ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Соловьева Н.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О КОНТРОЛЬНОЙ РАБОТЕ №2 |
| **по дисциплине: Web-Технологии** |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 4134К |  |  |  | Самарин Д. В. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2024

Оглавление

[Цель работы: 3](#_Toc165923213)

[Применяемые технологии 3](#_Toc165923215)

[Обзор на готовые страницы: 3](#_Toc165923216)

[Описание выполненных пунктов 4](#_Toc165923217)

[Скриншоты web-страниц с подписью и номерами рисунков 5](#_Toc165923218)

[Первая страница PHP: 8](#_Toc165923226)

[Вторая страница HTML: 13](#_Toc165923227)

Цель работы: знакомство с языком разметки HTML, работа с основными тегами

## Вариант задания № 4:

|  |  |
| --- | --- |
| № | Описание варианта |
| 8 | Написать программу для отправки письма на языке php. |
| 26 | Объясните назначение и дайте пример использования свойства childElementCount |

## Применяемые технологии

Редактор кода VSCode;

PHP/COMPOSER/PHPMALIER

Браузер Chrome;

## Обзор на готовые страницы:

1) Первая страница (PHP):

* [Рисунок 1;](#_Рисунок_1(HTML_4))
* [Рисунок 2;](#_Рисунок_2(HTML_4))
* [Рисунок 2.1;](#_Рисунок_2.1_(Результат)
* [Рисунок 2.2;](#_Рисунок_2.2_(Результат)
* [Cсылка на листинг;](#_Первая_страница_HTML)

2) Вторая страница (HTML):

* [Рисунок 3;](#_Рисунок_3_(HTML)
* [Ссылка на листинг;](#_Вторая_страница_HTML)

## 

## Описание выполненных пунктов

В ходе работы были реализованы веб страницы согласно вариантам 8 и 26

С помощью PHP была написана страница, которая позволяет отправлять письма на личную почту (использую smtp сервер и локальное хранилище компьютера).

WEB страница была стилизована под почтовый сервис.

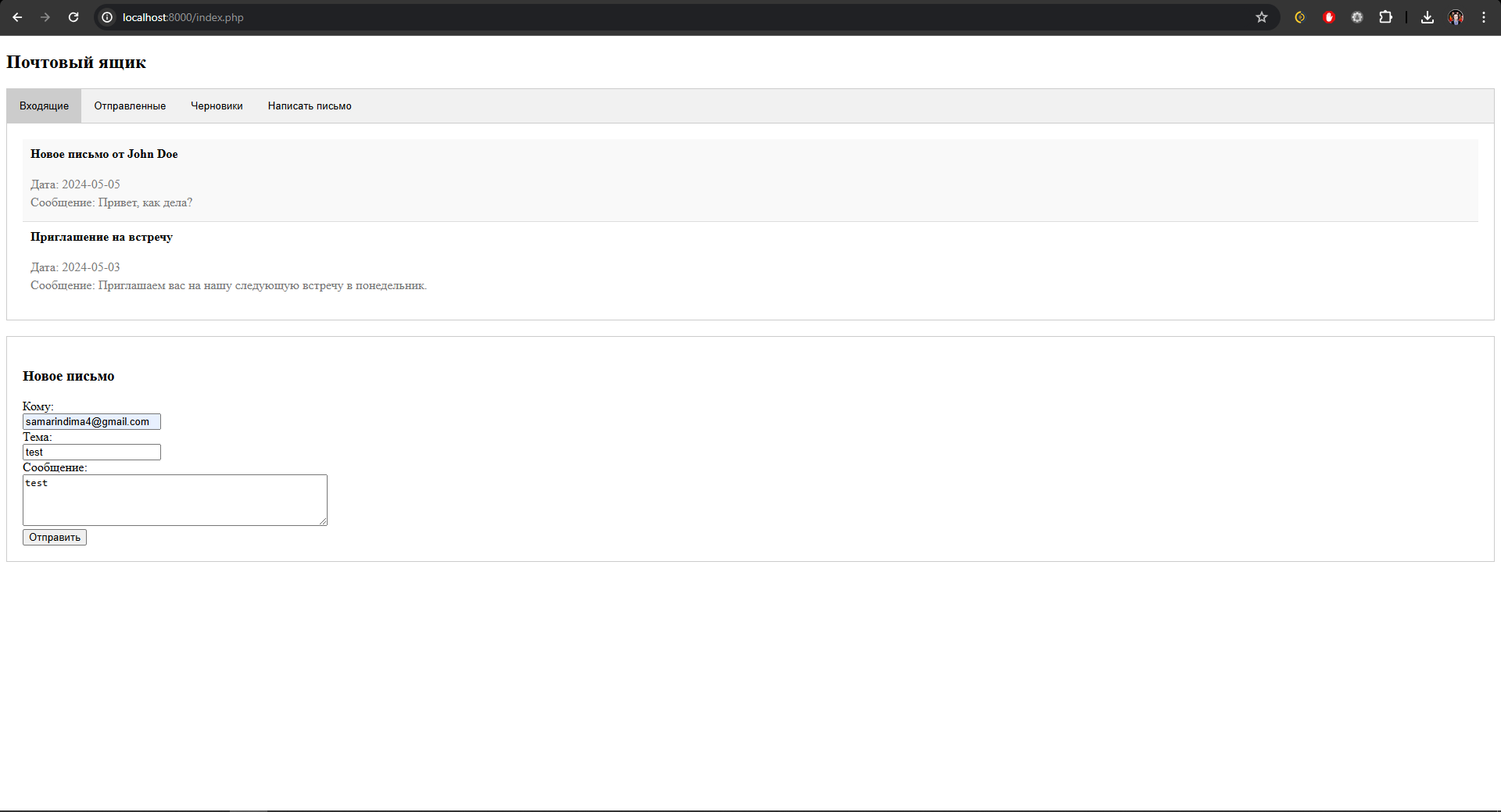
SMTP использует данные логина/пароля аккаунта сервиса Google. При этом пароль является сгенерированным паролем старого формата (для очень старых устройств).

Вторая WEB страница является наглядным примером использования свойства childElementCount.

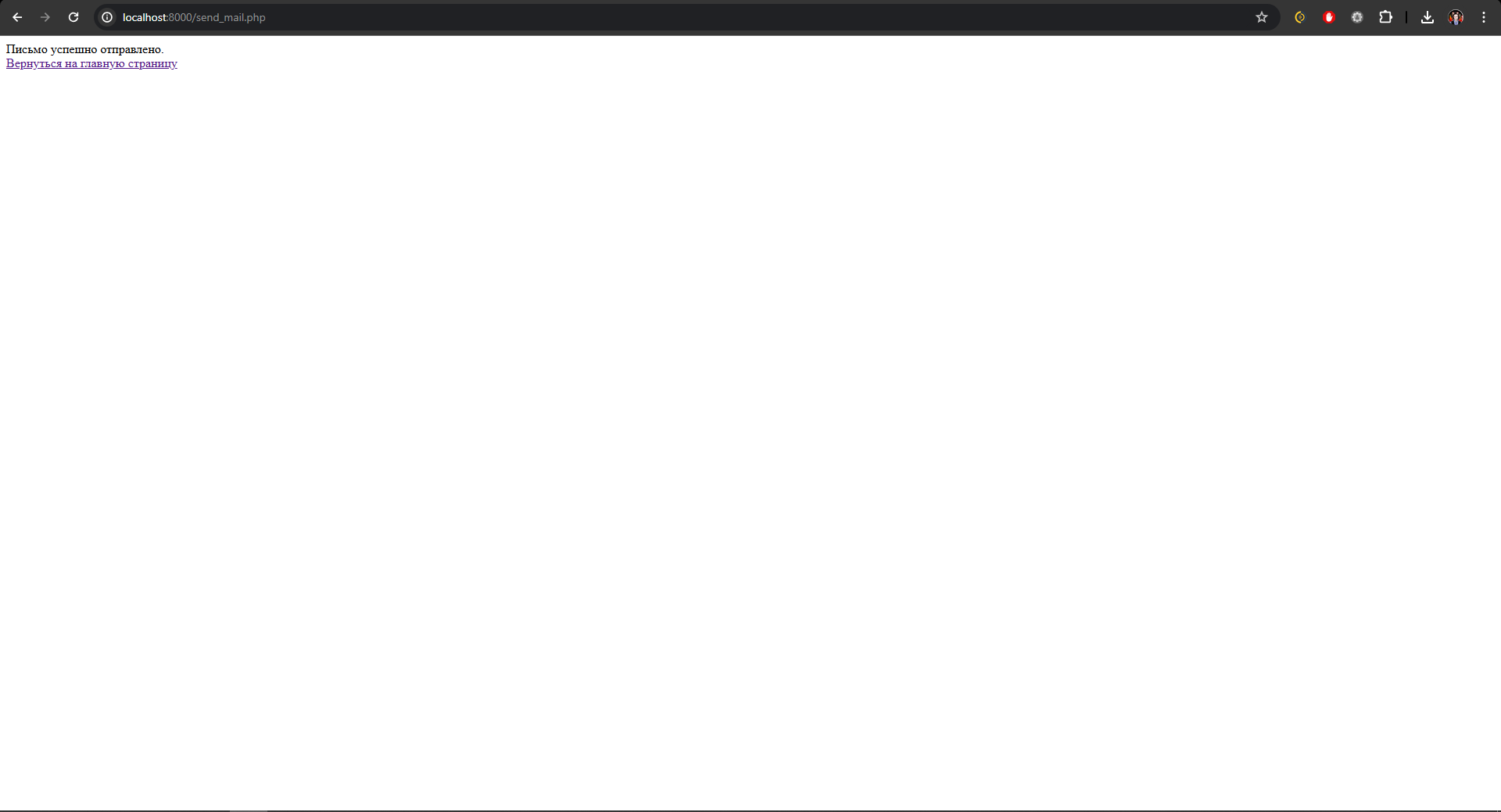
Свойство childElementCount в JavaScript используется для получения количества дочерних элементов у родительского элемента. Это свойство особенно удобно, когда вам нужно быстро определить количество элементов внутри другого элемента без необходимости явно перебирать или считать их.

Исходя из примера, у нас есть <div>, который имеет три дочерних элемента <p>. Мы используем свойство childElementCount, чтобы получить количество дочерних элементов этого <div>. Результат будет выведен в консоль, показывая количество дочерних элементов (в данном случае, это будет 3). Это может быть полезно, например, для автоматического вычисления количества элементов в списке или сетке.

## Скриншоты web-страниц с подписью и номерами рисунков



## Рисунок 1(PHP)



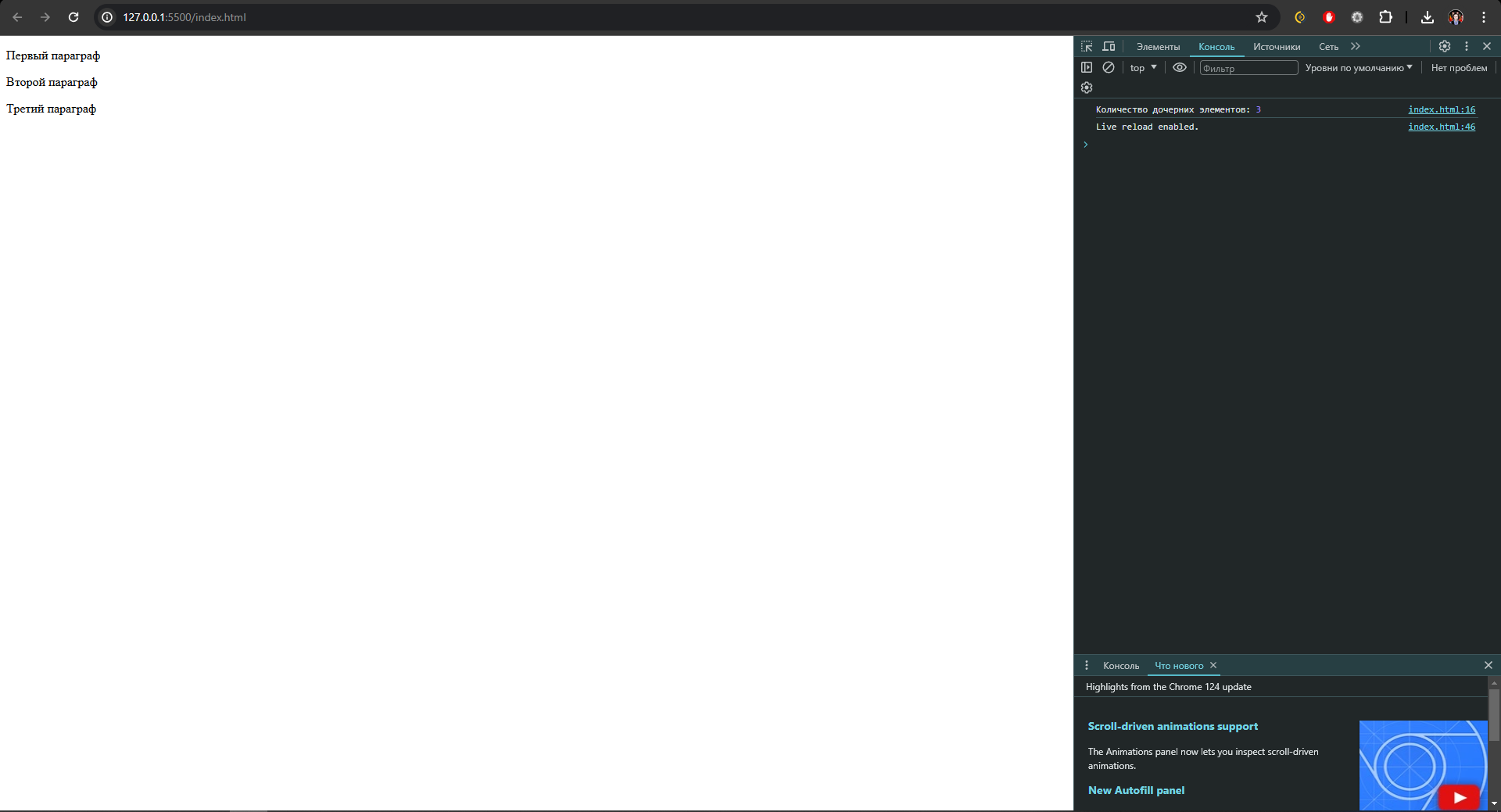
## Рисунок 2(PHP)

# 

# Рисунок 2.1 (Результат SMTP)

## 

## Рисунок 2.2 (Результат локального письма)



## Рисунок 3 (HTML)

Листинг лабораторной работы:

## Первая страница PHP:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>Почтовый ящик</title>  <style>  /\* Общие стили \*/  .tabcontent {  display: none;  padding: 20px;  border: 1px solid #ccc;  border-top: none;  }  .tab {  overflow: hidden;  border: 1px solid #ccc;  background-color: #f1f1f1;  }  .tab button {  background-color: inherit;  float: left;  border: none;  outline: none;  cursor: pointer;  padding: 14px 16px;  transition: 0.3s;  }  .tab button:hover {  background-color: #ddd;  }  .tab button.active {  background-color: #ccc;  }  #compose {  border: 1px solid #ccc;  padding: 20px;  margin-top: 20px;  }  /\* Стили для вкладки "Входящие" \*/  .mail-item {  border-bottom: 1px solid #ddd;  padding: 10px;  overflow: hidden;  }  .mail-item:hover {  background-color: #f9f9f9;  }  .mail-item:last-child {  border-bottom: none;  }  .mail-item h4 {  margin-top: 0;  }  .mail-item p {  margin: 5px 0;  color: #777;  }  </style>  </head>  <body>  <h2>Почтовый ящик</h2>  <!-- Вкладки -->  <div class="tab">  <button class="tablinks" onclick="openTab(event, 'Входящие')">Входящие</button>  <button class="tablinks" onclick="openTab(event, 'Отправленные')">Отправленные</button>  <button class="tablinks" onclick="openTab(event, 'Черновики')">Черновики</button>  <button onclick="openCompose()">Написать письмо</button>  </div>  <!-- Содержимое вкладок -->  <div id="Входящие" class="tabcontent">  <div class="mail-item">  <h4>Новое письмо от John Doe</h4>  <p>Дата: 2024-05-05</p>  <p>Сообщение: Привет, как дела?</p>  </div>  <div class="mail-item">  <h4>Приглашение на встречу</h4>  <p>Дата: 2024-05-03</p>  <p>Сообщение: Приглашаем вас на нашу следующую встречу в понедельник.</p>  </div>  <!-- Здесь можно добавить код для отображения входящих писем -->  </div>  <div id="Отправленные" class="tabcontent">  <div class="mail-item">  <h4>Тема письма 1</h4>  <p>Кому: example@example.com</p>  <p>Дата: 2024-05-01</p>  <p>Сообщение: ....</p>  </div>  <div class="mail-item">  <h4>Тема письма 2</h4>  <p>Кому: test@test.com</p>  <p>Дата: 2024-05-02</p>  <p>Сообщение: ....</p>  </div>  <!-- Здесь можно добавить код для отображения отправленных писем -->  </div>  <div id="Черновики" class="tabcontent">  <div class="mail-item">  <h4>Черновик 1</h4>  <p>Текст черновика 1...</p>  </div>  <div class="mail-item">  <h4>Черновик 2</h4>  <p>Текст черновика 2...</p>  </div>  <!-- Здесь можно добавить код для отображения черновиков -->  </div>  <!-- Форма для написания письма -->  <div id="compose" style="display:none;">  <h3>Новое письмо</h3>  <form action="send\_mail.php" method="post">  <label for="to">Кому:</label><br>  <input type="text" id="to" name="to"><br>  <label for="subject">Тема:</label><br>  <input type="text" id="subject" name="subject"><br>  <label for="message">Сообщение:</label><br>  <textarea id="message" name="message" rows="4" cols="50"></textarea><br>  <input type="submit" value="Отправить">  </form>  </div>  <script>  // Функция для открытия вкладок  function openTab(evt, tabName) {  var i, tabcontent, tablinks;  tabcontent = document.getElementsByClassName("tabcontent");  for (i = 0; i < tabcontent.length; i++) {  tabcontent[i].style.display = "none";  }  tablinks = document.getElementsByClassName("tablinks");  for (i = 0; i < tablinks.length; i++) {  tablinks[i].className = tablinks[i].className.replace(" active", "");  }  document.getElementById(tabName).style.display = "block";  evt.currentTarget.className += " active";  }  // Функция для открытия формы написания письма  function openCompose() {  var compose = document.getElementById("compose");  if (compose.style.display === "none") {  compose.style.display = "block";  } else {  compose.style.display = "none";  }  }  // По умолчанию открыть вкладку "Входящие"  document.getElementById("Входящие").style.display = "block";  </script>  </body>  </html> |

Способы реализации отправки писем (локальное хранилище/smtp сервер):

|  |
| --- |
| <?php  // Проверяем, была ли отправлена форма  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  // Получаем данные из формы  $to = $\_POST['to'];  $subject = $\_POST['subject'];  $message = $\_POST['message'];  // Путь к файлу, куда будет сохранено сообщение  $filePath = 'C:/Users/Dmitry/Desktop/6 сем/WEBKR/message.txt'; // Указать полный путь к файлу  // Формируем содержимое файла (в текстовом формате)  $fileContent = "To: " . $to . "\n";  $fileContent .= "Subject: " . $subject . "\n";  $fileContent .= "\n";  $fileContent .= $message;  // Записываем содержимое в файл  file\_put\_contents($filePath, $fileContent);  // Выводим сообщение об успешной отправке письма и кнопку для возврата на главную страницу  echo "Письмо успешно отправлено.";  echo "<br>";  echo '<a href="index.php">Вернуться на главную страницу</a>';  } else {  // Если форма не была отправлена, перенаправляем обратно на страницу  header("Location: index.html");  }  ?>  SMTP Сервер  <?php  // Проверяем, была ли отправлена форма  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  // Получаем данные из формы  $to = $\_POST['to'];  $subject = $\_POST['subject'];  $message = $\_POST['message'];  error\_reporting(E\_ALL);  ini\_set('display\_errors', 1);  // Настройки SMTP  $smtpHost = 'smtp.gmail.com';  $smtpPort = 587;  $smtpUsername = 'samarindima4@gmail.com';  $smtpPassword = 'тут должен быть пароль';  $smtpTimeout = 30;  // Формируем заголовки для письма  $headers = [  'MIME-Version: 1.0',  'Content-type: text/html; charset=utf-8',  'From: ' . $smtpUsername,  'Reply-To: ' . $smtpUsername,  'X-Mailer: PHP/' . phpversion()  ];  // Формируем строку параметров для отправки письма через SMTP  $params = [  'ssl' => [  'verify\_peer' => false,  'verify\_peer\_name' => false,  'allow\_self\_signed' => true  ],  'socket' => [  'tcp\_nodelay' => true  ]  ];  $smtpAddress = sprintf('tcp://%s:%d', $smtpHost, $smtpPort);  $context = stream\_context\_create($params);  // Устанавливаем соединение с SMTP-сервером  $smtpConnection = stream\_socket\_client($smtpAddress, $errno, $errstr, $smtpTimeout, STREAM\_CLIENT\_CONNECT, $context);  if ($smtpConnection) {  // Авторизуемся на сервере  fputs($smtpConnection, 'EHLO ' . $smtpHost . "\r\n");  fputs($smtpConnection, 'STARTTLS' . "\r\n");  fputs($smtpConnection, 'AUTH LOGIN' . "\r\n");  fputs($smtpConnection, base64\_encode($smtpUsername) . "\r\n");  fputs($smtpConnection, base64\_encode($smtpPassword) . "\r\n");  // Отправляем данные письма  fputs($smtpConnection, 'MAIL FROM: <' . $smtpUsername . ">\r\n");  fputs($smtpConnection, 'RCPT TO: <' . $to . ">\r\n");  fputs($smtpConnection, 'DATA' . "\r\n");  fputs($smtpConnection, 'Subject: ' . $subject . "\r\n");  foreach ($headers as $header) {  fputs($smtpConnection, $header . "\r\n");  }  fputs($smtpConnection, "\r\n");  fputs($smtpConnection, $message . "\r\n.\r\n");  // Завершаем сеанс  fputs($smtpConnection, 'QUIT' . "\r\n");  // Закрываем соединение  fclose($smtpConnection);  echo "Письмо успешно отправлено.";  } else {  echo "Ошибка при отправке письма.";  }  } else {  // Если форма не была отправлена, перенаправляем обратно на страницу  header("Location: index.php");  }  ?> |

## Вторая страница HTML:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>Пример использования свойства childElementCount</title>  </head>  <body>  <div id="parentElement">  <p>Первый параграф</p>  <p>Второй параграф</p>  <p>Третий параграф</p>  </div>  <script>  var parent = document.getElementById('parentElement');  var childCount = parent.childElementCount;  console.log('Количество дочерних элементов:', childCount);  </script>  </body>  </html> |