ". PHP pruža operatore i funkcije za upoređivanje, rastavljanje, sklapanje, pretraživanje kao i mnoštvo specijalizovanih string funkcija. U narednom primeru su prikazane neke od ovih funkcija i demonstriran je rad sa stringovima (slika 24).

primer12.php

```
<!DOCTYPE html>

<head>
  <meta charset="UTF-8">
  <meta name="autor" content="Ulfeta Marovac">
  <meta name="description" content="Znakovni nizovi u PHP-u">
  <meta name="keywords" content="PHP, ARRAY, STRING">
  <title>Primer 10: PHP nizovi</title>
  <style>
    main{
      margin: 30px;
      font-size: 17px;
    }
    table{
      height:100%;
      width:100%;
      margin: 15px;
    }
    tr{
      border-bottom: 3px solid;
    }
    th,td{
      vertical-align: top;
      text-align: left;
      width: 30% ;
    }
    th {
      height: 20% ;
    }
  </style>
</head>
<body style="font-size:120%;">
<main>
<?php
```

// Štampanje stringa sa belinama koje sadrži

```
$str = "      A string is series of characters, where a character is the same as  
a byte.      ";  
echo "<pre>str:-$str-</pre>";
```

// Funkcija trim()briše beline sa početka i kraja znakovnog podatka

```
$str=trim($str);
echo "<pre>trim(str):-$str-</pre>";

// Funkcija explode koristeći string separator deli string u niz stringova
foreach( explode(" ", $str) as $x => $x_value) {
    echo "|". $x_value;
}
echo "<br/>";

// Funkcija ltrim – oduzima stringove sa leva (prvi - drugi)
echo "<pre>ltrim(str):".ltrim($str,"A str")."</pre>";

// Funkcija rtrim – oduzima stringove sa desna (prvi - drugi)
echo "<pre>rtrim(str):".rtrim($str,"byte.")."</pre>";

// Funkcija strtoupper - menja sva slova u velika
echo "<pre>strtoupper(str):".strtoupper($str)."</pre>";

// Funkcija strtolower - menja sva slova u mala
echo "<pre>strtolower(str):".strtolower($str)."</pre>";

/* Funkcija addslashes vraća string sa dodatim znakom \
 * pre znakova: ',",\,the NUL byte) .*/
$special_c=addslashes("single quote ('),double quote (\"),backslash (\), NUL
(the NUL byte)");
echo "<pre>addslashes(str):".$special_c."</pre>";

// Funkcija stripslashes uklanja kose crte, koje je postavila funkcija
addslashes
echo "<pre>stripslashes(str):".stripslashes($special_c)."</pre>";
?>
</main>
</body>
```

```
str:-          A string is series of characters, where a character is the same as a byte.

trim(str):-A string is series of characters, where a character is the same as a byte.-

|A|string|is|series|of|characters,|where|a|character|is|the|same|as|a|byte.

ltrim(str):ing is series of characters, where a character is the same as a byte.

rtrim(str):A string is series of characters, where a character is the same as a

strtoupper(str):A STRING IS SERIES OF CHARACTERS, WHERE A CHARACTER IS THE SAME AS A BYTE.

strtolower(str):a string is series of characters, where a character is the same as a byte.

addslashes(str):single quote (\'),double quote (\"),backslash (\\), NUL (the NUL byte)

stripslashes(str):single quote ('),double quote ("),backslash (\), NUL (the NUL byte)
```

Slika 24. Rad sa znakovnim nizovima (*primer12.php*)

Primer 13. Regularni izrazi

Primer 13. Za složenije pretraživanje uzoraka u stringu koriste se regularni izrazi. Postoje tri upotrebe regularnih izraza: podudaranje stringa sa zadatim uzorkom, zamena pronađenog uzorka u tekstu sa zadatim tekstom, i cepanje teksta u niz manjih delova. PHP ima funkcije za sve i narednim primerom će biti prikazane neke od njih. Rezultati primene regularnih izraza i funkcija za njihovu upotrebu nad tekstom prikazani su na slici 25.

primer13.php

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
    <meta name="autor" content="Ulfeta Marovac">
    <meta name="description" content="Uzorci u PHP-u">
    <meta name="keywords" content="PHP, regex, STRING">
    <title>Primer 10: PHP nizovi</title>
</style>
main{
  margin: 30px;
  font-size: 17px;
}
table{
  height:100%;
  width:100%;
```

```

        margin: 15px;
    }
    tr{
        border-bottom: 3px solid;
    }
    th,td{
        vertical-align: top;
        text-align: left;
        width: 30% ;
    }
    th {
        height: 20% ;
    }
</style>
</head>
<body style="font-size:120%;">
<main>
<?php

```

// Primeri uzoraka

// uzorak koji označava pojavljivanje jednog od tri stringa "php" ili "html" ili "JS"

```
$p1="/php/html/JS/";
```

/ Uzorak koji označava pojavljivanje jedne od cifara na početku 0,1,3*

** iza koje sledi bilo koja cifra*

** zatim 0 ili jedan pa 10 različitih cifara*

**/*

```
$JMBG="^d{13}/";
```

// Jedna ili više puta praznina, zarez, tačka ili uzvičnik

```
$separator="/[\\ \\.!]+/";
```

/ Uzorak za registraciju vozila: Dva slova velika slova,*

** crta opciono, tri cifre,crta opciono i dva velika slova**

```
$registracija="/[A-Z]{2}\-?\d{3}\-[A-Z]{2}/ ";
```

/ Niz karaktera na početku (slova, cifre i . _ + -), @, opet*

** (slova,cifre i . _ + -) i na kraju slovo ili cifra */*

```
$email="^[a-zA-Z0-9_\.|\+|-]+@[a-zA-Z0-9_\.|\+|-]*[a-zA-Z0-9]+$";
```

/ preg_grep(imeUzorka,imeNiza) – daje niz sa svim elementima niza koji u sebi sadrže uzorak*/*

```
$str="Za statičke veb strane koristimo html, za dinamičke php i JS.";
echo '<p>Uzorak /php/html/JS/ funkcijom preg_grep je pronađen u stringu:
</p><p>'.$str.'"</p>';
echo '<p>'.count(preg_grep($p1,explode(" ",$str)))." puta i to:</p>";
echo print_r(preg_grep($p1,explode(" ",$str)));
echo "<br>";
echo "<hr>";
```

/ preg_split(imeUzorka,imeStringa) – stvara niz delova teksta koji su razdvojeni zadatim uzorkom*/*

```
echo "<p>Podela striga na niz sa zadatim separatorom.</p>";
echo print_r(preg_split($separator,$str));
echo "<br>";
echo "<hr>";
```

/ preg_replace(imeUzorka,imeStringa,imeNiza) – zadati uzorak se zamenjuje stringom u datom nizu stringova (ili nizu) */*

```
$str="1111111111111111 0123456789333 AA090BB 4422255555129 AA-090-BB AA-090-23";
echo "Zamenimo pojavu JMBG-a sa ### u stringu: <br>\"$str\"<br>";
echo preg_replace($JMBG,"###",$str);
echo "<br>";
echo "<hr>";
```

```
echo "Zamenimo pojavu registracije sa *** u stringu: <br>\"$str\"<br>";
echo preg_replace($registracija,"***",$str);
echo "<br>";
echo "<hr>";
?>
</main>
</body>
```

Uzorak /php/html/JS/ funkcijom `preg_grep` je pronađen u stringu:

"Za statičke veb strane koristimo html, za dinamičke php i JS."

3 puta i to:

`Array ([5] => html, [8] => php [10] => JS.) 1`

Podela striga na niz sa zadatim separatorom.

`Array ([0] => Za [1] => statičke [2] => veb [3] => strane [4] => koristimo [5] => html [6] => za [7] => dinamičke [8] => php [9] => i [10] => JS [11] =>) 1`

Zamenimo pojavu JMBG-a sa ### u stringu:

"11111111111111 0123456789333 AA090BB 4422255555129 AA-090-BB AA-090-23"
###11 ### AA090BB ### AA-090-BB AA-090-23

Zamenimo pojavu registracije sa *** u stringu:

"11111111111111 0123456789333 AA090BB 4422255555129 AA-090-BB AA-090-23"
11111111111111 0123456789333 *** 4422255555129 *** AA-090-23

Slika 25. Pretraživanje uzoraka u stringu (*primer13.php*)

Sledeći primeri predstavljaju nastavak primera Primer 7. (8. i 9) i koristiće isto zaglavlje i podnožje uz nadogradnju po potrebi.

Primer 14. Super globalna promenljiva \$_FILES

Podaci o otpremljenom fajlu se nalaze u superglobalnoj promenljivoj `$_FILES`, koja se automatski inicijalizuje po prijemu podataka putem HTTP POST metode. Sledeći primer prikazuje otpremanje podataka za novog korisnika koji još uvek nema nalog. Na slici 13. (primer 8) je prikazana forma za logovanje u kojoj korisnik ima mogućnost registracije pritiskom na link "Registruj se" koji je veza do skripte *registracija.php*. Ovom skriptom se kreira forma (slika 26) za unos podataka o novom korisniku, koja se POST metodom prosleđuje skripti *login.php* (ova skripta je proširena skriptom *registracijavall.php* kojom se vrši validacija registracije, slika 27). POST metoda se koristi jer korisnik prilikom registracije mora da otpremi sliku kao dodatni fajl. Nakon uspešne registracije podaci o registrovanom korisniku i otpremljena slika su vidljivi na strani za prijavljivanje (slika 28).

navigacija.php

*<!--navigacija.php iz Primera 7. sa umetnutim delom kojim se njen poziv iz
-- skripte registracija.php označava kao aktivnost logovanja pa je u
-- meniju aktivno dugme "Prijavi se"*

-- prikazan je samo izmenjeni element <a>Prijavi se

-->

```
<a class=" <?php if (($naziv_fajla === "login.php") || ($naziv_fajla ===  
"registracija.php")) echo 'active'; ?>" href="login.php"> Prijavi se  
</a>
```

registracija.php

```
<?php include "navigacija.php" ?>
```

```
<main>
```

<!--Forma za unos podataka o novom korisniku

--enctype="multipart/form-data" omogućava upload fajlova na server

-- opcija required u input poljima označava da se ova polja moraju

-- popuniti -->

```
<form action="login.php" method="post" enctype="multipart/form-data">
```

```
<div>
```

```
</br></br>
```

```
<label>Ime*</label>
```

```
<input type="text" name="ime" placeholder="Ime" required>
```

```
</div>
```

```
<div>
```

```
</br>
```

```
<label>Prezime*</label>
```

```
<input type="text" name="prezime" placeholder="Prezime" required>
```

```
</div>
```

```
<div>
```

```
</br>
```

```
<label>Email*</label>
```

```
<input type="email" name="email" placeholder="Email" required>
```

```
</div>
```

```
<div>
```

```
</br>
```

```
<label>Korisničko ime*</label>
```

```
<input type="text" name="korisnicko_ime" placeholder="Korisničko ime"  
required>
```

```
</div>
```

```
<div>
```

```
</br>
```



```

    <label>Lozinka*</label>
    <input type="password" name="lozinka1" placeholder="Lozinka"
required>
</div>
<div >
    <br>
    <label>Ponovite lozinku*</label>
    <input type="password" name="lozinka2" placeholder="Lozinka"
required>
</div>
<div>
    <br>
    <label>Adresa</label>
    <input type="text" name="adresa" placeholder="Adresa">
</div>
<div>
    <br>
    <label>JMBG</label>
    <input type="text" name="JMBG" placeholder="JMBG">
</div>
<div >
    <br>
    <label>Broj telefona</label>
    <input type="text" name="telefon" placeholder="Broj telefona">
</div>
<div>
    <br>
    <label>Slika*</label>

<!-- input element koji ispisuje klasičnu "browse" strukturu za unos fajla
-- obrada validnosti unosa slike se vrši na serverskoj strani -->
<input type="file" name="slika" placeholder="Molim Vas unesite sliku..." >
</div>
</div>
<small>Polja koja su označena sa * su obavezna</small>
<br><br>

<!-- polje za izvršenje forme kojim se poziva login.php -->
<button type="submit" >Registruj se</button>
</div>
</form>
</main>

```

<?php include "footer.php"; ?>

Početna	Anketa	Rezultati ankete	Prijavi se
---------	--------	------------------	------------

Ime*

Prezime*

Email*

Korisničko ime*

Lozinka*

Ponovite lozinku*

Adresa

JMBG

Broj telefona

Slika Pata.jpg

Polja koja su oznacena sa * su obavezna

[Nazad na vrh](#)

© 2021 [Kontakt](#)

Slika 26. Forma za registraciju novog korisnika (*registracija.php*)

login.php

```
<!-- login.php ista kao iz primera 8. samo što se na početku skripte
-- uključuje skripta registracijaval1.php kojom se vrši validacija
-- registracije prikazan je početak skripte ostalo je isto
-->
<?php include "navigacija.php" ?>
<?php include "registracijaval1.php" ?>
<main>
<!-- ... -->
```

registracijaval1.php

```
<?php
$greska = "<p>Još niste registrovani</p>";

// Preuzimaju se podaci preneti POST metodom
if (isset($_POST["ime"]) && isset($_POST["prezime"]) &&
isset($_POST["email"]) && isset($_POST["korisnicko_ime"]) &&
isset($_POST["lozinka1"]) && isset($_POST["lozinka2"])) {
    $greska="";
    $ime = $_POST["ime"];
    $prezime = $_POST["prezime"];
    $email = $_POST["email"];
    $korisnicko_ime = $_POST["korisnicko_ime"];
    $lozinka1 = $_POST["lozinka1"];
    $lozinka2 = $_POST["lozinka2"];

    // Provera da li se lozinke podudaraju
    if(strcmp($lozinka1, $lozinka2)!= 0)
        $greska .= " Lozinke se ne poklapaju. ";
    }
    $adresa = "";
    if (isset($_POST["adresa"]))
        $adresa = $_POST["adresa"];
    $jmbg = "";
```

```
if (isset($_POST["JMBG"]))
    $jmbg = $_POST["JMBG"];
$telefon = "";
if (isset($_POST["telefon"]))
    $telefon = $_POST["telefon"];

/* $_FILES-asocijativni niz stavki otpremljenih u trenutnu skriptu preko
* HTTP POST metode.
* prva dimenzija naziv odgovarajućeg input elementa za upload fajla
* druga dimenzija predstavlja naziv određenog atributa fajla.
* $_FILES["slika"]["tmp_name"] - lokacija fajla na serveru
* $_FILES["slika"]["name"] - originalno ime fajla na
* računaru korisnika
* $_FILES["slika"]["size"] - veličina fajla u B
* $_FILES["slika"]["type"] - tip datoteke (ako postoji)
* $_FILES["slika"]["error"] - podaci o grešci
*/
if($_FILES["slika"]["error"]== UPLOAD_ERR_OK){
    $thefile=$_FILES["slika"]["tmp_name"];
    $thefile_name=$_FILES["slika"]["name"];
    $thefile_size=$_FILES["slika"]["size"];
    $thefile_type=$_FILES["slika"]["type"];
}
else
    $greska .= "Došlo je do greške pri uploadu slike: ".
$_FILES["slika"]["error"];

/* Provera da li je korisnik izabrao fajl*/
if ( !empty( $thefile_name ) )
{
    /* Provera da li je učitana slika odgovarajućeg formata*/
    if ( ( $thefile_type == "image/gif" ) ||
        ( $thefile_type == "image/jpeg" ) ||
        ( $thefile_type == "image/jpeg" ) )
    {

        /* Provera da li je učitana slika odgovarajuće veličine*/
        if ( $thefile_size < ( 1024 * 100 ) )
        {
            move_uploaded_file($thefile, "slike/$thefile_name");
        }
        else
```

```
        {
            $greska .= "Fajl je prevelik";
        }
    }
    else
    {
        $greska .= "Fajl nije u odgovarajućem formatu";
    }
}
else
{
    $greska .= "Niste izabrali sliku";
}

/* Ukoliko je otkrivena greška štampa se poruka o njoj*/
if ( $greska != "" )
{
    print( "<b>Došlo je do greške</b>: $greska<br>" );
}
else
{
    /* Rezultat pozivanja fajla upisi.php biće prikazan u sledećem primeru
    */
    //include "upisi.php";
    print( "Registrovani ste kao: $ime $prezime" );
    print( "<img src=\"slike/$thefile_name\" border=\"0\">" );
}
}
?>
```

Početna Anketa Rezultati ankete **Prijavi se**

Došlo je do greške: Fajl nije u odgovarajućem formatu

Prijava

Korisničko ime

Lozinka

Trenutno nema poslatih zahteva za logovanje

[Zaboravio sam lozinku](#)


Nemate nalog? [Registrujte se](#)

[Nazad na vrh](#)

© 2021 [Kontakt](#)

Slika 27. Neuspešno slanje pdf dokumenta umesto slike (ispisana greška pomoću *registracijaval1.php*)

Početna Anketa Rezultati ankete **Prijavi se**



Registrovani ste kao: Pata Patak

Prijava

Korisničko ime

Lozinka

Trenutno nema poslatih zahteva za logovanje

[Zaboravio sam lozinku](#)

Nemate nalog? [Registrujte se](#)

Slika 28. Uspešna registracija (*login.php*)*Primer 15. Rad sa datotekama*

Kada se u *registracijavall.php* uključi *upis.php* skripta podaci o korisniku će biti upisani u *registracija.txt* (slika 29).

upis.php

<?php

```
// Roditeljski direktorijum osnovnog direktorijuma za dokumente na serveru
$DOCUMENT_ROOT = $_SERVER['DOCUMENT_ROOT'];
echo '<p style="color:red" >Vaši podaci su upisani na serveru na
lokaciji: '.$DOCUMENT_ROOT.'</p>';
```

```
// Lokacija datoteke
```

```
$lokacija_fajla=$DOCUMENT_ROOT."/registracija.txt";
```

```
/* Otvara datoteku za dodavanje i čitanje od kraja postojećeg sadržaja ako
* ga ima; ako datoteka ne postoji, sistem pokušava da je napravi */
```

```
$fregistracija = fopen($lokacija_fajla,'a+');
fwrite ($fregistracija,$ime.", ".$prezime.", ".$email.", ".$korisnicko_ime.",
        ".$lozinka1.", ".$adresa."\n");
```

```
// Učitava datoteku u niz i svaki red datoteke smešta u zaseban element niza
```

```
$fniz=file($lokacija_fajla);
```

```
// Niz imena ključnih polja
```

```
$polja=array("ime", "prezime", "email", "korisnicko_ime", "lozinka1", "adresa
");
```

```
/* Koristeći explode funkciju string separator "," podeljen je string
```

```
* u niz stringova koji predstavlja vrednosti
```

```
* nizu vrednosti dodeljeni su string indeksi iz niza $polja
```

```
* štampa se dobijeni niz. */
```

```
foreach( $fniz as $fred_str) {
    print_r(array_combine($polja,explode(",",$fred_str)));
    echo '</br>';
}
```

```
// Funkcija fclose- zatvara datoteku
```

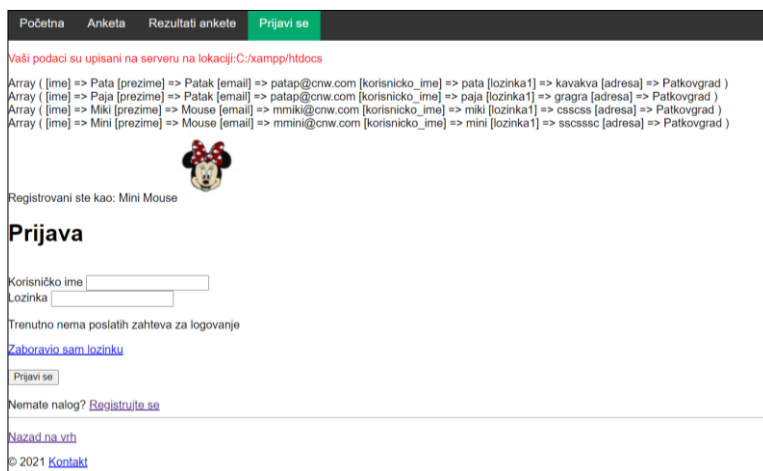
```
fclose($fregistracija);
```

```
?>
```

Sadržaj datoteke registracija.txt

Pata,Patak,patap@cnw.com,pata,kavakva,Patkovgrad
Paja,Patak,patap@cnw.com,paja,gragra,Patkovgrad
Miki,Mouse,mmiki@cnw.com,miki,csscss,Patkovgrad

Mini,Mouse,mmini@cnw.com,mini,sscssc,Patkovgrad



Slika 29. *login.php* pozvan nakon unetih podataka o novom korisniku (Mini Mouse)

Primer 16. Rad sa kolačićima

Kolačić je mali fajl koji server skladišti na računaru korisnika. PHP kolačići (eng.cookie) se uglavnom koriste za identifikaciju korisnika. Superglobalna promenljiva `$_COOKIE` sadrži sve vrednosti kolačića prosleđenih kao deo zahteva, gde su ključevi niza imena kolačića. U ovom primeru biće prikaza upotreba kolačića za čuvanje podataka o logovanju. Izmenjena je skripta *login.php* i rezultati poziva skripte kao i stanja kolačića u pregledaču su prikazani na slikama: slika 30, slika 31, slika 32. i slika 33.

login.php

```
<?php include "navigacija.php" ?>
<?php include "registracijavall.php" ?>
<?php
$poslato = false;
```

```
//Provera da li su podaci o korisniku prosleđeni POST metodom
if(isset($_POST['korisnickoIme']) && isset($_POST['lozinka']))
{
    $poslato = true;
    $korisnik=$_POST['korisnickoIme'];
```

```
/* Provera da li je zahtevano čuvanje podataka
* ukoliko se korisnik izjasnio da želi da čuva podatke
* funkcijom setcookie šalju se kolačići pregledaču
* putem HTTP odgovora */
if(!empty($_POST["remember"])) {

    if($_POST["remember"]=="DA"){

/* Parametri funkcije setcookie
* ime kolačića indeks u nizu $_COOKIE
* vrednosti
* dužina trajanja izražena u sekundama
**/
        setcookie ("username",$_POST["korisnickoIme"],time()+ 3600);
        setcookie ("password",$_POST["lozinka"],time()+ 3600);
        echo "Kolačići su uspešno postavljeni";
    }
    else {

/* Za brisanje kolačića postavlja se
* rok trajanja kolačića na neki prošli trenutak
* pa kako je isteklo vreme trajanja kolačić se briše
**/
        setcookie("username","", time() - 3600);
        setcookie("password","", time() - 3600);
        echo "Kolačići su uspešno izbrisani";
    }
}
}
?>
<main>
<div>
<h1>Prijava</h1><br>
<?php

/* Proverava se trenutno stanje sa kolčićima */
if (isset($_COOKIE["username"]) && isset($_COOKIE["password"]))
    echo "<p >Kolačići su postavljeni </p>";
else
    echo "<p >Kolačići nisu postavljeni </p>";
?>
</form>
```

<!--Forma za logovanje koja se prosleđuje POST metodom -->

```
<form action="login.php" method="POST" onsubmit="return provera()">
<div >
```

<!-- Ukoliko su kolačići postavljeni vrednost polja:

-- korisničkoIme dobija vrednost kolačića \$_COOKIE["username"]

-- lozinka dobija vrednost kolačića \$_COOKIE["password"]

-->

```
<label>Korisničko ime</label>
```

```
<input type="text" name="korisnickoIme" value="<?php
if(isset($_COOKIE["username"])) { echo $_COOKIE["username"];
}?>" id="username" autofocus>
```

```
</div>
```

```
<div >
```

```
<label>Lozinka</label>
```

```
<input type="password" name="lozinka" value="<?php
if(isset($_COOKIE["password"])) { echo $_COOKIE["password"]; }
?>" id="password" >
```

```
</div>
```

```
<div>
```

<!-- Promenljiva remember označava da li korisnik

-- želi da čuva podatke.

-->

```
<label> Da li želite da čuvate podatke o logovanju </label>
```

```
<select name="remember" id="zapamti">
```

```
<option value="NE"> NE </option>
```

```
<option selected value="DA"> DA </option>
```

```
</select>
```

```
</div>
```

```
<?php
```

```
if ($poslato)
```

```
    echo "<p > Trenutno je poslat zahtev za logovanje
korisnika<b>$korisnik</b>
```

```
</p>";
```

```
else
```

```
    echo "<p >Trenutno nema poslatih zahteva za logovanje </p>";
```

```
?>
```

<!--Ukoliko je korisnik zaboravio lozinku poziva se skripta

-- za resetovanje koja će biti predstavljena u primeru 20.

-->

`Zaboravio sam lozinku`

`

`

`<button type="submit" >Prijavi se</button>`

`</form>`

<!-- Za validaciju polja koristi se JS

-- funkcija proverava da li su polja username i password popunjena

-- i daje obaveštenje o nepopunjenom polju

-->

`<script>`

`function provera()`

`{`

`var username = document.getElementById('username').value.trim();`

`var password = document.getElementById('password').value.trim();`

`if(username.length == 0)`

`{`

`alert("Nije uneto korisničko ime!");`

`return false;`

`}`

`if(password.length == 0)`

`{`

`alert("Nije uneta lozinka!");`

`return false;`

`}`

`return true;`

`}`

`</script>`

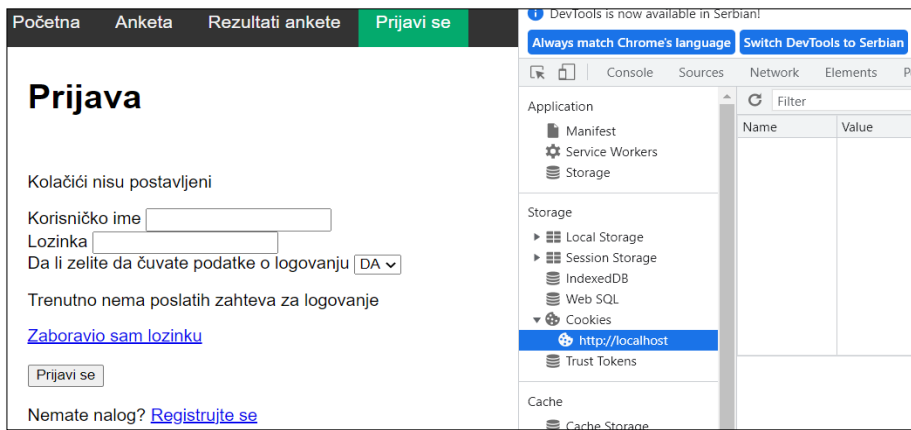
`
Nemate nalog? Registrujte se
`

`</div>`

`</main>`

<!--uključivanje skripte footer.php iz primera 7. -->

`<?php include "footer.php" ?>`

Slika 30. Forma za logovanje pre čuvanja kolačića (*login.php*)

HTTP zahtev za prvi poziv *login.php* skripte

GET /primeriphp/PHP/primer%2016/login.php HTTP/1.1

Host: localhost

Connection: keep-alive

sec-ch-ua: "Not;A Brand";v="99", "Google Chrome";v="97", "Chromium";v="97"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Windows"

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)

AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71

Safari/537.36

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9

Sec-Fetch-Site: same-origin

Sec-Fetch-Mode: navigate

Sec-Fetch-User: ?1

Sec-Fetch-Dest: document

Referer: http://localhost/primeriphp/PHP/primer%2016/login.php

Accept-Encoding: gzip, deflate, br

Accept-Language: sr-RS,sr;q=0.9,en-US;q=0.8,en;q=0.7

HTTP odgovor

HTTP/1.1 200 OK

Date: Fri, 21 Jan 2022 14:41:09 GMT

Server: Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.4.11

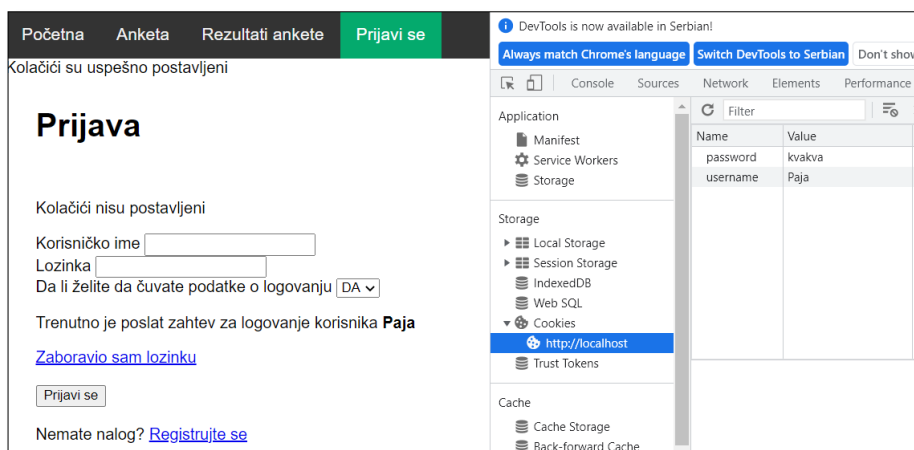
X-Powered-By: PHP/7.4.11

Content-Length: 3359

Keep-Alive: timeout=5, max=99

Connection: Keep-Alive

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



Slika 31. Forma za logovanje posle ubacivanja podataka sa opcijom za čuvanje podataka i izvršavanjem POST metodom

HTTP zahtev za prvi poziv *login.php* skripte

POST /primeriphp/PHP/primer%2016/login.php HTTP/1.1

Host: localhost

Connection: keep-alive

Content-Length: 45

Cache-Control: max-age=0

sec-ch-ua: "Not;A Brand";v="99", "Google Chrome";v="97", "Chromium";v="97"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Windows"

Upgrade-Insecure-Requests: 1

Origin: http://localhost

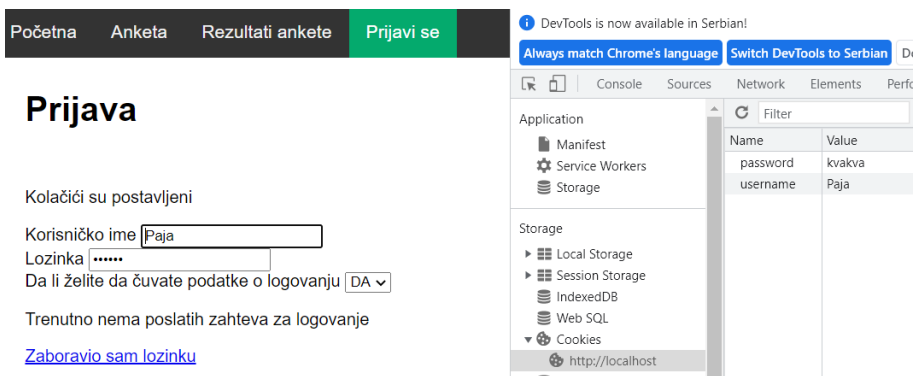
Content-Type: application/x-www-form-urlencoded

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/primeriphp/PHP/primer%2016/login.php
Accept-Encoding: gzip, deflate, br
Accept-Language: sr-RS,sr;q=0.9,en-US;q=0.8,en;q=0.7

HTTP odgovor

HTTP/1.1 200 OK
Date: Fri, 21 Jan 2022 17:09:34 GMT
Server: Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.4.11
X-Powered-By: PHP/7.4.11
Set-Cookie: username=Paja; expires=Fri, 21-Jan-2022 18:09:34 GMT; Max-Age=3600
Set-Cookie: password=kvakva; expires=Fri, 21-Jan-2022 18:09:34 GMT; Max-Age=3600
Content-Length: 3413
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

Prvi poziv *login.php* skripte je izvršen GET HTTP metodom. Tada nije postojao nikakav stari kolačić u pregledaču niti je putem HTTP odgovora postavljen. Nakon prvog unosa korisničkog imena i lozinke i izborom opcije za čuvanje podataka upućuje se HTTP zahtev za skiptom *login.php* POST metodom i na serverskoj strani se ubacuju kolačići i šalju kao odgovor pregledaču. Oni nisu još uvek upisani u niz `$_COOKIE` na serverskoj strani ali svaki naredni zahtev ka serveru će ići sa kolačićima. Tako da ako ponovo u meniju bude izabrano “Prijavi se” dobiće se ispisano korisničko ime i lozinka koji su sačuvani kao kolačići i prosleđeni serveru.



Slika 32. Forma za logovanje sa kolačićima

HTTP zahtev sa kolačićima

```
GET /primeriphp/PHP/primer%2015.1/login.php HTTP/1.1
Host: localhost
Connection: keep-alive
sec-ch-ua: " Not;A Brand";v="99", "Google Chrome";v="97",
"Chromium";v="97"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost/primeriphp/PHP/primer%2015.1/login.php
Accept-Encoding: gzip, deflate, br
Accept-Language: sr-RS,sr;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: username=Paja; password=kvakva
```


Ukoliko se izabere opcija da se informacije ne čuvaju zahtev koji ide ka serveru ide sa kolačićima ali odgovor koji prima klijent postavlja kolačiće sa isteklim vremenom i time ih briše.

HTTP odgovor sa naredbom za brisanje kolačića

HTTP/1.1 200 OK

Date: Fri, 21 Jan 2022 20:18:37 GMT

Server: Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.4.11

X-Powered-By: PHP/7.4.11

Set-Cookie: username=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0

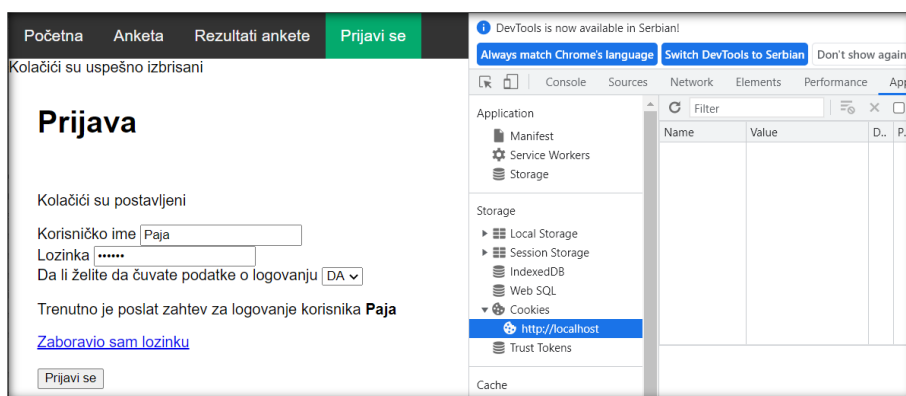
Set-Cookie: password=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0

Content-Length: 3419

Keep-Alive: timeout=5, max=100

Connection: Keep-Alive

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



Slika 33. Forma za logovanje nakon brisanja kolačića

Primer 17. Rad sa sesijama

PHP koristi sesije da upravlja kolačićima i kroz njih obezbeđuje trajne varijable koje su dostupne sa različitih veb stranica. U ovom primeru se koriste sesije da bi se omogućilo kreiranje forme sa više stranica i prilagođenom navigacijom, kod koje se čuvaju informacije o prijavljenom korisniku. Nakon

prijavljivanja korisnika (slika 34) gornja navigacija aplikacije se prilagođava novom korisniku i umesto prijave korisnik ima opciju da se odjavi (slika 35). Prilikom odjavljivanja poziva se skripta *logout.php* koja zatvara sesiju. Skripte koje su pozivale *navigacija.php* (*index.php*, *anketa.php*,...) skriptu će proverati da li ima ulogovanog korisnika i ako ima umesto nje poziva se *navigacija1.php*.

login.php

```
<?php include "navigacija.php" ?>
<?php include "registracijaval1.php" ?>
<?php
// stanje da li su podaci prosledeni
$poslato = false;

//Provera da li su podaci o korisniku prosledeni POST metodom
if(isset($_POST['korisnickoIme']) && isset($_POST['lozinka']))
{
    $poslato = true;
    $korisnik=$_POST['korisnickoIme'];
    /* Postavljanje kolačića koji čuvaju podatke o korisniku
    * odnosno brisanje informacija i njima */

    if(!empty($_POST["remember"])) {
        if($_POST["remember"]=="DA"){
            setcookie ("username",$_POST["korisnickoIme"],time()+ 3600);
            setcookie ("password",$_POST["lozinka"],time()+ 3600);
            echo " Kolačići su uspešno postavljeni";
        }
        else {
            setcookie("username","", time() - 3600);
            setcookie("password","", time() - 3600);
            echo " Kolačići su uspešno izbrisani";
        }
    }
}

/* Svaka skripta koja radi sa sesijama mora da pozove funkciju
* session_start()
* $_SESSION sadrži promenljive koje se čuvaju tokom jedne sesije dok
* je sesija aktivna.
* Otvorena je sesija i postavljena vrednost promenljive
* $_SESSION["korisnik"] na trenutnog aktivnog korisnika.
```

```
*/  
if (!isset($_SESSION)){  
    session_start();  
    $_SESSION["korisnik"] = $korisnik;  
  
    //Otvaranje skripte index.php sa prilagođenom navigacijom  
    header("Location:index.php");  
}  
}  
?>  
  
<!-- main deo login.php isti kao u primeru 16. -->  
<!--... -->  
<?php include "footer.php" ?>
```

index.php

```
<!-- index.php sa izmenjenim pozivom navigacije -->  
<!-- Uključuje se sadržaj iz programa navigacija.php ukoliko nema  
-- ulogovanih korisnika, ukoliko je u toku neka sesija odnosno  
-- postoji ulogovani korisnik uključuje se navigacija1.php -->  
<?php  
    session_start();  
    if (isset($_SESSION["korisnik"])){  
        include "navigacija1.php";  
        echo "Otvorena sesija korisnika ". $_SESSION["korisnik"];  
    }  
    else  
        include "navigacija.php";  
?>  
  
<!-- main deo index.php isti kao u primeru 7. -->  
<!--... -->  
<?php include "footer.php" ?>
```

logout.php

```
<?php
```

```
<!-- Funkcija session_destroy uništava aktivnu sesiju -->
```

```
    session_start();
```

```
    session_destroy();
```

```
    header("Location:index.php");
```

```
?>
```

navigacija1.php

```
<!-- navigacija1.php navigacija za ulogovanog korisnika -->
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
    <meta name="autor" content="Ulfeta Marovac">
```

```
    <meta name="description" content=" Navigacija za ulogovanog  
korisnika">
```

```
    <meta name="keywords" content="PHP, SESSION">
```

```
    <title>Primer 16: $_SESSION</title>
```

```
<style>
```

```
body {
```

```
    margin: 0;
```

```
    font-family: Arial, Helvetica, sans-serif;
```

```
}
```

```
main {
```

```
    margin: 30px;
```

```
}
```

```
.topnav {
```

```
    overflow: hidden;
```

```
    background-color: #333;
```

```
}
```

```
.topnav a {
```

```
    float: left;
```

```
    color: #f2f2f2;
```

```
    text-align: center;
```

```
    padding: 14px 16px;
```

```
    text-decoration: none;
```

```
    font-size: 17px;
```

```
}
```

```
.topnav a:hover {  
    background-color: #ddd;  
    color: black;  
}  
.topnav a.active {  
    background-color: #04AA6D;  
    color: white;  
}  
</style>  
</head>  
<body>  
<?php
```

```
// Preuzimanje informacija o fajlu iz koga je pokrenuta navigacija  
$putanja = $_SERVER["SCRIPT_NAME"];
```

```
// Poslednja pozicija znaka '/'  
$pozicija=strrpos($_SERVER["SCRIPT_NAME"], "/");  
$naziv_fajla=substr($_SERVER["SCRIPT_NAME"], $pozicija + 1);  
?>  
<div class="topnav">
```

```
<!-- Postavlja element navigacije stranice koja je trenutno učitana kao  
-- aktivna  
-- Ulogovani korisnik ima mogućnost da se izloguje (Odjavi se)  
-->
```

```
<a class=" <?php if ($naziv_fajla === "index.php") echo 'active'; ?>"  
href="index.php"> Početna</a>  
<a class=" <?php if ($naziv_fajla === "anketa.php") echo 'active'; ?>"  
href="anketa.php"> Anketa</a>  
<a class=" <?php if ($naziv_fajla === "rezultati.php") echo 'active'; ?>"  
href="rezultati.php">Rezultati ankete</a>  
<a class=" <?php if (($naziv_fajla === "logout.php")) echo 'active'; ?>"  
href="logout.php"> Odjavi se </a>  
</div>
```

Početna Anketa Rezultati ankete **Prijavi se**

Prijava

Kolačići su postavljeni

Korisničko ime

Lozinka

Da li želite da čuvate podatke o logovanju ☐

Trenutno nema poslatih zahteva za logovanje

[Zaboravio sam lozinku](#)

Nemate nalog? [Registrujte se](#)

[Nazad na vrh](#)

© 2021 [Kontakt](#)

Slika 34. Logovanje korisnika Paja - *login.php*



Slika 35. Poziv početne strane nakon logovanja korisnika *index.php*

Primer 18. Kreiranje baze podataka, korisnika i korisničkih privilegija

PHP ima podršku za rad sa preko 20 različitih baza podataka. Ovde će biti prikazan rad sa jednom od najpoznatijih open-source baza podataka zasnovana na SQL jeziku, MySQL bazom podataka. MySQL baza je jedna od najpoznatijih i najviše korišćenih open-source baza podataka zasnovana na SQL jeziku. Za pristup bazi podataka korišćen je phpMyAdmin klijent prilagođen PHP-u. Prikazano je definisanje i kreiranje baze podataka, korisnika i korisničkih privilegija (slika 36, slika 37).

Kreiranje baze anketa sa tabelama

-- Baza anketa za čuvanje podataka o popunjenoj anketi

CREATE DATABASE anketa;

-- Tabela Korisnik za čuvanje podataka o korisniku

CREATE TABLE anketa.Korisnik (
korisnicko_ime varchar(255) NOT NULL,
ime varchar(30) NOT NULL,
prezime varchar(30) NOT NULL,
email varchar(50) NOT NULL,
lozinka varchar(64) NOT NULL,
adresa varchar(50) DEFAULT NULL,
jmbg varchar(15) DEFAULT NULL,
telefon varchar(20) NOT NULL,
slika varchar(50) DEFAULT NULL,
tip_korisnika enum('korisnik','admin') NOT NULL,
stanje varchar(11) DEFAULT NULL
PRIMARY KEY (korisnicko_ime,email)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4

Kreiranje korisnika baze

-- Kreiranje korisnika koji mogu da se loguju na server sa bilo kog hosta

CREATE USER 'korisnik'@'%' IDENTIFIED BY "lozinka";

-- Dodeljivanje privilegije nad svim podacima korisniku

GRANT ALL PRIVILEGES ON *.* TO 'korisnik'@'%'

Host	User	Password	Select_priv	Insert_priv	Update_priv	Delete_priv	Create_priv	Drop_priv	Reload_priv
localhost	root		Y	Y	Y	Y	Y	Y	Y
%	korisnik	*90C1006F6FDD5738C455CF359F9EE76CFAB98265	N	N	N	N	N	N	N

Slika 36. Tabela mysql.user nakon kreiranja korisnika “korisnik”

Host	User	Password	Select_priv	Insert_priv	Update_priv	Delete_priv	Create_priv	Drop_priv	Reload_priv
localhost	root		Y	Y	Y	Y	Y	Y	Y
%	korisnik	*90C1006F6FDD5738C455CF359F9EE76CFAB98265	Y	Y	Y	Y	Y	Y	Y

Slika 37. Tabela mysql.user posle dodele privilegija korisniku „korisnik“

Primer 19. Povezivanje i rad sa bazom podataka

PHP ima više interfejsa za rad sa MySQL bazom podataka. U ovom primeru će biti prikazan objektno-orjentisani pristup upotrebom klase MySQLi (MySQL Improved extension). Ovaj interfejs je ugrađen u PHP pa za njegovu upotrebu je potrebno instanciranje objekta klase mysqli. Sledeće PHP skripte prikazuju povezivanje sa bazom podataka, unos, čitanje, ispravljanje i brisanje podatka iz tabele Korisnik. Prikazano je stanje baze nakon punjenja podataka (slika 38) kao i koji ulogovani korisnik može da pristupi, promeni i izbriše svoje podatke (slika 39, slika 40, slika 41. i slika 42).

baza/db.php

<?php

```
/* Funkcija veza pravi objekat $konekcija klase mysqli
* parametri konstruktora su:
* ime računara na kome je baza
* ime korisnika baze
* lozinka korisnika baze
* naziv baze podataka kojoj želite da pristupite
*/
function veza()
{
    $konekcija = new mysqli('localhost', 'korisnik', 'lozinka','anketa');
    if($konekcija->connect_error)
        die('Greška prilikom konekcije');
    return $konekcija;
}

/* Funkcija za upisivanje novog korisnika
* lozinka se upisuje kriptovana u bazu pomoću md5 algoritma
* metoda query(...) izvršava upit nad objektom klase msqli
* rezultat metode query je tipa : mysqli_result
* klasa mysqli_result ima svojstva
* $current_field;
* $field_count;
* array $lengths;
* $num_rows;
```

```

*/
function upis_korisnika($konekcija, $korisnicko_ime, $ime, $prezime,
$email, $lozinka, $adresa, $jmbg, $telefon, $thefile_name)
{
    $kriptovanalozinka = md5($lozinka);
    $sql="INSERT INTO Korisnik (korisnicko_ime, ime, prezime,
        email, lozinka, adresa, jmbg, telefon,slika, tip_korisnika)
        VALUES
        ('$korisnicko_ime','$ime','$prezime','$email','$kriptovanalozinka',
        '$adresa','$jmbg','$telefon','$slike/$thefile_name','korisnik') ";
    $konekcija->query($sql);
}

```

```

/* Funkcija broj_korisnika()
* izvršava upit kojim se proverava da li u tabeli korisnik postoji
* korisnik sa datim korisničkim imenom i lozinkom
* vraća broj pronađenih redova
*/
function broj_korisnika($konekcija,$korisnik,$lozinka)
{
    $kriptovanalozinka = md5($lozinka);
    $sql=" SELECT * FROM Korisnik
        WHERE
        korisnicko_ime='$korisnik'
        AND lozinka='$kriptovanalozinka' ";
    $rezultat=$konekcija->query($sql);
    return $rezultat->num_rows;
}

```

```

/* Funkcija broj_korisnika_email()
* izvršava upit kojim se proverava da li u tabeli korisnik postoji
* korisnik sa datim korisničkim imenom ili email-om
* vraća broj pronađenih redova
*/
function broj_korisnik_email($konekcija,$korisnik,$email)
{
    $sql=" SELECT * FROM Korisnik
        WHERE
        korisnicko_ime='$korisnik'
        or email='$email' ";
    $rezultat=$konekcija->query($sql);
}

```

```

    return $rezultat->num_rows;
}

/* Funkcija podaci_korisnika() vraća podatke
 * korisnika na osnovu unetog korisničkog imena
 */
function podaci_korisnika($konekcija,$korisnik)
{
    $sql=" SELECT ime, prezime, email, adresa, jmbg, telefon, slika,
            tip_korisnika
        FROM Korisnik
        WHERE
            korisnicko_ime='$korisnik'";
    $rezultat=$konekcija->query($sql);
    return $rezultat;
}

/* Funkcija promeni_podatke_korisnika()
 * ažurira podatke korisnika sa odgovarajućim
 * korisničkim imenom
 */
function promeni_podatke_korisnika($konekcija, $korisnicko_ime, $ime,
    $prezime,$adresa,$jmbg,$telefon,$thefile_name)
{
    $sql="UPDATE Korisnik SET
            ime='$ime', prezime='$prezime', adresa='$adresa',
            jmbg='$jmbg', telefon='$telefon', slika='$thefile_name',
            tip_korisnika='korisnik'
        WHERE
            korisnicko_ime='$korisnicko_ime'";
    return ($konekcija->query($sql));
}

/* Funkcija promeni_lozinku ()
 * ažurira podatke korisnika sa odgovarajućim
 * korisničkim imenom
 */
function      promeni_lozinku($konekcija,      $korisnicko_ime,
    $lozinka_stara, $lozinka)
{
    $kriptovanalozinka = md5($lozinka_stara);
    $kriptovanalozinka1 = md5($lozinka);
    $sql="UPDATE korisnik SET

```

```
        lozinka='$kriptovanalozinka1'
    WHERE
        korisnicko_ime='$korisnicko_ime'
    AND lozinka='$kriptovanalozinka' ";
    $rezultat=$konekcija->query($sql);
    return $rezultat;
}

/* Funkcija brisanje_korisnika vrši
 * brisanje korisnika sa datim korisničkim imenom
 */
function brisanje_korisnika($konekcija,$korisnicko_ime)
{
    $sql=" DELETE FROM Korisnik
    WHERE
        korisnicko_ime='$korisnicko_ime'";
    $rezultat=$konekcija->query($sql);
    return $rezultat;
}

/* Funkcija raskid uništava prosleđeni objekat
 * close()- metoda klase mysqli će počistiti
 * objekat klase mysqli iz memorije
 */
function raskid($konekcija)
{
    $konekcija->close();
}

?>
```

login.php

```
<?php include "navigacija.php" ?>
<?php include "registracijavall.php" ?>
<?php include_once "baza/db.php"; ?>
<?php
// Stanje da li su podaci prosleđeni
$poslato = false;
```

```
/* Provera da li su podaci o korisniku prosleđeni POST metodom
*/
if(isset($_POST['korisnickoIme'])&& isset($_POST['lozinka']))
{
    $poslato = true;
    $korisnik=$_POST['korisnickoIme'];
    $lozinka=$_POST['lozinka'];

/* Poziv funkcija skripte db.php
* $broj predstavlja broj korisnika u tabeli Korisnik sa
* odgovarajućim korisničkim imenom i lozinkom
*/
    $anketa=veza();
    $broj=broj_korisnika($anketa,$korisnik,$lozinka);
    raskid_veze($anketa);

/* Ako je broj korisnika u tabeli Korisnik
* različit od 1 onda je greška
*/
    if($broj <> 1){
        $poslato = false;
        echo "Neispravno korisničko ime ili lozinka!!!";
    }
    else
        /* Ostatak programa (sivo označen) je isti kao u prethodnom
        * primeru */
        {

            /* Provera da li je zahtevano čuvanje podataka
            * ukoliko se korisnik izjasnio da želi da čuva podatke
            * funkcijom setcookie šalju se kolačići pregledaču
            * putem HTTP odgovora */
            if(!empty($_POST["remember"])) {

                if($_POST["remember"]=="DA"){

/* parametri funkcije setcookie
* ime kolačića indeks u nizu $_COOKIE
* vrednosti
* dužina trajanja izražena u sekundama
**/
                setcookie ("username",$_POST["korisnickoIme"],time()+ 3600);
```

```
setcookie ("password",$_POST["lozinka"],time()+ 3600);
echo " Kolačići su uspešno postavljeni";
}
else
{

/* Za brisanje kolačića postavlja se
 * rok trajanja na prošli trenutak
 * pa kako je isteklo vreme trajanja kolačić se briše
 */
setcookie("username","", time() - 3600);
setcookie("password","", time() - 3600);
echo " Kolačići su uspešno izbrisani";
}
}

/* Svaka skripta koja radi sa sesijama mora da pozove funkciju
 * session_start()
 * $_SESSION sadrži promenljive koje se čuvaju tokom jedne sesije dok je
 * sesija aktivna.
 * Otvorena je sesija i postavljana vrednost promenljive
 * $_SESSION["korisnik"] na trenutno aktivnog korisnika.
 */
if (!isset($_SESSION)){
    session_start();
    $_SESSION["korisnik"] = $korisnik;

//Otvaranje skripte index.php koja će navigaciju prilagoditi
header("Location:index.php");
}
}
}
?>
<main>
<div>
<h1>Prijava</h1><br>
<?php

/* Proverava se trenutno stanje sa kolačićima */
if( isset($_COOKIE["username"]) && isset($_COOKIE["password"]))
    echo "<p>Kolačići su postavljeni </p>";
else
```

```

        echo "<p >Kolačići nisu postavljeni </p>";
    ?>
</form>

<!--Forma za logovanje koja se prosleđuje POST metodom -->
<form action="login.php" method="POST" onsubmit="return provera()">
    <div >

        <!-- Ukoliko su kolačići postavljeni vrednost polja:
            -- korisničkoIme dobija vrednost kolačića $_COOKIE["username"]
            -- lozinka dobija vrednost kolačića $_COOKIE["password"]
            -->
        <label>Korisničko ime</label>
        <input type="text" name="korisnickoIme" value="<?php
            if(isset($_COOKIE["username"])) { echo $_COOKIE["username"];
        }?>"
            id="username" autofocus>
        </div>
        <div >
            <label>Lozinka</label>
            <input type="password" name="lozinka" value="<?php
                if(isset($_COOKIE["password"])) { echo $_COOKIE["password"]; }
            ?>"
                id="password" >
            </div>
        <div>

            <!-- Promenljiva remember označava da li korisnik želi
                -- da čuva podatke.
                -->
            <label> Da li želite da čuvate podatke o logovanju </label>
            <select name="remember" id="zapamti">
                <option value="NE"> NE </option>
                <option selected value="DA"> DA </option>
            </select>
        </div>

        <?php
        if ($poslato)
            echo "<p > Trenutno je poslat zahtev za logovanje
korisnika<b>$korisnik</b>
            </p>";

```

```
else
    echo "<p>Trenutno nema poslatih zahteva za logovanje </p>";
?>

<!--Ukoliko je korisnik zaboravio lozinku poziva se skripta
-- za resetovanje koja će biti predstavljena u primeru 20.
-->
<a href=" zaboravljena_lozinka.php ">Zaboravio sam lozinku</a>
<br><br>
<button type="submit" >Prijavi se</button>
</form>

<!-- Za validaciju polja koristi se JS
-- funkcija proverava da li su polja username i password popunjena
-- i daje obaveštenje o nepopunjenom polju
-->
<script>
function provera()
{
    var username = document.getElementById('username').value.trim();
    var password = document.getElementById('password').value.trim();

    if(username.length == 0)
    {
        alert("Nije uneto korisničko ime!");
        return false;
    }

    if(password.length == 0)
    {
        alert("Nije uneta lozinka!");
        return false;
    }

    return true;
}
</script>
<br>Nemate nalog? <a href="registracija.php">Registrujte se</a><br>
</div>
</main>
<!--uključivanje skripte footer.php iz primera 7. -->
<?php include "footer.php" ?>
```

registracijaval1.php

*<!-- Prvi deo programa (sivi označeni deo) je isti kao u prethodnom primeru**-->**<?php**include_once "baza/db.php";**\$greska = "<p>Još niste registrovani</p>";**if (isset(\$_POST["ime"]) && isset(\$_POST["prezime"]) &&
isset(\$_POST["email"]) && isset(\$_POST["korisnicko_ime"]) &&
isset(\$_POST["lozinka1"]) && isset(\$_POST["lozinka2"])) {**\$greska="";**\$ime = \$_POST["ime"];**\$prezime = \$_POST["prezime"];**\$email = \$_POST["email"];**\$korisnicko_ime = \$_POST["korisnicko_ime"];**\$lozinka1 = \$_POST["lozinka1"];**\$lozinka2 = \$_POST["lozinka2"];**// Provera da li se lozinke podudaraju**if(strcmp(\$lozinka1, \$lozinka2)!= 0)**\$greska .= " Lozinke se ne poklapaju. ";**}**\$adresa = "";**if (isset(\$_POST["adresa"]))**\$adresa = \$_POST["adresa"];**\$jmbg = "";**if (isset(\$_POST["JMBG"]))**\$jmbg = \$_POST["JMBG"];**\$telefon = "";**if (isset(\$_POST["telefon"]))**\$telefon = \$_POST["telefon"];**/* \$_FILES-asocijativni niz stavki otpremljenih u trenutnu skriptu preko
HTTP*** POST metode.*** prva dimenzija naziv odgovarajućeg input elementa za upload fajla*** druga dimenzija predstavlja naziv određenog atributa fajla.*** \$_FILES["slika"]["tmp_name"] - lokacija fajla na serveru*** \$_FILES["slika"]["name"] - originalno ime fajla na računaru korisnika*** \$_FILES["slika"]["size"] - veličina fajla u B*

```
* $_FILES["slika"]["type"] - tip datoteke (ako postoji)
* $_FILES["slika"]["error"] - podaci o grešci
*/
if($_FILES["slika"]["error"]== UPLOAD_ERR_OK){
    $thefile=$_FILES["slika"]["tmp_name"];
    $thefile_name=$_FILES["slika"]["name"];
    $thefile_size=$_FILES["slika"]["size"];
    $thefile_type=$_FILES["slika"]["type"];
}
else
    $greska .= "Došlo je do greške pri uploadu slike: ".
$_FILES["slika"]["error"];

/* Provera da li je korisnik izabrao fajl*/
if( !empty( $thefile_name ) )
{
    /* Provera da li učitana slika odgovarajućeg formata*/
    if( ( $thefile_type == "image/gif" ) ||
        ( $thefile_type == "image/pjpeg" ) ||
        ( $thefile_type == "image/jpeg" ) )
    {
        /* Provera da li učitana slika odgovarajuće veličine*/
        if( $thefile_size < ( 1024 * 100 ) )
        {
            move_uploaded_file($thefile, "slike/$thefile_name");
        }
        else
        {
            $greska .= "Fajl je prevelik";
        }
    }
    else
    {
        $greska .= "Fajl nije u odgovarajućem formatu";
    }
}
else
{
    $greska .= "Niste izabrali sliku";
}
```

```

/* Ukoliko je otkrivena greška štampa se poruka o njoj*/
if ( $greska != "" )
{
    print( "<b>Došlo je do greške</b>: $greska<br>");
}
else
{

    /* Nakon validacije podataka vrši se upisivanje novog korisnika
    * pozivaju se funkcije iz skripte db.php za
    * povezivanje na bazu podataka
    * proveru da li postoji korisnik sa datim korisničkim imenom i mejlom
    * raskidanje veze sa bazom
    */
    $anketa=veza();
    if(broj_korisnik_email($anketa,$korisnicko_ime,$email)>0)
    {
        raskid_veze($anketa);

        /* Ukoliko se u bazi pronadu korisnici sa istim korisničkim imenom ili
        * email-om onda se korisniku šalje poruka u okvir za potvrđivanje
        * nakon potvrđivanja ponovo se otvara skripta za registraciju
        */
        echo '<script>
            if (confirm("Već postoji nalog sa ovim korisničkim imenom ili
mejlom!"))
            {
                location.href = "registracija.php";
            }
        </script>';
    }
    else
    {

        /* Ako ne postoji korisnik sa datim korisničkim imenom i mejlom
        * poziva se funkcija za unos podataka u bazu
        * ukoliko prođe bez problema korisnik dobija poruku da je
        * registrovan u slučaju problema upisa podataka u bazu korisnik
        * se obaveštava i vraća na registraciju.
        */
        if( upis_korisnika($anketa, $korisnicko_ime, $ime, $prezime, $email,
$lozinka1, $adresa,$jmbg,$telefon,$thefile_name))

```

```

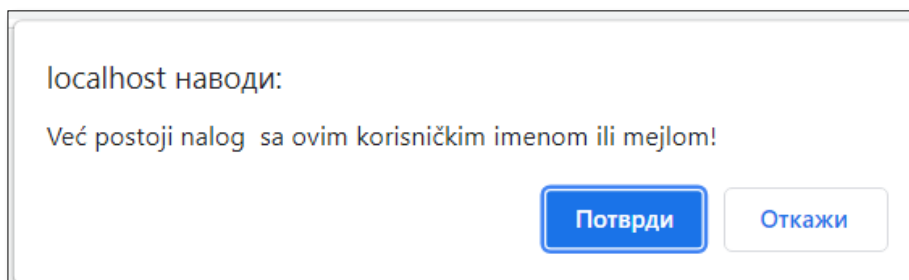
{
    raskid_veze($anketa)

    /* Kod uspešne registracije poziva se program za verifikaciju
    * mejla kome se GET metodom prosleđuje vrednost
    * e-maila i korisničkog imena.
    * Realizacija skripte potvrdi_mail.php je sledećem primeru.
    */
    echo "<script> if (confirm(\"Podaci su uspešno upisani!\")) {
location.href=\"potvrdi_mail.php?email=$email&korisnik=$korisnicko_ime\";
    }</script>";
}
else
{
    raskid_veze($anketa);
    echo '<script> if (confirm("INSERT nije uspeo!"))
        { location.href = "registracija.php"; }
        </script>';
}
}
}
?>

```

korisnicko_ime	ime	prezime	email	lozinka	adresa	j
mini	Mini	Mouse	mmini@cnw.com	3514ca7dbcce6c8feb7ce4b705fccdd	Patkovgrad	0
pata	Pata	Patak	patap@cnw.com	kvakva	Patkovgrad	0

Slika 38. Prikaz tabele Korisnik nakon ubacivanja korisnika Mini sa šifrovanom lozinkom (uključena funkcija md5) i bez šifrovane lozinke Pata

Slika 39. Neuspešna registracija (*registracijaval1.php*)*navigacija1.php*

```

<!-- U navigacija1.php dodata su još dva polja za izmenu podataka i
-- brisanje korisničkog naloga
-->
<a class=" <?php if (($naziv_fajla === "izmena_podataka.php")) echo
'active'; ?>" href="izmena_podataka.php"> Izmeni podatke </a>
<a class=" <?php if (($naziv_fajla === "login.php"))||($naziv_fajla ===
"brisanje?korisnika.php")) echo 'active'; ?>" href="brisanje.php"> Obriši
nalog </a>

```

izmena_podataka.php

```

<?php
    session_start();
    if (isset($_SESSION["korisnik"])){
        include "navigacija1.php";
        echo "Otvorena sesija korisnika ". $_SESSION["korisnik"];
    }
    else
    {

        //Ukoliko nije u toku sesija otvara se forma za logovanje
        header("Location:login.php");
    }
?>

```

```

<!-- Preuzimaju se podaci sa servera o ulogovanom korisniku
-->

```

```

<?php include_once "baza/db.php";
$korisnicko_ime=$_SESSION["korisnik"];
$anketa=veza();
$rezultat=podaci_korisnika($anketa,$korisnicko_ime);

/* Metoda fetch_assoc() obezbeđuje jedan po jedan red podataka,
 * koji se čuvaju u promenljivoj $korisnicki_podaci.
 * Rezultat ove metode je asocijativni niz čiji su indeksi
 * imena polja izdvojenih sql upitom.
 */
$korisnicki_podaci=$rezultat->fetch_assoc();
$ime= $korisnicki_podaci['ime'];
$email= $korisnicki_podaci['email'];
$prezime = $korisnicki_podaci['prezime'];
$adresa= $korisnicki_podaci['adresa'];
$jmbg= $korisnicki_podaci['jmbg'];
$telefon= $korisnicki_podaci['telefon'];
$thefile_name= $korisnicki_podaci['slika'];
$tip_korisnika= $korisnicki_podaci['tip_korisnika'];
raskid_veze($anketa);
?>
<main>

<!-- Forma u kojoj je uneto trenutno stanje podataka korisnika
-- ukoliko korisnik želi da promeni podatke može sve osim
-- korisničkog imena, mejla i tipa korisnika
-->
<form action="izmenaval.php" method="post" enctype="multipart/form-
data"> <div>
<div>
<label>Tip korisnika - <?php echo $tip_korisnika; ?></label>

<!-- Skriveno polje koje nosi informaciju o tipu korisnika -->
<input name="tip_korisnika" value="<?php echo $tip_korisnika; ?>"
type="hidden" > </div>
<div>
<br></br>
<label>Ime</label>
<input type="text" name="ime" value="<?php echo $ime; ?>" >
</div>
<div>

```

```
</br>
<label>Prezime</label>
<input type="text" name="prezime" value="<?php echo $prezime; ?>" >
</div>
</div>
```

*<!-- Korisnici se identifikuju pomoću korisničkog imena i mejla
-- pa je zabranjeno editovanje polja sa opcijom readonly*

-->

```
<div>
```

```
<div>
```

```
</br>
```

```
<label>Email*</label>
```

```
<input type="email" name="email" value="<?php echo $email; ?>"
readonly>
```

```
</div>
```

```
<div>
```

```
</br>
```

```
<label>Korisničko ime*</label>
```

```
<input type="text" name="korisnicko_ime" value="<?php echo
$korisnicko_ime; ?>" readonly>
```

```
</div>
```

```
</div>
```

*<!-- Ukoliko korisnik želi da promeni lozinku mora prvo uneti staru
-- a zatim potvrditi novu lozinku*

-->

```
<div>
```

```
</br>
```

```
<label>Za promenu lozinke morate uneti staru lozinku:</label>
```

```
</br>
```

```
<div>
```

```
</br>
```

```
<label>Stara lozinka</label>
```

```
<input type="password" name="lozinka_stara" placeholder="stara
lozinka">
```

```
</div>
```

```
<div >
```

```
</br>
```

```
<label>Nova lozinka</label>
```

```
<input type="password" name="lozinka1" placeholder="nova lozinka">
```

```
</div>
```

```

<div>
  </br>
  <label>Ponovite novu lozinku</label>
  <input type="password" name="lozinka2" placeholder="nova lozinka">
</div>
</div>
<div>
<div >
  </br>
  <label>Adresa</label>
  <input type="text" name="adresa" value="<?php echo $adresa; ?>" >
</div>
<div>
  </br>
  <label>JMBG</label>
  <input type="text" name="JMBG" value="<?php echo $jmbg; ?>">
</div>
<div >
  </br>
  <label>Broj telefona</label>
  <input type="text" name="telefon" value="<?php echo $telefon; ?>">
</div>
<div>
  <!--Tabela u kojoj je prikazana slika i data mogućnost izbora nove slike-->
  <table>
    <td>

      <!-- Skriveno polje koje nosi informaciju o lokaciji slike -->
      <input name="thefile_name" value="<?php echo $thefile_name; ?>"
      type="hidden">

      <!--Prikaz trenutne slike korisnika -->
      
      </td>
      <td>
      </br>
      <label>Odaberite novu sliku</label>
      <input type="file" name="slika" placeholder="Ako zelite Vas unesite novu
      sliku..." >
      </td>
    </table>
  </div>

```



```

<small>Polja koja su označena sa * se ne mogu promeniti</small>
<br><br>
<div >

  <!--Ukoliko korisnik želi da odustane vraća se na početnu stranu-->
  <a href="index.php"> Odustani od izmene i vrati se na početnu stranu </a>
  <br><br>
  <button type="submit" >Izmeni podatke</button>
</div>
</form>
</main>
<?php include "footer.php"; ?>

```

The screenshot shows a web application interface with a dark navigation bar at the top containing the following links: Početna, Anкета, Rezultati ankete, Odjavi se, **Izmeni podatke** (highlighted in green), and Obriši nalog. Below the navigation bar, a message states 'Otvorena sesija korisnika pata'. The main content area contains a form for editing user data. The form includes the following fields and labels:

- Tip korisnika - korisnik
- Ime: Pata
- Prezime: Patak
- Email*: patap@cnw.com
- Korisničko ime*: pata
- Za promenu lozinke morate uneti staru lozinku:
- Stara lozinka: stara lozinka
- Nova lozinka: nova lozinka
- Ponovite novu lozinku: nova lozinka
- Adresa: Patkovgrad
- JMBG: 0906937837838
- Broj telefona: +381 641111111
- A profile picture of a purple duck is shown next to the text 'Odaberite novu sliku'. There is a 'Choose File' button and the text 'No file chosen'.

At the bottom of the form, there is a note: 'Polja koja su označena sa * se ne mogu promeniti'. Below this note is a link: 'Odustani od izmene i vrati se na početnu stranu'. At the very bottom of the form is a button labeled 'Izmeni podatke'.

Slika 40. Izgled poziva skripte *izmena_podataka.php* kada je ulogovan korisnik pata

izmenaval.php

```
<?php
include_once "baza/db.php";

/* Preuzimaju se promenljive prosleđene POST metodom */
$ime = $_POST["ime"];
$prezime = $_POST["prezime"];
$email = $_POST["email"];
$korisnicko_ime = $_POST["korisnicko_ime"];
$adresa = $_POST["adresa"];
$jmbg = $_POST["JMBG"];
$telefon = $_POST["telefon"];
$greska = "";

/* Ukoliko je Korisnik uneo neke podatke u polja za lozinku pokreće se
 * promena lozinke. */
$promena_lozinke = 0;
if (isset($_POST["lozinka_stara"]) && isset($_POST["lozinka1"]) &&
    isset($_POST["lozinka2"]))
{
    $promena_lozinke = 1;
    $lozinka_stara = $_POST["lozinka_stara"];
    $lozinka1 = $_POST["lozinka1"];
    $lozinka2 = $_POST["lozinka2"];
    if(strcmp($lozinka1, $lozinka2)!= 0)
        $greska .= " Lozinke se ne poklapaju. ";
}

/* Provera da li je korisnik izabrao novu sliku
 * i da li je uspešno preneti */
if($_FILES["slika"]["error"]== UPLOAD_ERR_OK){
    $thefile=$_FILES["slika"]["tmp_name"];
```

```
$thefile_name=$_FILES["slika"]["name"];
$thefile_size=$_FILES["slika"]["size"];
$thefile_type=$_FILES["slika"]["type"];
if ( !empty( $thefile_name ) )

{

    /* Provera da li je učitana slika odgovarajućeg formata
    */
    if ( ( $thefile_type == "image/gif" ) ||
        ( $thefile_type == "image/pjpeg" ) ||
        ( $thefile_type == "image/jpeg" ) )
    {

        /* Provera da li je učitana slika odgovarajuće veličine
        */
        if ( $thefile_size < ( 1024 * 100 ) )
        {
            move_uploaded_file($thefile, "slike/$thefile_name");
            $thefile_name="slike/".$thefile_name;
        }
        else
        {
            $greska .= " Fajl je prevelik";
        }
    }
    else
    {
        $greska .= " Fajl nije u odgovarajućem formatu";
    }
}
else
{
    /* Ukoliko nije izabrana slika postavlja se stara lokacija
    */
    $thefile_name= $_POST["thefile_name"];
}
}
else
    $thefile_name= $_POST["thefile_name"];

/* Ukoliko je otkrivena greška štampa se poruka o njoj
```

```
*/
if ( $greska != "" )
{
    print( "<b>Došlo je do greške</b>: $greska<br>" );
}
else
{

    /* Ažuriranje podataka u tabeli Korisnik
    */
    $anketa=veza();
    if(promeni_podatke_korisnika($anketa,$korisnicko_ime,$ime,$prezime,
    $adresa,$jmbg,$telefon,$thefile_name))
        echo 'Uspešno su promenjeni podaci o korisniku!</br>';
    else
        echo 'Ažuriranje podataka nije uspelo!</br>';

    /* Promena lozinke
    */
    if($promena_lozinke===1 && $lozinka1 != ""){

        /* Provera validnosti stare lozinke
        * ukoliko je stara lozinka validna menja se novom lozinkom
        * inače se šalje poruka o neuspeloj promeni
        */
        if ( broj_korisnika($anketa,$korisnicko_ime,$lozinka_stara)===1 )
        {
            if(promeni_lozinku($anketa,$korisnicko_ime,$lozinka_stara,$lozinka1))

                echo 'Šifra je uspešno promenjena!<br>';
            else
                echo 'Ažuriranje šifre nije uspelo!</br>';
        }
        else
            echo 'Promena šifre nije uspešna, nevalidna stara lozinka!<br>';
        }
        else
            echo 'Šifra nije promenjena!<br>';
        raskid_veze($anketa);

        echo '<a href="index.php"> <br>Vrati se na početnu stranu </a>';
    }
}
```

?>

Uspešno su promenjeni podaci o korisniku!
Šifra je uspešno promenjena!

[Vrati se na početnu stranu](#)

Slika 41. Poruka skripte *izmenaval.php* nakon uspešno izmenjenih podataka i korisničke šifre

brisanje.php

```
<?php
    session_start();
    include_once "baza/db.php";
    $korisnicko_ime=$_SESSION["korisnik"];
?>
```

```
<!-- Poruka upozorenja da se narednom akcijom vrši brisanje naloga
-- ukoliko je potvrđeno brisanje poziva se skripta briši.php
-- inače se korisnik vraća na početnu stranu
-->
```

```
<script>
    if (confirm("Da li ste sigurni da želite da obrišete nalog?")) {
        location.href = "brisi.php";
    }
    else
        location.href = "index.php";
</script>;
```

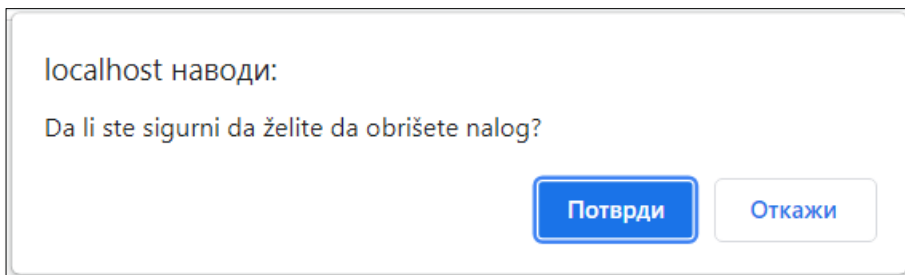
brisi.php

```
<?php
    session_start();
    include_once "baza/db.php";
```

```
$korisnicko_ime=$_SESSION["korisnik"];  
$anketa=veza();
```

```
/* Brisanje podataka o ulogovanom korisniku iz  
* tabele Korisnik  
*/
```

```
if ( brisanje_korisnika($anketa,$korisnicko_ime))  
    { echo '<script> alert("Uspešno brisanje naloga");</script>';  
      session_destroy();  
    }  
else  
    echo '<script> alert("Problem sa brisanjem");</script>';  
raskid_veze($anketa);  
header("Location:index.php");  
?>
```



Slika 42. Poruka skripte nakon uspešnog brisanja naloga

Primer 20. Slanje elektronske pošte

Neizostavni deo internet aplikacija je slanje elektronske pošte putem aplikacije. Prikazaćemo primer verifikacije mejla koji je korisnik naveo pri registraciji. Nakon upisa podataka o korisniku u tabelu Korisnik poziva se skripta *potvrdi_mail.php* kojom se šalje poruka sa verifikacionim kodom na korisnikov e-mail (slika 43). Takođe u ovom primeru biće prikazano i resetovanje lozinke ukoliko je korisnik zaboravio (*zaboravljena_lozinka.php*, slika 44, slika 45). Link ka skripti za resetovanje lozinke nalazi se u *login.php* skripti. Za ove aktivnosti potrebno je proširenje baze tabelom u kojoj će se čuvati kodovi za verifikaciju mejla i resetovanje lozinke. Za slanje mejla koristićemo *mail()* funkciju PHP-a.

Kreiranje tabele za verifikaciju mejla

-- Tabela Verifikacioni_kodovi čuva generisani kod za korisnika koji mu se šalje putem mejla

```
CREATE TABLE Verifikacioni_kodovi (  
    korisnicko_ime varchar(255) NOT NULL,  
    email varchar(50) NOT NULL,  
    kod varchar(250) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

baza/maildb.php

```
<?php
```

```
/* Funkcija postoji_verifikacioni_kod proverava da li  
 * postoji aktivan odgovarajući kod za datog korisnika  
 * sa datom e-mail adresom (vraća broj redova)  
 */
```

```
function postoji_verifikacioni_kod($konekcija, $korisnik, $email, $kod)  
{  
    $sql=" SELECT * FROM Verifikacioni_kodovi  
        WHERE  
        korisnicko_ime='$korisnik'  
        AND kod='$kod'  
        AND email='$email'";  
    $rezultat=$konekcija->query($sql);  
    return $rezultat->num_rows;
```

```
}
```

```
/* Funkcija postoji_verifikacioni_kod_e provera da li postoji  
* aktivan odgovarajući kod za dati mejl bez informacije  
* o korisničkom imenu (vraća broj redova)  
*/
```

```
function postoji_verifikacioni_kod_e($konekcija, $email, $kod)  
{  
    $sql=" SELECT * FROM Verifikacioni_kodovi  
        WHERE  
        kod='$kod'  
        AND email='$email';  
    $rezultat=$konekcija->query($sql);  
    return $rezultat->num_rows;  
}
```

```
/* Funkcija upis_verifikacionog_koda vrši upisivanje verifikacionog  
* koda za odgovarajućeg korisnika i mejl */
```

```
function upis_verifikacionog_koda($konekcija, $korisnicko_ime, $email,  
$kod)  
{  
    $sql="INSERT INTO Verifikacioni_kodovi  
        (korisnicko_ime, email, kod)  
        VALUES  
        ($korisnicko_ime, $email, $kod);  
    return ($konekcija->query($sql));  
}
```

```
/* Funkcija brisanje_verifikacionog_koda vrši brisanje  
* odgovarajućeg korisničkog koda iz tabele  
*/
```

```
function brisanje_verifikacionog_koda($konekcija, $korisnik, $kod)  
{  
    $sql=" DELETE FROM Verifikacioni_kodovi  
        WHERE  
        korisnicko_ime='$korisnik'  
        AND kod='$kod' ";  
    $rezultat=$konekcija->query($sql);  
    return $rezultat;
```



```
    }

    /* Funkcija aktiviraj_korisnika postavlja podatak o stanju
    * korisničkog naloga na 'aktiviran'
    */
function aktiviraj_korisnika($konekcija,$korisnicko_ime)
{
    $sql="UPDATE Korisnik SET
        stanje='aktiviran'
        WHERE
            korisnicko_ime='$korisnicko_ime'";
    return ($konekcija->query($sql));
}

/* Funkcija promeni_lozinku_e menja lozinku korisnika
    * sa zadatim mejlom
    */
function promeni_lozinku_e($konekcija,$email,$lozinka)
{
    $kriptovanaložinka = md5($lozinka);
    $sql="UPDATE korisnik SET
        lozinka='$kriptovanaložinka'
        WHERE
            email='$email' ";
    $rezultat=$konekcija->query($sql);
    return $rezultat;
}
?>
```

potvrdi_mail.php

<?php

```
/* Ova skripta se poziva iz registracijavall.php skripte nakon
    * uspešnog upisa podataka o korisniku u tabelu Korisnik i tim
    * pozivom se GET metodom šalju podaci o e-mail-u i korisničkom imenu.
    * potvrdi_mail.php?email=$email&korisnik=$korisnicko_ime
    */
include_once "baza/db.php";
include_once "baza/maildb.php";
```

//Preuzimanje podataka prosleđenih GET metodom

```
if(!isset($_GET['email'])){
    exit("Nije pronađen mejl za potvrdu.");
}

if(!isset($_GET['korisnik'])){
    exit("Nije pronađeno korisničko ime za potvrdu.");
}
```

```
$korisnik = $_GET['korisnik'];
$email = $_GET['email'];
```

//Kreiranje jedinstvenog koda

```
$kod = uniqid(true);
```

//Upisuju se podaci za validaciju u bazu

```
$anketa=veza();
upis_verifikacionog_koda($anketa,$korisnik,$email,$kod);
raskid_veze($anketa);
```

// Kreiranje linka do skripte za verifikaciju mejla (verifikacija.php)

```
$url="http://".$_SERVER["HTTP_HOST"].
dirname($_SERVER["PHP_SELF"])."/verifikacija.php?kod=$kod";
```

// Kreiramo telo elektronske poruke

```
$body = "
<h2 style='font-weight:400;'>
    Da bi ste završili registraciju pokrenite sledeći link:
    <br>
    <a href=".$url.">Završi registraciju!!!</a>
</h2>";
```

// Predmet elektronske poruke

```
$subject = "Potvrda registracije";
```

```
/* Za slanje mejla čiji sadržaj je HTML moraju se u poruku ubaciti
* dodatna polja koja opisuju format poruke
*/
```

```
$headers[] = 'MIME-Version: 1.0';
```

```
$headers[] = 'Content-type: text/html; charset=iso-8859-1';
```

```
//Dodajemo primaoca i pošiljaoca
```

```
$headers[] = 'To: $to_email';
```

```
$headers[] = 'From: programiranje.ia@gmail.com';
```

```
// Funkcija mail vrši slanje mejla
```

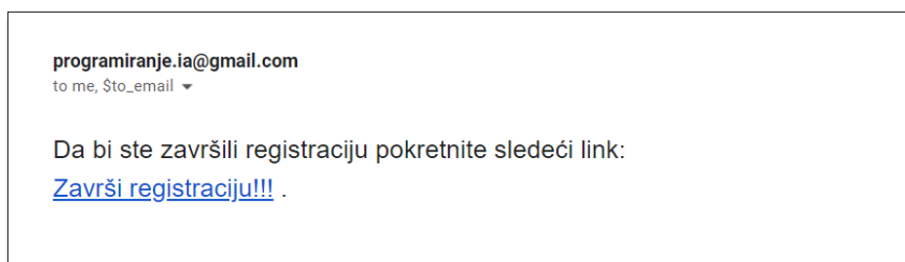
```
if (mail($to_email, $subject, $body, implode("\r\n", $headers)))
```

```
    { echo " Poruka za potvrdu registracije poslata je na $to_email...";}
```

```
else
```

```
    { echo "Problem pri slanju poruke za registraciju!"; }
```

```
?>
```



Slika 43. Poruka za verifikaciju mejla sa likom na primer:

<verifikacija.php?kod=1620449b80c639&email=mina@gmail.com&korisnik=mina>

Ovaj verifikacioni kod se upisuje u tabelu Verifikacioni_kodovi i nakon poziva skripte *verifikacija.php* sa ovim kodom biće aktiviran odgovarajući korisnički nalog i izbrisan kod iz tabele kodova.

verifikacija.php

```
<?php
```

```
include_once "baza/db.php";
```

```
include_once "baza/maildb.php";
```

```
if(!isset($_GET['kod'])){
```

```
    exit("Nije pronađen kod za potvrdu.");
```

```
}
```

```
if(!isset($_GET['email'])){
```

```
        exit("Nije pronađen mejl za potvrdu.");
    }

    if(!isset($_GET['korisnik'])){
        exit("Nije pronađeno korisničko ime za potvrdu.");
    }

    //Preuzimaju se prosleđeni podaci
    $kod = $_GET['kod'];
    $korisnik = $_GET['korisnik'];
    $to_email = $_GET['email'];

    /* Proverava se da li postoji kod za dati mejl i korisničko ime
    * ako postoji aktivira se nalog odgovarajućeg korisnika
    * polje stanje u tabeli Korisnik se postavlja na aktivno
    * i briše se verifikacioni kod
    */
    $anketa=veza();
    if(postoji_verifikacioni_kod($anketa,$korisnik,$to_email,$kod)>0)
    {
        if(aktiviraj_korisnika($anketa,$korisnik)){
            if(!brisanje_verifikacionog_koda($anketa,$korisnik,$kod))
                echo '<script> alert("Greška pri brisanju koda!")</script>';
            echo '<script> alert("Korisnički nalog je aktiviran!")</script>';
        }
        else
            echo '<script> alert("Korisnički nalog nije aktiviran!")</script>';
        raskid_veze($anketa);
    }
    else
    {
        raskid_veze($anketa);
        exit("Nije validan kod.");
    }
?>
```

Ukoliko korisnik zaboravi lozinku, na strani za logovanje postoji link koji vodi do skripte zaboravljena lozinka i daje korisniku mogućnost da unese e-mail na koji će mu biti poslata poruka za resetovanje lozinke.

Slika 44. Forma za resetovanje lozinke

zaboravljena_lozinka.php

```
<?php include "navigacija.php" ?>
<main>

<!-- Forma za resetovanje lozinke
-- POST metodom šalje e-mail skripti za generisanje mejla
-- sa linkom za resetovanje lozinke
-->
<form action="mejl_zaboravljena_lozinka.php" method="post" >
  <div>
    <div>
      </br>
      <label> Unesite e-mail za resetovanje poruke</label>
      <input type="email" name="email" placeholder="E-mail" required>
    </div>
  </div>
  <div class="text-center">
    <button type="submit" >Resetuj lozinku</button>
  </div>
```

```
</form>
</main>
<?php include "footer.php"; ?>
```

mejl_zaboravljena_lozinka.php

```
<?php
include_once "baza/db.php";
include_once "baza/maildb.php";

if(!isset($_POST['email'])){
    exit("Nije pronađen mejl za promenu lozinke.");
}
```

//Preuzimaju se prosleđeni podaci

```
$to_email = $_POST['email'];
```

//Kreiranje jedinstvenog koda

```
$kod = uniqid(true);
```

//Upisuju se podaci za validaciju u tabelu

```
$anketa=veza();
```

```
upis_verifikacionog_koda($anketa,'korisnik',$to_email,$kod);
raskid_veze($anketa);
```

/ Kreiramo link do skripte za resetovanje lozinke nova_lozinka.php */*

```
$url="http://".$_SERVER["HTTP_HOST"].
```

```
dirname($_SERVER["PHP_SELF"]).
```

```
"/nova_lozinka.php?kod=$kod&email=$to_email";
```

/ Kreiramo telo elektronske poruke*/*

```
$body = "
```

```
<h2 style='font-weight:400;'>
```

```
Da bi ste promenili lozinku pokrenite link: <br>
```

```
<a href=".$url.">Promeni lozinku!!!</a>
```

```
</h2>";
```

/ Predmet elektronske poruke*/*

```
$subject = "Promena lozinke";
```

```
/* Dodatna polja koja opisuju format poruke*/
```

```
$headers[] = 'MIME-Version: 1.0';
```

```
$headers[] = 'Content-type: text/html; charset=iso-8859-1';
```

```
//Dodajemo primaoca i pošiljaoca
```

```
$headers[] = 'To: $to_email';
```

```
$headers[] = 'From: programiranje.ia@gmail.com';
```

```
// Funkcija mail vrši slanje mejla
```

```
if (mail($to_email, $subject, $body, implode("\r\n", $headers)))
```

```
    {echo " Poruka za promenu lozinke poslata je na $to_email...";}
```

```
else
```

```
    { echo "Problem pri slanju poruke za registraciju!"; }
```

```
?>
```

nova_lozinka.php

```
<?php
```

```
    include_once "baza/db.php";
```

```
    include_once "baza/maildb.php";
```

```
    if(!isset($_GET['kod'])){
```

```
        exit("Nije pronađen kod za potvrdu.");
```

```
    }
```

```
    if(!isset($_GET['email'])){
```

```
        exit("Nije pronađen mejl za potvrdu.");
```

```
    }
```

```
    $email=$_GET["email"];
```

```
    $kod=$_GET["kod"];
```

```
?>
```

```
<main>
```

```
<!-- Forma za unos nove lozinke koja POST metodom
```

```
-- prosleđuje podatke za resetovanje lozinke skripti
```

```
-- reset.php
```

```
-->
```

```
<form action="reset.php" method="POST" >
```

```
<div>
```

```
</br>
<label>Za promenu lozinke morate uneti novu lozinku:</label>
</br>

<!-- Skriveno polje koje nosi informaciju o mejlu korisnika i kodu -->
<input name="email" value="<?php echo $email; ?>" type="hidden" >

<!-- Skriveno polje koje nosi informaciju o mejlu korisnika -->
<input name="kod" value="<?php echo $kod; ?>" type="hidden" >
<div >
  </br>
  <label>Nova lozinka</label>
  <input type="password" name="lozinka1" placeholder="nova lozinka">
</div>
<div >
  </br>
  <label>Ponovite novu lozinku</label>
  <input type="password" name="lozinka2" placeholder="nova lozinka">
</div>
</div>

<!--Ukoliko korisnik želi da odustane vraća se na početnu stranu-->
<div class="text-center">
  <a href="index.php"> Odustani od izemene i vrati se na početnu stranu
  </a>
  <br><br>
  <button type="submit" >Promeni lozinku</button>
</div>
</form>
</main>
<?php include "footer.php"; ?>
```

reset.php

```
<?php
  include_once "baza/db.php";
  include_once "baza/maildb.php";
  if (isset($_POST["kod"]) && isset($_POST["email"]) &&
      isset($_POST["lozinka1"]) && isset($_POST["lozinka2"]))
  {
```



```
$email = $_POST["email"];
$lozinka1 = $_POST["lozinka1"];
$lozinka2 = $_POST["lozinka2"];

if(strcmp($lozinka1, $lozinka2) != 0)
    $greska .= " Lozinke se ne poklapaju. ";
$kod = $_POST["kod"];
}

/* Proverava se da li postoji kod za dati mejl
* ako postoji vrši se promena lozinke kod korisnika sa datim mejlom
* i briše se verifikacioni kod
* nakon uspešno promenjene lozinke otvara se login strana
*/
$anketa=veza();
if(postoji_verifikacioni_kod_e($anketa,$email,$kod)>0)
{
    if(promeni_lozinku_e($anketa,$email,$lozinka1))
    {
        if(!brisanje_verifikacionog_koda($anketa,'korisnik',$kod))
            echo '<script> alert("Greška pri brisanju koda!")</script>';
        raskid_veze($anketa);
        echo '<script> if (confirm("Vaša lozinka je uspešno promenjena!")){
            location.href = "login.php"; </script>';
    }
    else
        echo '<script> alert("GREŠKA: Vaša lozinka nije
promenjena!")</script>';
}
else
    echo '<script> alert("Nevalidnan kod!")</script>';
raskid_veze($anketa);
?>
```

Za promenu lozinke morate uneti novu lozinku:

Nova lozinka

Ponovite novu lozinku

[Odustani od izmene i vrati se na početnu stranu](#)

[Nazad na vrh](#)

© 2021 [Kontakt](#)

Slika 45. Forma za unos nove lozinke
(nova_lozinka.php?kod=16204539d806a4&email=patap@cnw.com)

Literatura

1. Bush, V. (1945). As we may think. The atlantic monthly, 176(1), 101-108.
2. Kurose, J., & Ross, K. (2017). Computer Networking: A Top-Down Approach, Global Edition.
3. Kajan, E. (2014). Internet programiranje
4. Niederst Robbins, J. (2020). Learning Web Design: a beginner's guide to HTML, CSS, Javascript, and Web Graphics. O'Reilly Media.
5. Kereki, F. (2018). Modern JavaScript Web Development Cookbook: Easy solutions to common and everyday JavaScript development problems. Packt Publishing Ltd.
6. Tatroe, K., & MacIntyre, P. (2020). Programming PHP: Creating Dynamic Web Pages. O'Reilly Media.