



VILNIUSCODINGSCHOOL

PHP 5

Kontaktinė forma



Bendraukime

Susisiekite su mumis šiandien!

Siųsti

HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="PHP forma">
    <title>Forma</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <header>
      <div class="container">
        
      </div>
    </header>
    <main>
      <div class="container">
        <form id="contact" action="forma.php" method="post">
          <h3>Bendraukime</h3>
          <h4>Susisiekite su mumis šiandien!</h4>
          <p><input type="text" name="vardas" placeholder="Jūsų vardas" required autofocus></p>
          <p><input type="email" name="email" placeholder="Jūsų el. pašto adresas" required></p>
          <p><textarea placeholder="Jūsų žinutė..." name="message" required></textarea></p>
          <p><button name="submit" type="submit" id="contact-submit">Siųsti</button></p>
        </form>
      </div>
    </main>
    <footer>
      <div class="container">
        <p class="copyright">&copy; 2018. Visos teisės saugomos.</p>
      </div>
    </footer>
  </body>
</html>
```

Mūsų kontaktinė forma



- HTML, CSS ir logotipas įkelti į Google Classroom.

- Atkreipkite dėmesį, kad turime tris kertinius duomenis, kuriuos forma kaupia: vardas, email, message.

- Kaip pasieksime šiuos duomenis???

A screenshot of a web browser's developer tools, specifically the 'Network' tab. The 'Headers' sub-tab is selected. The 'General' section shows the following details:

- Request URL: http://localhost:8888/php/diena05/index.php
- Request Method: POST
- Status Code: 200 OK (with a green status icon)
- Remote Address: [::1]:8888
- Referrer Policy: no-referrer-when-downgrade

Below this, there are expandable sections for 'Response Headers (7)', 'Request Headers (13)', and 'Form Data'. The 'Form Data' section is expanded, showing the following data:

- vardas: Andrius
- email: andrius@vilniuscoding.lt
- message: Labas
- submit:

Mūsų kontaktinė forma



- Failo viršuje pridedame:

```
<?php
    // spausdiname visus duomenis iš POST
    var_dump($_POST);
?>
```

- Tai atspausdins visus POST duomenis pačiame puslapio viršuje ir matysime rezultatą naršyklėje

```
array(4) ( ["vardas"]=> string(7) "Andrius" ["email"]=> string(24) "andrius@vilniuscoding.lt" ["message"]=> string(5) "Labas" ["submit"]=> string(0) "" )
```

- Taigi, mūsų duomenys yra masyve! O masyvo duomenis galime išvesti.

```
<?php
    echo $_POST['vardas'] . '<br>' . $_POST['email'] . '<br>' . $_POST['message'];
?>
```

Mūsų kontaktinė forma



- Šiek tiek kosmetinių pakeitimų ir laikinai spausdiname formos duomenis pačiame puslapyje.

```
<?php
    $vardas = $_POST['vardas'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    echo "<p>Vardas: $vardas</p>";
    echo "<p>El. paštas: $email</p>";
    echo "<p>Žinutė: $message</p>";
?>
```

Vardas: Darius

El. paštas: darius@vilniuscoding.lt

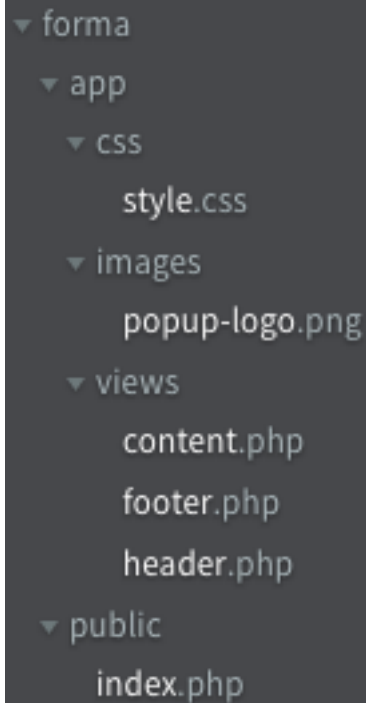
Žinutė: Geriau

Tvarkome failų struktūrą



- Mūsų index failas plečiasi.
Mums reikia daugiau struktūros.
Sukurkime tokį direktorių medį:

- **css**: style.css
- **images**: popup-logo.png
- **public**: index.php
- **views**: header.php, content.php, footer.php



Tvarkome failų nuorodas



- Koreguojame savo index.php:

- css nuoroda: `<link rel="stylesheet" href="../app/css/style.css">`

- logotipo nuoroda: ``

- `<header>` dalį perkeliame į header.php, `<main>` dalį perkeliame į content.php, `<footer>` dalį perkeliame į footer.php

- vietoje jų:

```
<?php
    include('../app/views/header.php');
    include('../app/views/content.php');
    include('../app/views/footer.php');
?>
```


Kitų failų įtraukimas



- **Privalumai:**

- Failus apjungia serveris
- Gali būti PHP, HTML, tekstas
- Daugkartinis kodo panaudojimas
- Bendri puslapio elementai atnaujinami keičiant tik vieną failą

include - serveris bando apdoroti kodą net jei failo neranda

include_once - leidžia prijungti failą tik vieną kartą

require - stabdo kodo apdorojimą, jei failo neranda

require_once - leidžia prijungti failą tik vieną kartą

Tvarkome failų nuorodas



- Išimkime iš index.php kodą, kurį naudojome formos duomenų pasiekimui.
- Sukurkime naują direktoriją **app** viduje - **src**, kurios viduje sukurkime failą **app.php**
- Į app.php įkelkime:

```
<?php
    $vardas = $_POST['vardas'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    echo "<p>Vardas: $vardas</p>";
    echo "<p>El. paštas: $email</p>";
    echo "<p>Žinutė: $message</p>";
```

- Dėmesio:

- gryname php faile
dėti ?> pačioje
pabaigoje neprivaloma

Skirtumas tarp include ir require



- index.php failo viršuje turime reikalauti, kad būtų pakrautas app.php

```
<?php  
    require __DIR__ . '/../app/src/app.php';  
?>
```

- Tiek **include**, tiek **require** funkcijos įkelia kitus failus. Tačiau kai naudojame **include**, jei mūsų nuoroda neteisinga, gausime tik **warning**, bet failas vis dar veiks. Tuo tarpu, jei naudosisime **require**, o mūsų nuoroda bus neteisinga, gausime **fatal error**, ir mūsų puslapis visai neveiks.

Tvarkome app.php



- Iki šiol mūsų formos įvestus duomenis dėjome tiesiai į index.php failą. Žinoma, tai blogas sprendimas. Norime, kad mūsų gauti duomenys pasiektų duomenų bazę.
- Tačiau turime užtikrinti, kad negausime tuščių duomenų, ir kad gauti duomenys yra teisingo formato (pvz. el. paštas).
- Dėlei saugumo ir formatavimo turime pašalinti bet kokį HTML ar kitokį kodą.
- Jei esama neatitikimų, turime gražinti vartotoją prie formos pildymo.

Tvarkome app.php



- Įsitikinkime, kad duomenys buvo užpildyti:

```
<?php
    $vardas = $_POST['vardas'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    if(!empty($vardas) && !empty($email) && !empty($message)) {
        echo "<p>Vardas: $vardas</p>";
        echo "<p>El. paštas: $email</p>";
        echo "<p>Žinutė: $message</p>";
    }
```

Tvarkome app.php



- Pašalinkime bet kokius prieš ir po sekančius tarpus su **trim** funkcija:

```
<?php
    $vardas = trim($_POST['vardas']);
    $email = trim($_POST['email']);
    $message = trim($_POST['message']);
```

- Jei vartotojas bandytų rašyti su HTML kodu mūsų formos viduje, mums reikia galimybės kopijuoti kodo žymes kaip įprastą tekstą:

```
if(!empty($vardas) && !empty($email) && !empty($message)) {
    echo "<p>Vardas: $vardas</p>";
    echo "<p>El. paštas: $email</p>";
    echo '<p>Žinutė: ' . htmlspecialchars($message) . '</p>';
}
```

El. pašto validavimas



- Kodėl filtruojame ir validuojame? Mūsų duomenų bazėje esantys duomenys turi būti patikrinti ir teisingi.

```
if(!empty($vardas) && !empty($email) && !empty($message)) {  
    echo "<p>Vardas: $vardas</p>";  
    if(filter_var($email, FILTER_VALIDATE_EMAIL)) {  
        echo "<p>El. paštas: $email</p>";  
    }  
    echo "<p>Žinutė: ' . htmlspecialchars($message) . '</p>';  
}
```

- `filter_var` patikrina mūsų kintamąjį pagal PHP filtravimo konstantą.

Formos siuntimas (modifikuojame app.php)



```
<?php
    $vardas = trim($_POST['vardas']);
    $email = trim($_POST['email']);
    $message = trim($_POST['message']);

    if(!empty($vardas) && !empty($email) && !empty($message)) {
        if(filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $from = "$email";
            $to = "andrius@vilniuscoding.lt";
            $subject = "Nauja žinutė";
            $autorius = 'Nuo: ' . $vardas . ', ' . $email;
            $zinute = htmlspecialchars($message);
            mail($to, $subject, $autorius, $zinute, $from);
            echo "<script>alert('Dėkojame. Jūsų žinutė gauta. Netrukus susisieksime.');
```


SQL



- **SQL yra skirta valdyti duomenų bazėms. SQL programavimo kalba leidžia mums sukurti, atnaujinti ir ištrinti duomenis iš duomenų bazės.**
- **Duomenų bazės lentelės turi eilutes ir stulpelius, lygiai taip pat kaip Excel lentelės.**
- **Svarbu atsiminti, jog gerai parašyta duomenų bazės lentelė visuomet turi turėti savo automatiškai generuojamą raktą, pvz. ID.**
- **MySQL yra programinė įranga, skirta darbui su duomenų bazėmis, o SQL yra programavimo kalba, kurios dėka atliekame komandas/funkcijas pasirinktoje programinėje įrangoje.**

Naujos duomenų bazės sukūrimas



- Įsitikiname, kad mūsų Apache ir MySQL yra įjungti ir atsidarome naršyklėje: localhost/phpmyadmin
- Du būdai sukurti duomenų bazę:
 - Databases tab: įvedame **Database name** ir **Create** mygtukas
 - SQL tab:

Run SQL query/queries on server "localhost": ?

```
1 CREATE DATABASE forma;
```

Clear Format Get auto-saved query

☐ Bind parameters ?

[Delimiter ;] ☒ Show this query here again ☐ Retain query box
☐ Rollback when finished ☒ Enable foreign key checks

Go

Lentelės sukūrimas



- Kol kas mūsų duomenų bazė tuščia. Mums reikalinga lentelė, kurioje galėsime talpinti mūsų vartotojų duomenis. Paspaudžiame ant **forma** kairėje esančiame duomenų bazių medyje ir SQL tab rašome:

Run SQL query/queries on database **forma**: ?

```
1 CREATE TABLE klientai (  
2     id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
3     vardas TEXT NOT NULL,  
4     email VARCHAR(320) NOT NULL,  
5     message TEXT NOT NULL  
6     ) DEFAULT CHARACTER SET utf8;
```

☐ Bind parameters ?

[Delimiter ;] ☒ Show this query here again ☐ Retain query box

☐ Rollback when finished ☒ Enable foreign key checks

Duomenys testavimui



- Kad įsitikintume, jog mūsų duomenų bazė veikia, įvedame pirmuosius duomenis:

```
1 INSERT INTO klientai SET
2 vardas = "Andrius",
3 email = "andrius@vilniuscoding.lt",
4 message = "testas";
```

- Tikriname lentelėje esančius duomenis:

```
1 SELECT * FROM klientai
```

+ Options

				id	vardas	email	message
<input type="checkbox"/>	Edit	Copy	Delete	1	Andrius	andrius@vilniuscoding.lt	testas
	<input type="checkbox"/> Check all	With selected:		Edit	Copy	Delete	Export

PHP bendravimas su duomenų baze



- src direktorijoje, sukuriame naują failą: **db.php**
- app.php faile pridedame jungtį: **include('db.php');** // failo apačioje
- db.php faile:

```
<?php
define("DB_SERVER", "localhost");
define("DB_USER", "root");
define("DB_PASSWORD", "root");
define("DB_NAME", "forma");

$mysqli = new mysqli(DB_SERVER, DB_USER, DB_PASSWORD, DB_NAME);
if($mysqli->connect_error) {
    echo "Atsiprašome, bet svetainė susidūrė su problema.\n";
    echo 'Klaida: ' . $mysqli->connect_error . '\n';
    exit();
}

mysqli_query($mysqli, "INSERT INTO klientai (vardas, email, message)
VALUES('$_POST[vardas]', '$_POST[email]', '$_POST[message]')");
```

PHP bendravimas su duomenų baze



- Turime patikrinti ar galime mūsų formos puslapyje išvesti duomenų bazės duomenis, todėl papildome db.php:

// jei spausdiname tik vieną eilutę

```
$sql = "SELECT * FROM klientai";  
$result = $mysqli->query($sql);  
$array = $result->fetch_assoc();  
var_dump($array);
```

// jei spausdiname visą DB lentelę

```
$sql = "SELECT * FROM klientai";  
$result = $mysqli->query($sql);  
while($array = $result->fetch_assoc()) {  
    var_dump($array);  
}
```

```
array(4) ( ["id"]=> string(1) "1" ["vardas"]=> string(7) "Andrius" ["email"]=> string(24) "andrius@vilniuscoding.lt" ["message"]=> string(6) "testas" )
```












PHP bendravimas su duomenų baze



- Kadangi patikrinome ir viskas veikia, koreguojame db.php - ką tik pridėtą kodo dalį pakeičiame:

```
mysqli_query($mysqli, "INSERT INTO klientai (vardas, email, message)  
VALUES('$_POST[vardas]', '$_POST[email]', '$_POST[message]')");
```

+ Options

 				id	vardas	email	message
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	Andrius	andrius@vilniuscoding.lt	testas
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	Darius	darius@vilniuscoding.lt	testas2
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	Saulius	saulius@vilniuscoding.lt	testas3

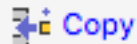


☐ Check all

With selected:



Edit



Copy



Delete



Export

Valio!



- Mes ką tik sugebėjome vartotojo kontaktinėje formoje įvestus duomenis įkelti į duomenų bazę (kurią išmokome sukurti)!!!

