**Instalacija i priprema za rad**

**00:23:13**

Jedinica: 1 od 19

**Rezime**

* PHP je jezik koji se interpretira, te ne može raditi bez programa za interpretaciju;
* Značajan deo sintakse PHP-a je identičan sintaksi jezika C;
* Najčešća upotreba PHP-a je u izgradnji dinamičkih web aplikacija;
* PHP ne može emitovati HTML strane bez web servera ;
* PHP se može izvršavati na različitim web serverima;
* PHP kod može biti pisan u bilo kom alatu za editovanje teksta;
* Da bi dokument bio prosleđen PHP-u, mora imati ekstenziju .php;
* Da bi skripta unutar dokumenta bila startovana, mora biti aktivirana PHP tagovima **<?php ?>**ili**<? ?>**;
* Uz pomoć PHP-a nije moguće praviti animacije u pretraživaču.

U ovoj lekciji naučićemo kako da podesimo razvojno okruženje i šta je sve neophodno da imamo kako bismo radili sa PHP-om. PHP je skript programski jezik opšte namene. Prvobitno je dizajniran kao jezik za kreiranje dinamičkih web strana. Omogućava brzo procesiranje i učitavanje strana, jednostavan je za razumevanje i korišćenje i izvršava se na skoro svim operativnim sistemima. PHP se interpretira od strane web servera i generiše HTML kod ili neki drugi izlaz koji klijenti (koji su najčešće [web čitači](https://www.link-elearning.com/linkdl/opisPojma.php?id=139086)) mogu da prepoznaju.

Da biste mogli da kreirate interaktivne web strane i da izvršavate PHP programe, neophodno je da imate pristup nekom serveru koji podržava PHP. Međutim, često je u toku kreiranja nekog interaktivnog web sajta pomoću PHP-a veoma nepraktično da svaki put kada napravite neku PHP stranu morate da je prebacite na udaljeni server kako biste je testirali. Zbog toga se preporučuje da PHP instalirate lokalno, na svom računaru.

Neki operativni sistemi, kao što su Linux i mnoge verzije Unixa, isporučuju se sa već instaliranim PHP-om. Kod drugih operativnih sistema, kao što su Windows ili Mac OSX, morate to da uradite sami.

Dakle, PHP je skript jezik čija je primarna, ali ne i jedina funkcija (jer je u pitanju jezik opšte namene) obrada podataka na web serveru i njihova implementacija u [HTML kod](https://www.link-elearning.com/linkdl/opisPojma.php?id=139089). U pitanju je direktan derivat jezika C i veći deo sintakse je identičan ovom programskom jeziku. Besplatan je i njegova primena je veoma jednostavna.

Kako bismo omogućili dalji rad sa PHP-om, potrebno je ispuniti nekoliko preduslova:

* + **Instalirati PHP na računaru na kome želimo da se izvršava**PHP skripte se interpretiraju, što znači da se ne startuju direktno, već putem programa za interpretaciju. Instalacijom PHP paketa na računaru, ova komponenta biće omogućena;
  + **Instalirati web server na računaru na kome želimo da se izvršava**PHP-u zapravo i nije potreban web server da bi se izvršavao, već samo interpreter. Ali, s obzirom na to da ćemo se držati PHP-a u kontekstu weba, neophodan je i web server;
  + **Instalirati bazu podataka na računaru na kome želimo da se izvršava**Ovo takođe nije neophodna komponenta, ali s obzirom na to da je PHP produžena ruka web strana ka podacima, baza podataka je veoma preporučljiva;
  + **Odabrati alat za PHP kodiranje**[Izvorni PHP kod](https://www.link-elearning.com/linkdl/opisPojma.php?id=139090) je moguće pisati u najprimitivnijim alatima za editovanje teksta (npr. Notepad). Ipak, najbolje je koristiti neki alat koji prepoznaje (i markira) PHP naredbe. Moguće je (i poželjno) u ovu svrhu koristiti i neki od programa za proizvodnju HTML strana (Adobe Dreamweaver, NetBeans, Eclipse, Notepad++, SublimeText) koji u sebi ima i deo za dizajniranje. Za potrebe kursa, koristićemo Notepad++.

**Instalacija PHP-a**

Postoji više načina da se PHP instalira na operativni sistem. Ukoliko je reč o Linuxu, instalacija se vrši pre ili posle instalacije samog OS-a pridruživanjem odgovarajućih servisa iz distribucije OS-a.

Kada je reč o MS Windows operativnim sistemima, instalacija se generalno može izvršiti na dva načina. Jedan način je *ručna instalacija*, komponentu po komponentu, a drugi instalacija nekog paketa koji u sebi sadrži ceo sistem. Ovakvi paketi su izuzetno praktični, i mi ćemo se orijentisati na jedan od njih – ***WampServer***. Njime ćemo pokriti tri gorepomenuta uslova (PHP, web server i bazu).

Za potrebe ovih predavanja, preuzmite i instalirajte WampServer sa adrese: (<http://www.wampserver.com/en/>).

**Instalacija alata za kodiranje**

Za potrebe predavanja, preuzmite i instalirajte alat Notepad++ sa adrese: (<http://notepad-plus-plus.org/>).

Nakon pravilnog podešavanja svih servisa i pripreme alata, moguće je početi sa upotrebom sistema i proizvodnjom PHP aplikacija. Ipak, da bismo olakšali rad, potrebno je upoznati se sa osnovnim načinom funkcionisanja ovakvog sistema.

Pre svega, treba pomenuti da pravila pomenuta gore nisu striktna kao što se isprva možda čine. Na primer, ukoliko čak i ne posedujemo PHP interpreter na sopstvenom računaru, to ne znači da ne možemo da stvorimo kod, a posle ga aktiviramo na drugom računaru. Takođe, MySQL server / baza podataka ne moraju biti na istom računaru na kome je i web server. Web server ne mora biti samo na jednom računaru, niti je uopšte neophodno da to bude baš **Apache** web server (moguće je PHP izvršiti i na MS IIS web serveru)... Ukratko, postoje razne varijacije koje se mogu implementirati prema potrebama celog sistema, ali koje su takođe irelevantne u trenutku proučavanja osnova programskog jezika. Zato, u našem upoznavanju sa jezikom koristimo linearnu proceduru, iako možda ona ne daje najefikasnije rezultate ili nije savršeno preslikana slika nekog realnog sistema. Ono što je bitno to je da u radnom okruženju stvorimo uslove bliske produkcionom, kako bi implementacija prošla što bezbolnije.

**Koordinacija PHP-a i web servera**

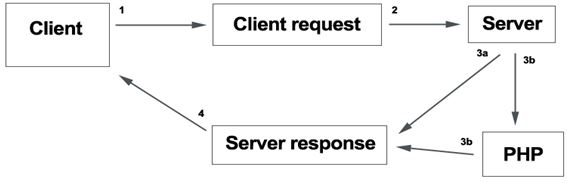
Uloga svakog web servera svodi se, manje ili više, na jednu funkciju – a to je prihvatanje određenog **zahteva** od korisnika putem interneta, obrada tog zahteva, što podrazumeva pronalaženje dokumenta koji je u tom zahtevu sadržan, te slanje **odgovora** korisniku.

Ukoliko korisnički zahtev sadrži naziv i tip dokumenta koji postoji na web serveru u nekom statičkom obliku, web server će jednostavno preuzeti dokument sa fajl sistema i isporučiti ga internetom korisniku. Međutim, ukoliko zahtev ne podrazumeva već postojeći dokument, već zahteva sadržaj koji će biti dinamički generisan, web server mora stvoriti odgovor tako što će startovati skriptu koju zahtev sadrži, te njen rezultat implementirati u dokument u kome se skripta nalazi i proslediti korisniku. Na kraju, s obzirom na to da je pomenuta skripta/program implementirana u neki dokument, korisnik će dobiti svoj rezultat u formi validnog HTML dokumenta.

Prilikom ovog procesa, web server i klijent (a kada kažemo [klijent](https://www.link-elearning.com/linkdl/opisPojma.php?id=139087), mislimo na pretraživač IE, FireFox, Chrome...) koriste isti jezik – protokol za komunikaciju. Taj jezik naziva se **HTTP** (HyperText Transfer Protocol) i osnovni je način komunikacije između računara na internetu.

Gde se u sve to uklapa PHP? Uloga PHP-a je u stvari da prevede i startuje pomenutu skriptu, te da prosledi dobijeni rezultat klijentu. Treba znati da PHP nije jedini jezik koji može uraditi tako nešto. Postoje i drugi jezici koji se dobro snalaze u ovom poslu (C#, VB, Java...), uz preduslov da [server](https://www.link-elearning.com/linkdl/opisPojma.php?id=139087) zna koji od njih i kada treba da startuje i gde da pronađe interpretere za njih.

Ako bismo hteli da ono što smo napisali u prethodnom pasusu prikažemo kroz sliku, ona bi izgledala odprilike ovako:

**

*1.1. Ilustracija transporta podataka*

Prethodna slika ilustruje kretanje podataka od klijenta do servera. Isprva, klijent generiše zahtev i pakuje ga u formu HTTP zahteva, nakon čega ga šalje (1). Iako zahtev može imati različite karakteristike, on u osnovi ima sledeći sadržaj (2):

GET /index.html HTTP/1.1  
Host: www.some\_site.com

Kada zahtev stigne do servera, server razmatra na koji način će ga tretirati. Vidimo da je u prvom redu zahteva napisano index.html. To je naziv dokumenta koji klijent traži od servera. S obzirom na to da dokument ima ekstenziju html, server zaključuje da ne mora da aktivira serverski kod, već samo prosleđuje dokument nazad klijentu, jer je HTML jezik koji klijent u potpunosti razume (3a). Ukoliko bi dokument imao ekstenziju .php, server bi angažovao php interpreter i prosledio mu dokument (3b). Php interpreter bi preveo i izvršio dokument (odnosno, PHP kod koji se nalazi u njemu), zatim bi kreirao HTTP odgovor (4) i prosledio ga klijentu. Ovakvu interpretaciju je neophodno izvršiti na serveru jer klijent (web pretraživač) ne razume jezik PHP-a i ukoliko bi i došao do njega, na strani bi ga prikazao kao običan tekst jer ne bi bio u stanju da ga interpretira. HTTP odgovor nije mnogo različit od HTTP zahteva. U njemu server opisuje dokument koji šalje klijentu. Jedan deo odgovora izgleda otprilike ovako (4):

HTTP/1.1 200 OK  
Connection: close  
Content-Type: text/html; charset=UTF-8

U drugom delu odgovora nalazi se sam dokument.

Po dobijanju odgovora, pregledač ga analizira, a zatim preuzima dokument, prevodi HTML sadržaj u formu razumljivu korisniku i prikazuje ga na strani.

**Identifikacija skript jezika**

Da bi web server prepoznao zahtev korisnika za izvršenjem neke skripte, potrebno je da bude ispunjen najmanje jedan uslov. Taj uslov je da dokument ima odgovarajuću ekstenziju. U slučaju PHP-a, ova ekstenzija je .php. Da je u pitanju neki drugi jezik, ekstenzija bi bila drugačija (npr. C#, VB, J# - .aspx ili .asp, java-.jsp itd.).

Drugi uslov je da dokument uopšte sadrži skriptu u sebi. Ovo nije uistinu preduslov da se dokument prosledi korisniku, jer će, čak i ukoliko ne sadrži skriptu, njegov sadržaj biti prosleđen. Jednostavno, web server će startovati PHP (interpreter) koji će proći kroz dokument, ali s obzirom na to da u njemu nema skripte, ništa neće biti startovano, pa će takav dokument u svom izvornom obliku biti prosleđen klijentu. Ali, ukoliko želimo (a želimo) da se neka skripta u našem dokumentu i izvrši, moramo to dati do znanja interpreteru. Odgovor leži u **parsiranju**.

Koji od sledećih tagova predstavlja validno napisan PHP tag?

Znamo da se ceo HTML svet sastoji od tagova, odnosno, markera koji ograničavaju i identifikuju određene celine. Kada se strana sa ekstenzijom .php prosledi PHP-u, biva [parsirana](https://www.link-elearning.com/linkdl/opisPojma.php?id=139082) tako što se analiziraju svi njeni tagovi. PHP sve „nezanimljive” tagove prosleđuje u formi u kojoj ih je i pročitao sve dok ne dođe do taga koji se tiče samo njega. Taj tag se obeležava sa **<?php ?>, <? ?>,<?= ?>** ili <script language="php"> </script>. U veoma retkim slučajevima za PHP se koriste i sledeći tagovi **<% %>**. Ovi tagovi poznati su još i kao ASP tagovi, i potrebno ih je posebno omogućiti u podešavanjima PHP-a kako bi mogli da se koriste (do verzije PHP-a 5.4 isto važi i za tagove <? ?> i <?= ?>).

|  |
| --- |
| **Napomena**  PHP7 predviđa ukidanja ASP (<% %>) i script ( ). Radi kompatibilnosti preporuka je da se jedino koriste tagovi <?php ?> ili <?= ?>. |

Kada PHP parser prepozna navedeni tag, dolazi do aktivacije sadržaja unutar taga i skripta biva izvršena.

Treba obratiti pažnju na to da smo PHP tagove označili sa **<?php ?>**, **<? ?>**ili**<?= ?>,** iako ovo „ili” ne mora u svakom slučaju biti pravilo. Naime, podrazumevano setovanje konfiguracije PHP-a je da prepoznaje samo jasno naznačene, dugačke otvarajuće tagove (<?php ?>). Ovo setovanje najčešće biva promenjeno tako da PHP prepoznaje i skraćene tagove, ali za slučaj da su setovanja na produkcionom serveru (onom na kome želimo da trajno držimo naš sajt) ostala na podrazumevanim vrednostima, dugački otvarajući tag je najsigurnija varijanta, jer najverovatnije nećemo (ukoliko setovanja ne promenimo na nivou skripte) imati mogućnost pristupa konfiguracionim fajlovima za slučaj da je ova opcija isključena. U sklopu kursa koristićemo isključivo dugačke otvarajuće tagove.

Veoma je bitno (iako je logično iz prethodnog teksta) naglasiti da PHP ne omogućava nikakvu aktivnost nakon što je dokument prosleđen klijentu. Dakle, onog trenutka kada je PHP izvršio skriptu, generisao HTML uz pomoć te skripte i prosledio stranu, toj strani više ne može pristupiti. Zbog toga u PHP-u ne možemo napraviti animacije ili bilo kakve pokrete u [pretraživaču](https://www.link-elearning.com/linkdl/opisPojma.php?id=139085). Za to koristimo druge programske alate, poput JavaScripta, ActionScripta, Silverlighta ili Jave (Java Applet i JavaFX).

**Zaključak**

Iz ovog uvoda zaključuje se da će krajnji produkt učenja biti proizvodnja PHP skripti u službi HTML-a. To upravo i jeste cilj jer PHP i dominira u pomenutoj infrastrukturi. Bez HTML-a, učenje PHP-a svelo bi se na suvoparno baratanje brojevima i matematičkim i logičkim operacijama, uslovima i ostalim kvalifikatorima programskih jezika. Bez ovih komponenti, svakako u PHP-u nećemo daleko dogurati, ali će učenje svakako biti zanimljivije uz povezivanje sa HTML-om.

**Više podataka**

**Nastanak PHP-a**

PHP se u svom najosnovnijem obliku pojavio 1995. godine. Stvorio ga je danski programer Rasmus Lerdorf u želji da putem skripte prati broj posetilaca web stranice svog CV-ja. Rasmus je ovaj jezik nazvao **P**ersonal **H**ome **P**ages.

Ova primitivna verzija jezika (nasleđenog od jezika C) bila je samo osnova za njegov dalji razvoj, pa je Ramus vrlo brzo ovaj svoj projekat deklarisao kao Open Source, kako bi ga uz pomoć većeg broja programera razvio i unapredio.

Najveće unapređenje jezika izvršili su Zeev Suraski i Andi Gutman 1997. godine, preradivši kompletan glavni parser jezika. Dodavši mu novi naziv (PHP/FI), dodali su i podršku za bazu podataka, čime je jezik dobio formu koju u manjoj ili većoj meri poseduje i danas.

U ovom trenutku smatra se da oko 60% web servera na svetu funkcioniše na Apache tehnologiji, pa se samim tim, u velikom broju slučajeva, podrazumeva i obrada podataka u PHP-u. Ako se ovome doda i činjenica da se PHP (doduše, ne tako često) koristi i na drugim platformama (IIS), jasno je da je u pitanju jezik visoke funkcionalnosti.

Suraski i Gutman (ujedinjeni pod imenom **Zend**) i danas su glavni developeri jezika PHP. Puno ime PHP-a danas je PHP-Hypertext Processor.

**Ručna instalacija PHP-a**

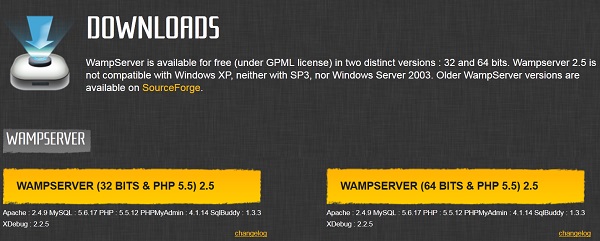
Binarni paket sa instalacijom PHP-a se nalazi na adresi <http://www.php.net/downloads.php>. U okviru datoteke koje preuzmete nalazi se i uputstvo za instalaciju. Na primer, za Windows 7 operativni sistem neophodno je uraditi sledeće:

* 1. instalirati neki web server (Apache ili Microsoftov Internet Information Server – IIS),
  2. instalirati PHP,
  3. instalaciju PHP-a povezati sa serverom.

Na gorepomenutoj adresi za preuzimanje instalacije PHP-a nalazi se nekoliko fajlova. Jedan od njih se naziva *installer.exe*. Ukoliko preuzmete ovaj fajl, njegovim pokretanjem će biti instaliran PHP i automatski podešeni neki od web servera.  Drugi način instalacije je pomoću .zip fajla sa iste adrese, ali tada morate sami da podešavate i web server. Kompletno uputstvo za instalaciju je sastavni deo ovog fajla.

**Detaljan opis preuzimanja i instalacije WampServera**

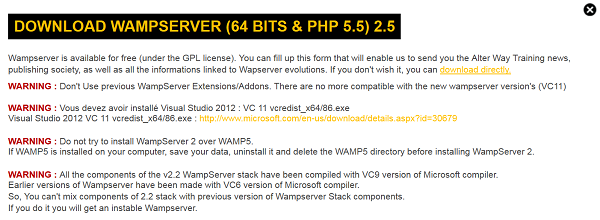
1. Pristupite [url](https://www.link-elearning.com/linkdl/opisPojma.php?id=139128) adresi  <http://www.wampserver.com/en/>,  
2. Odaberite verziju WampServera koja odgovara vašem operativnom sistemu,



*1.2. Izbor WampServer verzije*

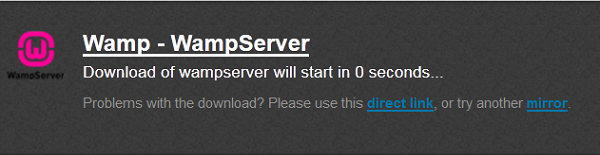
3. Selektujte hiperlink download directly,

|  |
| --- |
| **Napomena** Ukoliko je na vašem sistemu to potrebno, pre instalacije WAMP servera, instalirajte Visual Studio 2012 : VC 11 vcredist\_x64/86.exe sa datog linka. |



*1.3. WampServer Download*

4. Pojaviće vam se prikaz na vrhu stranice,

**

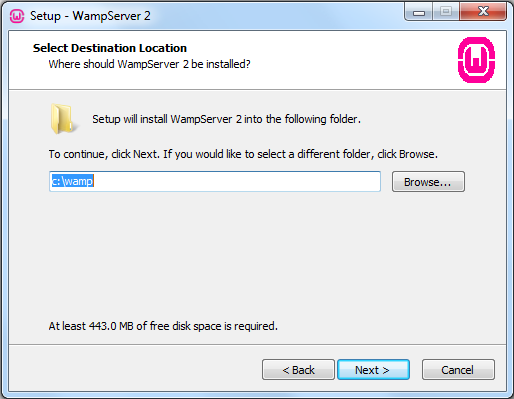
*1.4. WampServer direct link*

5. Odaberite lokaciju za snimanje na vašem disku i time započnite preuzimanje,  
6. Po završenom preuzimanju, pokrenite instalaciju skinutog programa,



*1.5. WampServer Setup Wizard*

7. Odabirom instalacionog foldera (što je jedini bitan interaktivni korak instalacije) odabraćete i folder gde će se nalaziti vaš MySQL Server, Apache server, kao i web sadržaj,

**

*1.6. WampServer Select Destination*

8. Korake za postavljanje podrazumevanog pretraživača i mail servera možete ostaviti na podrazumevanim vrednostima,  
9. U poslednjem koraku, ostavite selektovan „Launch wamp server now” opciju kako bi ceo sistem automatski bio startovan,  
10. U Trayu će se pojaviti ikonica WampServera. Ukoliko svi servisi rade pravilno, ikonica će izgledati kao na sledećoj slici:

**

*1.7. WampServer OK*

Sve ostale varijante ikonice, npr:



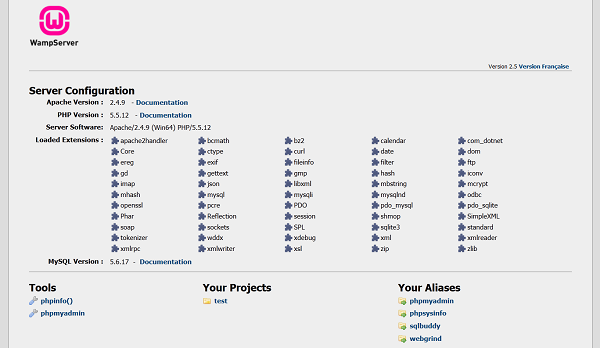
*1.8. WampServer Error*

označavaju da neki od servera nije u stanju da funkcioniše. Ovo se obično dešava kada je neki od TCP/IP portovapotrebnih za funkcionisanje servisa zatvoren (MySQL server: 3306 ili Apache server: 80). Jedan od najčešćih uzročnika je Skype, pa ga treba isključiti (kasnije će ovaj problem biti elegantnije rešen),

11. Testirajte pravilnost rada web servera tako što ćete:

* Startovati web pretraživač,
* Uneti u Address polje pretraživača adresu: http://localhost/ .

Rezultat bi trebalo da bude otvaranje sledeće strane:

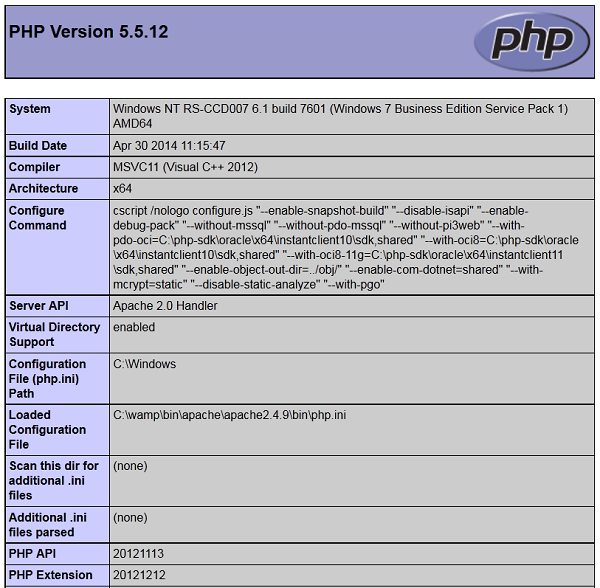
**

*1.9.WampServer*–*localhost*

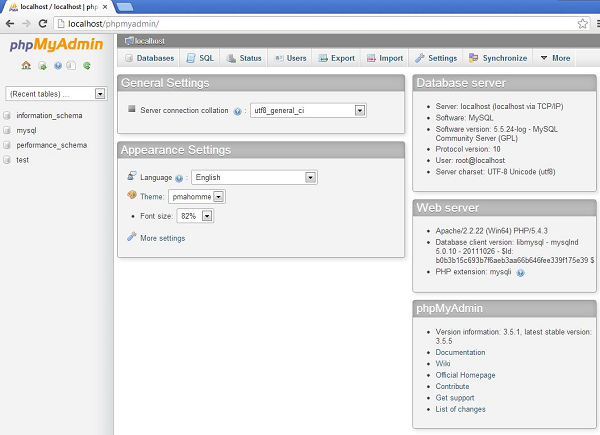
Ovakva strana u pretraživaču znači da web server funkcioniše.

Da biste proverili da li funkcionišu i ostali servisi (PHP i MySQL), odaberite jedan od dva hiperlinka ispod podnaslova Tools (*phpinfo()* za PHP i *phpmyadmin* za MySQL).

Ukoliko su rezultati dva prozora kao na sledećim slikama, znači da oba servisa (PHP i MySQL respektivno) funkcionišu i time je instalacija uspešno izvršena.



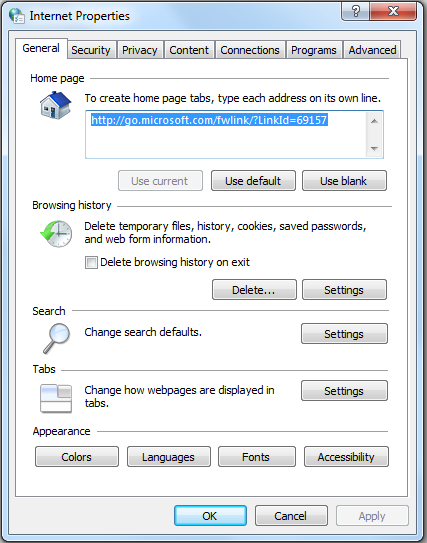
*1.10. PHP\_INFO*



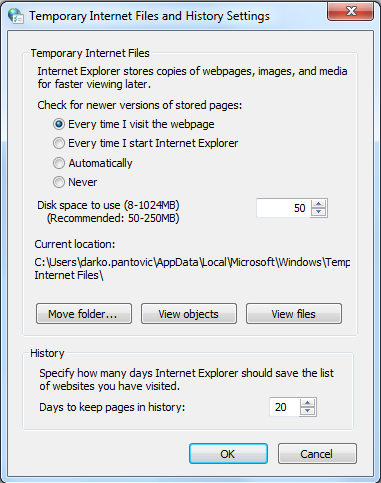
*1.11. PhpMyAdmin*

Na kraju, poželjno je omogućiti u pretraživaču učitavanje keširanih strana iznova, kako bi se izbeglo da pretraživač ne prikaže promene urađene u kodu strane na kojoj radimo.

Da bi se ovo ostvarilo, potrebno je otići na ***Tools > Internet Options*** stavku padajućeg menija **Internet Explorera**, selektovati opciju **Settings** iz okvira **Temporably Internet Files** a zatim u novootvorenom prozoru štiklirati stavku **Every visit to the page**. Ovo uradite čak i ako u ovom trenutku za to ne vidite poseban značaj.



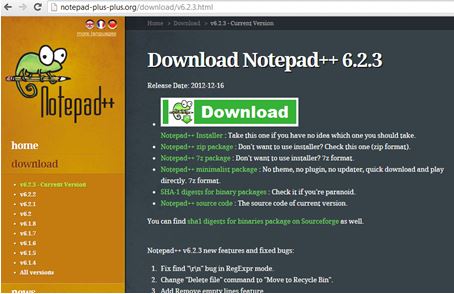
*1.12. Internet Properties*



*1.13. History Settings*

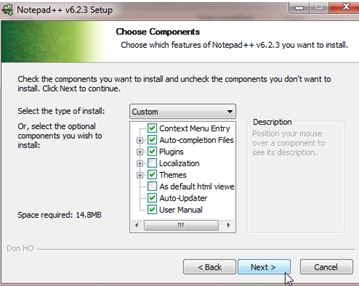
**Detaljan opis preuzimanja i instalacije editora Notepad++**

1. Pristupite adresi <http://notepad-plus.sourceforge.net/uk/site.htm>,
2. Odaberite opciju Download,



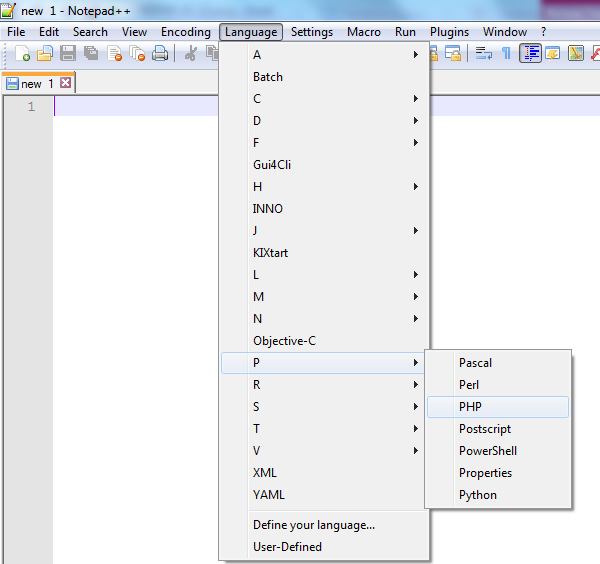
*1.14. Notepad++ Download*

1. Odaberite hiperlink Download,
2. Nakon što ste preuzeli fajl, pokrenite instalaciju. Podesite nekoliko linearnih koraka (čitanje licence i izbor foldera) sa komponentama prema sledećoj slici (ukoliko već nije tako podešen),

**

*1.15. Notepad++ Setup*

1. Na poslednjem koraku instalacije ostavite čekirano polje za startovanje aplikacije, i nakon startovanja, u samoj aplikaciji podesite jezik na PHP, nakon čega je možete zatvoriti.



*1.16. Notepad++ Language*

Nakon što je okruženje pripremljeno za rad, možete kreirati i sopstvenu PHP stranu. Pristupite direktorijumu c:/wamp/www (uz napomenu da vam je Wamp možda instaliran i u neki drugi direktorijum). Unutar direktorijuma www možete postaviti neki sopstveni fajl, u kome želite da se nalazi vaš php kod. Na primer, to može biti fajl test.php.

U fajlu test.php kreirajte sledeći sadržaj:

|  |  |
| --- | --- |
| 1  2  3 | <?php  echo "Hello!";  ?> |

A zatim u pregledaču otkucajte adresu do fajla:

http://localhost/test.php

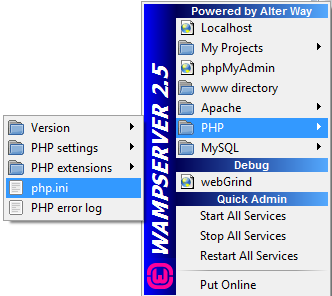
Ako ste sve ispravno uradili, na strani će biti prikazana poruka:

Hello!

**Rad sa konfiguracionim fajlovima**

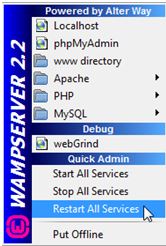
Wamp u svom korisničkom interfejsu omogućava jednostavan pristup svim konfiguracionim fajlovima servisa koji ispod njega funkcionišu. Da biste pristupili konfiguracionom fajlu za PHP, potrebno je:

* + Otvoriti opcije WampServera pritiskom na levi taster miša u Trayu,
  + Odabrati opciju **php.ini** iz podmenija **PHP,**



*1.17. WAMP*–*php.ini*

* + Promeniti željene opcije u konfiguracionom fajlu,
  + Snimiti konfiguracioni fajl,
  + Restartovati servise.

**

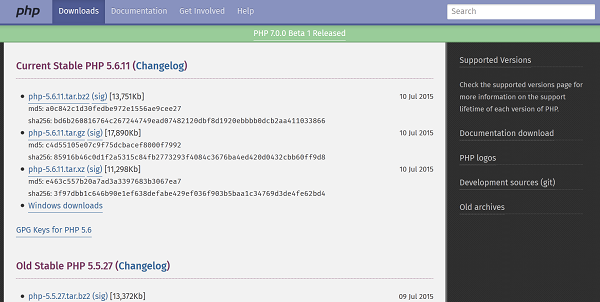
*1.18. WAMP*–*Restart All Services*

Isto ovo možemo uraditi i na drugačiji način, tako što ćemo otići u folder gde se nalazi PHP (a obično je to folder wamp/php) i ručno otvoriti i izmeniti *php.ini* fajl, ali i na jednostavniji način, tako što ćemo na korisničkom interfejsu Wampa odabrati opciju **PHP settings**i jednostavno uključiti ili isključiti željenu opciju.

Razumljivo da ovakav pristup važi samo kada je u pitanju WampServer okruženje.

**Instalacija PHP-a na Linux operativnom sistemu**

Skinite verziju PHP-a sa adrese <http://www.php.net/downloads.php>.



*1.19. PHP Download*

Zatim, iz konzole, locirajte fajl koji ste preuzeli (ime fajla će verovatno biti: *php5.x.x.tar.gz*, mada tačan naziv zavisi od verzije PHP-a koju ste skinuli) i pozicionirajte se u taj direktorijum.

Raspakujte arhivu koju ste skinuli uz pomoć komande:

tar –xzf php5.x.x.tar.gz

Potom se premestite u folder koji se automatski kreirao (php5.x.x) i startujte instalaciju naredbom:

|  |  |
| --- | --- |
| 1 | ./configure –prefix=/usr/local/php –with-apxs2=/usr/local/apache2/bin/apxs –with-mysql=/usr/local/mysql |

ili jednom od sledeće dve naredbe, ukoliko ne znate podatke o Apache serveru ili MySQL serveru:

|  |  |
| --- | --- |
| 1  2 | ./configure –prefix=/usr/local/php –with-mysql=/usr/local/mysql  ./configure –prefix=/usr/local/php |

Ovakva instalacija podrazumeva da već imate instaliran Apache i MySQL servere na računaru na kome se nalazi Linux. Zato instalaciji pridružujete putanje ove dve aplikacije.

Nakon uspešno izvršene instalacije, izvršite kompajliranje naredbom: make

i instalaciju naredbom: make install

**PHP Framework**

S obzirom na unapređenje i sve veće korisničke zahteve u upotrebu je ušao PHP Framework. Framework je struktura koja se obično sastoji od velikog broja klasa i koja treba da ubrza razvoj, održavanje i sigurnost web aplikacije. Neki od trenutno najpopularnijih frameworkova su: Zend, Cake, Yii, Symfony, Laravel.

Zend Framework je open source framework za razvijanje web aplikacija i servisa korišćenjem PHP 5.3+. S obzirom na ozbiljnost primene, koristi se u okviru objektno orijentisanih blokova koji izlaze iz okvira ovog kursa, ali nam o njegovoj upotrebi govori podatak o preko 15 miliona downloadova. Može se preuzeti sa linka:

<http://framework.zend.com/downloads/latest>

**

*1.20. Zend framework logo*

CakePHP je besplatan open source, brz razvojni framework za PHP. Prioriteti razvoja su omogućavanje rada programerima u strukturiranom i brzom načinu rada bez gubitaka fleksibilnosti. Unapređen je svim potrebnim alatima koji omogućavaju pisanje koda i baziranje na suštinu dolaska do rešenja. Može se preuzeti sa linka: <http://cakephp.org/>

**

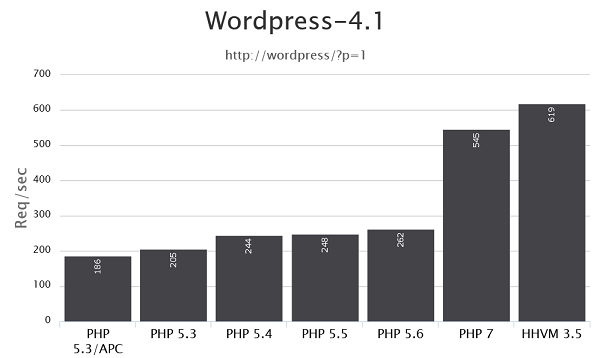
*1.21. CakePHP framework logo*

**Laravel** je [open-source](https://hr.wikipedia.org/wiki/Open-source) PHP [web framework](https://hr.wikipedia.org/w/index.php?title=Web_framework&action=edit&redlink=1). Baziran je na [Simfony framework](https://hr.wikipedia.org/w/index.php?title=Simfony_framework&action=edit&redlink=1)u. Razvio ga je [Taylor Otwell](https://hr.wikipedia.org/w/index.php?title=Taylor_Otwell&action=edit&redlink=1), sa ciljem da razvoje web aplikaciju prateći [Model-View-Controller](https://hr.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller) (MVC) arhitekturu. Smatra se jednim od najpopularnijih PHP frameworka. Laravel je nastao kao pokušaj da se obezbedi naprednija alternativa [CodeIgniter](https://hr.wikipedia.org/w/index.php?title=CodeIgniter&action=edit&redlink=1)frameworku. Larevel se može preuzeti sa linka:

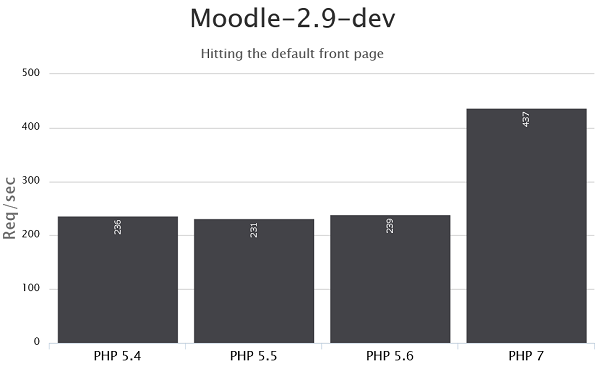
<https://github.com/laravel/laravel/blob/master/readme.md>

**Uvod u PHP7**

Godina 2015. se smatra značajnom godinom za razvoj PHP-a, jer se za kraj ove godine predviđa predstavljanje najnovije verzije ovog jezika, PHP7. Pored novih funkcionalnosti koje PHP7 donosi, biće postignut značajan pomak na polju performansi. Poboljšanje performansi se smatra jednim od najbitnijih razloga za prelazak na novu verziju. Prema zvaničnim, trenutno dostupnim pokazateljima, veliki broj aplikacija koje trenutno rade na verziji PHP 5.6 radiće gotovo dva puta brže na verziji PHP7. U prilog ovoj tvrdnji govore rezultati istraživanja o kojima možete saznati više na: <http://talks.php.net/oz15#/> Ovde ćemo predstaviti samo dva grafikona kojima se prezentuje uporedni prikaz WordPress CMS-a i Moodle LMS-a na različitim verzijama PHP-a:



*1.22. Wordpress* – *PHP7*



*1.23. Moodle* – *PHP7*

Kako u ovom trenutku još uvek ne poznajemo sintaksu PHP jezika, kroz naredne lekcije ćemo se vraćati na promene u sintaksi koje ova verzija PHP-a donosi.

**Koji od sledećih tagova predstavlja validno napisan PHP tag?**

 <\*php \*>

 <php >

 <<<php />>>

 <?php ?>