



SERVERSKE TEHNOLOGIJE

Vežba 3 – Uvod u PHP

Teorijske napomene

PHP (Hypertext Preprocessor) je serverski programski jezik posebno kreiran za izradu dinamičkih veb stranica. Dok HTML prikazuje statički sadržaj, PHP omogućava promenu sadržaja stranice na osnovu korisničkog unosa, podataka iz baze, ili drugih informacija. Kada korisnik poseti stranicu sa PHP kodom, taj kod se izvršava na serveru, a rezultat (uglavnom u obliku HTML-a) se šalje korisnikovom pretraživaču.

XAMPP je integrisani paket koji obuhvata ključne alate potrebne za razvoj veb aplikacija:

- **Apache:** Web server koji omogućava izvršavanje PHP koda lokalno.
- **MySQL:** Sistem za upravljanje bazama podataka gde možete skladištiti podatke.
- **PHP:** Programski jezik za serverski deo veb aplikacija.
- **phpMyAdmin:** Alat za rad sa MySQL bazama, olakšava kreiranje tabela, manipulaciju podacima, i administraciju baza.

XAMPP simulira rad veb servera na vašem računaru i omogućava da efikasno testirate i razvijate aplikacije bez potrebe za eksternim serverom.

Instalacija XAMPP-a

1. Posetite XAMPP sajt i izaberite verziju koja odgovara vašem operativnom sistemu (Windows, Linux, macOS) i kliknite na dugme za preuzimanje.
2. Kada preuzmete instalacioni fajl, pokrenite ga (na Windows-u kliknite dvaput na .exe fajl, a na macOS-u prevucite XAMPP aplikaciju u Applications folder).
3. Kada se pojavi instalacioni wizard, možda će se pojaviti upozorenje o bezbednosti. Kliknite na "Yes" ili "Allow" da biste nastavili instalaciju.
4. Tokom instalacije, dobićete opciju da odaberete komponente koje želite da instalirate.
5. Preporučuje se da ostavite **Apache, MySQL, PHP, i phpMyAdmin** obeležene, jer su oni potrebni za pokretanje PHP aplikacija i rada sa bazama podataka.
6. Na Windows-u, XAMPP se obično instalira u **C:\xampp**, dok se na macOS-u instalira u **Applications**.
7. Kliknite na "Finish" kada se instalacija završi.
8. Po završetku, XAMPP će automatski otvoriti kontrolni panel (XAMPP Control Panel).

Pokretanje XAMPP-a

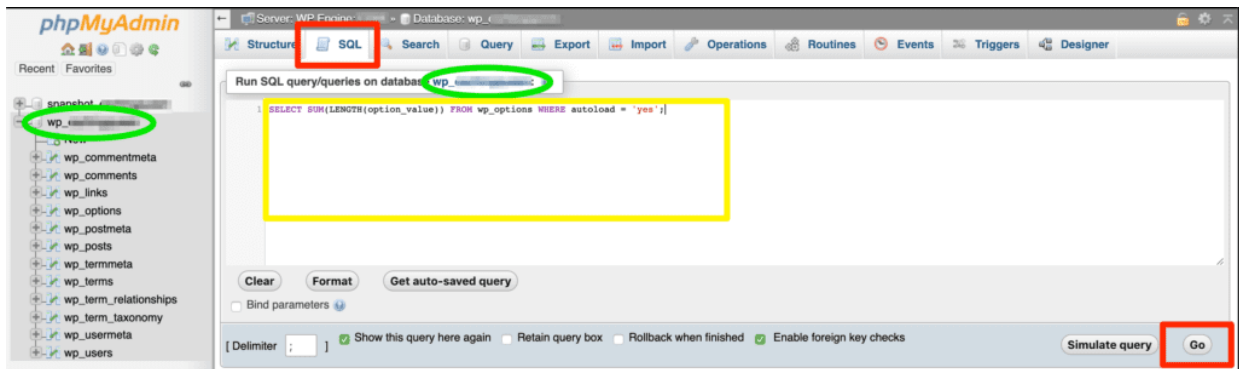
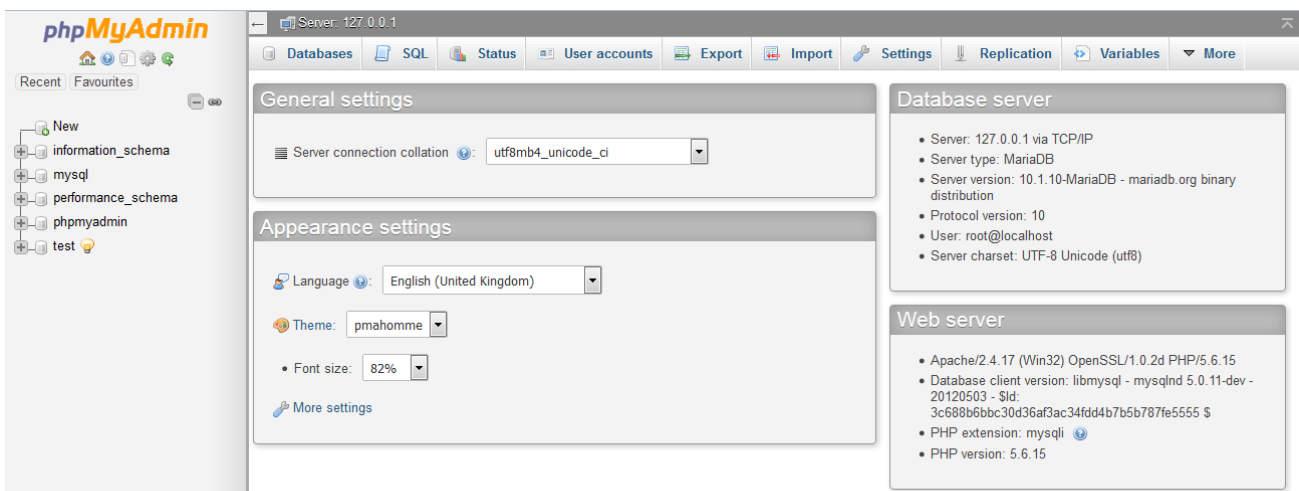
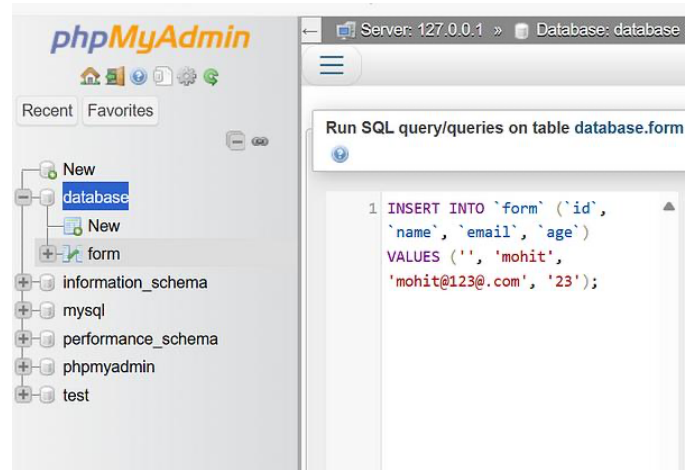
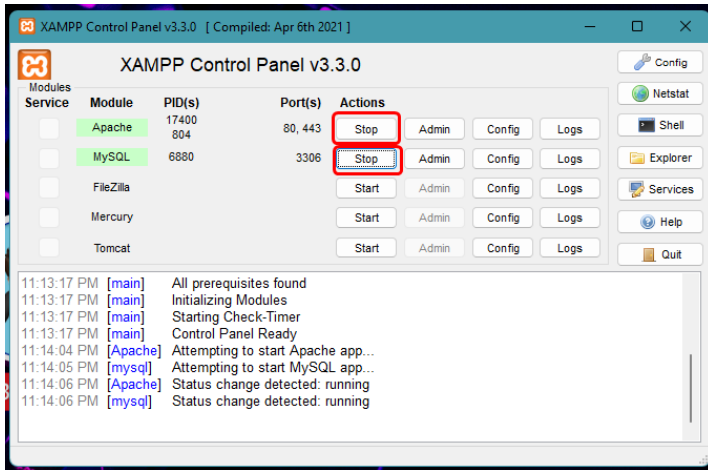
1. U XAMPP **Control Panel-u** videćete listu servisa, uključujući **Apache** i **MySQL**.. Kliknite na dugme **Start** pored Apache i MySQL da biste ih pokrenuli.
2. Ako je sve ispravno instalirano, status Apache i MySQL servisa bi trebao postati zelen i pokrenuti se bez grešaka.
3. Otvorite pretraživač i idite na <http://localhost/>. Trebalo bi da vidite početnu stranicu XAMPP-a, što znači da je Apache uspešno pokrenut.

Ponekad port 80, koji Apache koristi kao podrazumevani, može biti zauzet drugim aplikacijama (kao što su Skype, IIS, ili neki drugi server). Ako primetite grešku prilikom pokretanja Apache-a, verovatno je problem u portu. Evo kako da promenite port:

1. **Otvorite Apache konfiguracioni fajl:**
 - U **XAMPP Control Panel-u** kliknite na **Config** dugme pored Apache servisa.
 - Izaberite opciju **httpd.conf**. Otvoriće se konfiguracioni fajl za Apache u tekstualnom editoru.
2. **Promenite podrazumevani port:**
 - Pronađite liniju koja sadrži komandu: Listen 80
 - Promenite broj porta (na primer, promenite 80 u 8080): Listen 8080
3. **Izmenite port u drugim konfiguracijama:**
 - Pronađite liniju sa ServerName localhost:80 i zamenite je sa ServerName localhost:8080.
 - Sačuvajte promene i zatvorite fajl.
4. **Pokretanje Apache-a na novom portu:**
 - Vratite se u XAMPP Control Panel i ponovo pokrenite Apache.
 - Sada možete pristupiti Apache-u na <http://localhost:8080/>

Takođe, ako imate neki konflikt između verzija MySQL-a, tj. ako već imate instaliran MySQL na računaru od ranije, evo nekoliko opcija kako da ovo izbegnete:

1. **Korišćenje samo MySQL-a iz XAMPP-a:**
 - Isključite ili deinstalirajte staru instalaciju MySQL-a i koristite onu iz XAMPP-a.
 - To može biti lakši pristup ako želite sve na jednom mestu.
2. **Rekonfiguracija porta za MySQL:**
 - Ako želite da koristite obe verzije, možete promeniti port na jednoj od njih.
 - XAMPP MySQL podrazumevano koristi port **3306**. Da biste promenili port za MySQL izvan XAMPP-a, pronađite konfiguracioni fajl i načinite modifikacije.
 - Konfiguracioni fajl za MySQL obično se zove **my.cnf** (na Linux-u i macOS-u) ili **my.ini** (na Windows-u).
 - Otvorite taj fajl u editoru kao što je Notepad (Windows) ili TextEdit (macOS).
 - U ovom fajlu potražite sekciju **[mysqld]**, koja sadrži podešavanja za MySQL server. Unutar ove sekcije, pronađite liniju koja glasi: port=3306 i promenite broj porta, na primer na 3307 ili 3308.
 - Sačuvajte promene, zatvorite fajl i restartujte MySQL server



Detaljan vodič za instalaciju na Windows-u: <https://www.youtube.com/watch?v=G2VEf-8nepc>

Detaljan vodič za instalaciju na MacOS-u: <https://www.youtube.com/watch?v=CCENIDeGwv8>

Praktični primer

Nakon uspešne instalacije i pokretanja neophodnog softvera, kreiranje baze je prvi korak u radu sa podacima. Kreiraćemo bazu **fakultet** koja će sadržati podatke o studentima.

1. U phpMyAdmin-u, kliknite na **Databases** i unesite naziv baze, a zatim kliknite **Create**.
2. Baza fakultet sada postoji i možemo joj dodati tabele.

Tabela je struktura unutar baze gde se podaci skladište u formatu redova i kolona.

1. Kliknite na naziv baze fakultet kako biste joj dodali tabelu za studente.
2. Kliknite na **SQL** tab i unesite SQL kod za kreiranje.
3. Kliknite **Go** nakon toga. Sada imamo tabelu **studenti** sa osnovnim informacijama o studentima.

```
CREATE TABLE studenti (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    ime VARCHAR(50) NOT NULL,  
    prezime VARCHAR(50) NOT NULL  
);
```

- id INT AUTO_INCREMENT PRIMARY KEY: Kolona id se automatski povećava za svaki novi red i služi kao jedinstveni identifikator.
- ime VARCHAR(50) NOT NULL: Kolona za ime studenta, sa maksimalnom dužinom od 50 karaktera.
- prezime VARCHAR(50) NOT NULL: Kolona za prezime studenta, takođe sa maksimalnom dužinom od 50 karaktera.

Svakako, u tabelu treba dodati bar 5 studenata kako bismo imali dobre rezultate upita kasnije. To možete učiniti na sledeći način:

```
INSERT INTO studenti (ime, prezime) VALUES  
( 'Marko', 'Petrović'),  
( 'Ana', 'Jovanović'),  
( 'Ivan', 'Nikolić'),  
( 'Milica', 'Stojanović'),  
( 'Petar', 'Milošević');
```

Povezivanje PHP-a sa bazom omogućava čitanje, zapisivanje, izmenu i brisanje podataka (tzv. CRUD operations). Za ovaj primer, kreiraćemo osnovni PHP kod koji se povezuje sa bazom za fakultet i ispisuje poruku o uspehu ili grešci.

Otvorite folder **htdocs** (nalazi se unutar XAMPP instalacionog direktorijuma) i kreirajte fajl pod imenom **connect.php**. Unesite sledeći kod:

```
<?php  
// Podaci za povezivanje sa bazom podataka  
$servername = "localhost";  
$username = "root";  
$password = ""; // Prazno polje, jer XAMPP podrazumevano nema lozinku za root korisnika  
$dbname = "fakultet";
```

```

// Kreiranje konekcije
$conn = new mysqli($servername, $username, $password, $dbname);

// Provera konekcije
if ($conn->connect_error) {
    die("Konekcija nije uspjela: " . $conn->connect_error);
}
echo "Uspesno povezano sa bazom!";
?>

```

- \$servername, \$username, \$password, \$dbname: Ove promenljive definišu podatke potrebne za povezivanje sa bazom.
- new mysqli(...): Kreira novu konekciju ka bazi.
- connect_error: Proverava da li je došlo do greške u povezivanju. Ako postoji greška, skripta se zaustavlja i prikazuje poruku o grešci.

Sada otvorite pretraživač i idite na <http://localhost/connect.php>. Ako je povezivanje uspešno, prikazaće se poruka "Uspesno povezano sa bazom!".

Sledeći korak je da integrišemo PHP kod na HTML stranici. Kao što je već poznato, PHP i HTML se često koriste zajedno kako bi kreirali dinamičke stranice.

Kreirajte novi fajl index.php u folderu htdocs i unesite sledeći kod:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Pozdrav iz PHP-a</title>
</head>
<body>
    <h1>Dobrodošli na moju PHP stranicu!</h1>
    <?php
        // Ova linija ispisuje tekst koristeći PHP
        echo "<p>Ovo je poruka iz PHP koda.</p>";
    ?>
</body>
</html>

```

Ovo je jednostavan primer gde se samo ispisuje poruka iz PHP-a na HTML stranici. Sada treba povezati i bazu podataka za fakultet kako bismo prikazali, dodali, obrisali ili izmenili studente. U nastavku možete videti kako se oni prikazuju i dodaju, a slično je i za brisanje i izmenu.

Potrebni fajlovi

1. **connect.php** - za povezivanje sa bazom podataka
2. **index.php** - osnovna stranica koja prikazuje podatke
3. **add_student.php** - stranica za dodavanje novih studenata

Ukoliko ste uspešno implementirali connect.php fajl iz prethodnih koraka, možete da pređete na **index.php**. U ovom fajlu imate tri ključne komponente:

- include 'connect.php': Uključuje fajl za povezivanje sa bazom.
- \$sql = "SELECT id, ime, prezime FROM studenti": SQL upit za preuzimanje svih redova iz tabele studenti.
- while(\$row = \$result->fetch_assoc()): Prikazuje podatke iz baze koristeći petlju.

Ceo kod je dat u nastavku:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prikaz Studenata</title>
</head>
<body>
    <h1>Lista Studenata</h1>
    <?php
        // Uključivanje konekcije
        include 'connect.php';
        // SQL upit za čitanje podataka
        $sql = "SELECT id, ime, prezime FROM studenti";
        $result = $conn->query($sql);

        // Provera rezultata i prikazivanje podataka
        if ($result->num_rows > 0) {
            while($row = $result->fetch_assoc()) {
                echo "ID: " . $row["id"] . " - Ime: " . $row["ime"] . " Prezime: " .
                    $row["prezime"] . "<br>";
            }
        } else {
            echo "Nema studenata u bazi.";
        }
    }
```

```

        $conn->close(); // Zatvaranje konekcije
    ?>
    <br>
    <a href="add_student.php">Dodaj Novog Studenta</a>
</body>
</html>

```

Ovi rezultati se mogu ispisati i u HTML tabeli što omogućuje još lepši i detaljniji prikaz. Naredni screenshot pokazuje jedan takav primer. Tabelu možete stilizovati bojama, veličinama slova i ostalim CSS naredbama. Iako je **echo** efikasan za početne primere, u zadatku za samostalni rad se očekuje da na ovakav sličan način uredite stranicu.

```

$sql = "SELECT id, ime, prezime FROM studenti";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // Prikazivanje svakog reda kao HTML red u tabeli
    while ($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo "<td>" . $row["id"] . "</td>";
        echo "<td>" . $row["ime"] . "</td>";
        echo "<td>" . $row["prezime"] . "</td>";
        echo "</tr>";
    }
} else {
    echo "<tr><td colspan='3'>Nema rezultata.</td></tr>";
}

```

Stranica **add_student.php** omogućava unos novih studenata u bazu. Sadrži formu za unos podataka koja poziva isti fajl za unos u bazu.

```

!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Dodaj Studenta</title>
</head>
<body>
    <h1>Dodavanje Novog Studenta</h1>
    <form action="add_student.php" method="post">
        <label for="ime">Ime:</label>
        <input type="text" name="ime" id="ime" required>

```

```

        <br>
        <label for="prezime">Prezime:</label>
        <input type="text" name="prezime" id="prezime" required>
        <br>
        <input type="submit" value="Dodaj Studenta">
    </form>

    <?php
        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            include 'connect.php'; // Uključivanje konekcije
            // Preuzimanje podataka iz forme
            $ime = $_POST['ime'];
            $prezime = $_POST['prezime'];

            // SQL upit za umetanje podataka
            $sql = "INSERT INTO studenti (ime, prezime) VALUES ('$ime', '$prezime)";

            if ($conn->query($sql) === TRUE) {
                echo "Novi student je uspešno dodat!";
            } else {
                echo "Greška: " . $sql . "<br>" . $conn->error;
            }

            $conn->close(); // Zatvaranje konekcije
        }
    ?>
    <br>
    <a href="index.php">Nazad na Listu Studenata</a>
</body>
</html>

```

- `$_SERVER["REQUEST_METHOD"] == "POST"`: Proverava da li je forma poslata metodom POST.
- `$ime = $_POST['ime']`: Prikuplja unos iz forme.
- `$sql = "INSERT INTO studenti (ime, prezime) VALUES ('$ime', '$prezime)";`: SQL upit za dodavanje novog studenta.
- `if ($conn->query($sql) === TRUE)`: Proverava uspešnost umetanja podataka u bazu.

Nakon svih ovih koraka, možete konačno pokrenuti i testirati svoj primer. Ukoliko ste u htdocs direktno stavljali fajlove, preporuka je da ih prebacite u jedan poseban folder, jer ćemo svakako imati još dosta vežbanja u budućnosti. Dakle, u htdocs otvorite folder „primer“ i tu ubacite ove kodove, a onda ćemo za sledeće vežbanje otvoriti novi folder. Dalje možete ispratiti ove korake:

- Pokrenite Apache i MySQL iz XAMPP Control Panela.
- Otvorite pretraživač i posetite <http://localhost/primer/index.php> za prikaz studenata.
- Idite na http://localhost/primer/add_student.php da biste dodali novog studenta.

Zadatak za samostalni rad

U ovom zadatku je neophodno kreirati bazu **prodavnica** i tabelu **proizvod**. Svaki proizvod treba da ima ID (može automatski broj), naziv, kategoriju (elektronika, garderoba, hrana...) i cenu (u dinarima ili evrima). Kategoriju možete realizovati i preko stranih ključeva i tabele **kategorija**, što je i preporučljivo za efikasniju implementaciju.

Nakon toga, potrebno je dodati 5-10 proizvoda u tabelu za proizvode i prikazati ih na glavnoj stranici (index.php). Konekciju sa bazom svakako smestite u poseban fajl (connect.php).

Što se dizajnerskih komponenti tiče, obavezno neka vaša stranica ima stilizovani header sa nazivom ili logotipom prodavnice (budite kreativni), footer sa vašim podacima (ime, prezime, indeks ili boja vašeg tima), podaci iz baze neka se prikazuju u urednoj tabeli, a omogućite i redirekciju na stranicu za dodavanje novog proizvoda (možete staviti dugme ili link negde na glavnoj stranici). Sav CSS stil treba biti u posebnom fajlu (style.css).

Kada korisnik klikne na dugme ili link za dodavanje novog proizvoda, treba da se otvori nova stranica sa formom za dodavanje proizvoda. Korisnik treba da ukuca naziv, kategoriju i cenu proizvoda. Kategorija može biti realizovana i preko padajuće liste (select komponenta). Evo jednog primera, gde iz HTML liste kurseva korisnik uzima jedan i smešta ga u PHP promenljivu:

```
<select name="course" id="course">
    <option value="math">Mathematics</option>
    <option value="science">Science</option>
    <option value="history">History</option>
</select>

// Provera da li je forma poslata putem POST metode i da li je kurs odabran
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['course'])) {
    // Uzimanje vrednosti koju je korisnik izabrao iz select menija
    $selected_course = $_POST['course'];
    ...
}
```

Ostatak dodavanja novog proizvoda se realizuje isto kao i sa studentima na primeru što je pokazano. Možete ubaciti još neki korak validacije da se uverite da je sve u redu sa unetim podacima, kao na primer:

```
// Provera da li je forma poslata
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Uzimanje podataka iz forme
    $naziv = trim($_POST['naziv']);
    $cena = trim($_POST['cena']);
    $kategorija = trim($_POST['kategorija']);
```

```
// Server-side validacija: Provera da li su svi podaci uneseni
if (empty($naziv) || empty($cena) || empty($kategorija)) {
    echo "Sva polja moraju biti popunjena!";
} else {
    // Provera da li je cena validan broj
    if (!is_numeric($cena) || $cena <= 0) {
        echo "Cena mora biti broj veći od 0!";
    } else {
        // Sve je u redu, tako da sledi dalji rad sa bazom
        ...
    }
}
```

Na kraju testirajte obe stranice i proverite da li sve funkcioniše ispravno i sinhronizovano. Za bilo kakve greške možete odraditi inspekciju u pretraživaču ili ispisati dodatne echo naredbe.

Korisni resursi

1. Jednostavna MySQL vežbanja:

https://www.w3schools.com/mysql/mysql_exercises.asp

2. Jednostavna PHP vežbanja:

https://www.w3schools.com/php/php_exercises.asp

3. Odličan YouTube kurs koji se fokusira na PHP samo:

https://www.youtube.com/watch?v=OK_JCtrrv-c

4. Odličan YouTube kurs koji obuhvata sve:

https://www.youtube.com/watch?v=4uSakWW_dMI

5. Veliki broj besplatnih (i plaćenih) PHP projekata:

<https://phpgurukul.com/php-projects-free-downloads/>