ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

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

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6  Язык HTML. Приемы верстки |
| **по дисциплине: Web-Технологии** |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

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

Санкт-Петербург

2023

Оглавление

[Цель работы: 3](#_Toc167854813)

[Вариант задания: 3](#_Toc167854814)

[Описание выполненных пунктов базового и расширенного заданий: 3](#_Toc167854815)

Цель работы: изучение трехуровневой архитектуры веб-приложений (Клиент – Веб-сервер – Сервер БД) и языка написания серверного сценария php для построения динамического контента web-сайтов.

## Описание выполненных пунктов базового и расширенного заданий:

Базовое:

Установить компоненты 3-х уровневой архитектуры. В качестве звеньев 3-х уровневой архитектуры разрешается использовать Apache+PHP+MySQL (пакет программ WAMP – XAMPP, Денвер и т.п.). Язык серверных сценариев - PHP.

Подготовить базу данных, состоящую из 2 таблиц, связанные между собой отношением «один-ко-многим». Каждая таблица должна находиться в 3 нормальной форме и содержать не менее 5 полей. Содержимое полей таблицы должно соответствовать теме сайта. Все данные из таблиц должны отображаться на странице, включенной в разрабатываемый сайт. Для формирования веб-страницы использовать команду echo.

Расширенное:

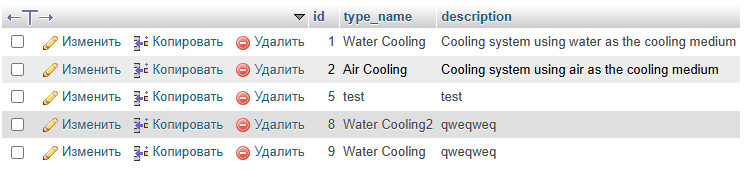
1. не отображать служебные поля (первичные и вторичные ключи).
2. все поля отобразить в одной таблице на веб-странице
3. добавить возможность ввода новой строки в таблицу базы на веб-странице
4. добавить возможность редактировать строку в таблицу базы на веб-странице
5. добавить возможность удаления строки из таблицы базы на веб-странице
6. применить оформление через css

Рассмотрим структуру таблиц БД:

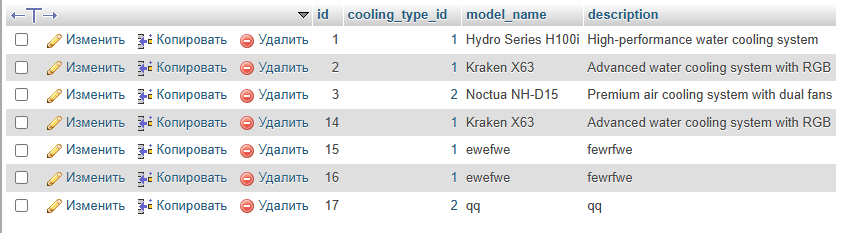
Таблицы связаны между собой отношением "один ко многим", где одна запись в таблице cooling\_types может быть связана с несколькими записями в таблице models.

Рассмотрим данные, хранимые в таблицах:

Содержимое таблицы типы систем охлаждения:



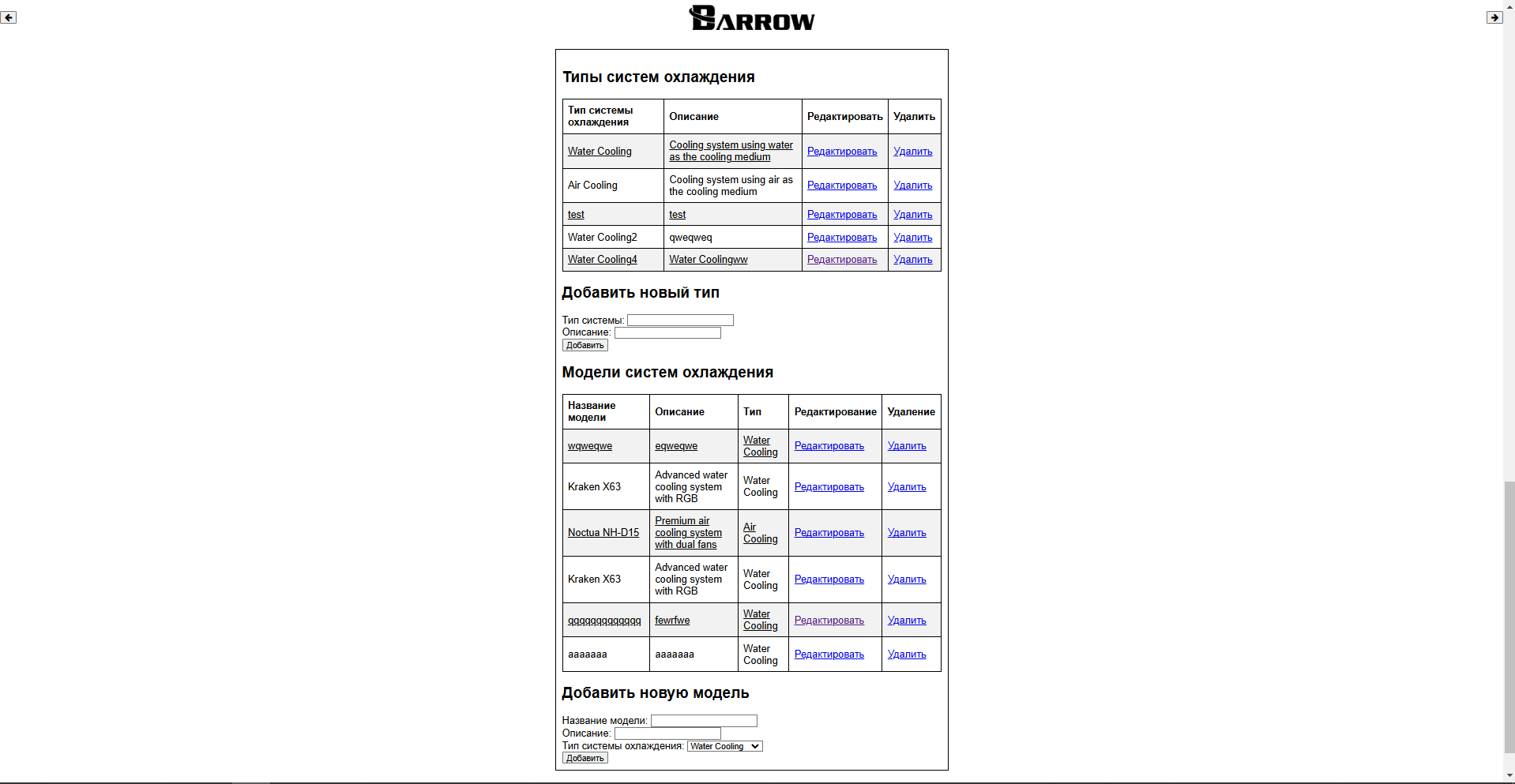
Содержимое таблицы модели систем охлаждения:



Листинг:

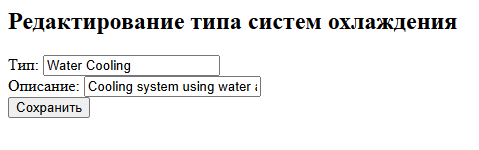
|  |
| --- |
| laba6.php  <?php  // Подключение к базе данных  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "laba6";  $conn = new mysqli($servername, $username, $password, $dbname);  // Проверка подключения  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  // Проверка, была ли отправлена форма для добавления новой модели  if(isset($\_POST['add\_model\_submit'])) {      $new\_model\_name = $\_POST['new\_model\_name'];      $new\_model\_description = $\_POST['new\_model\_description'];      $new\_cooling\_type\_id = $\_POST['new\_cooling\_type\_id'];      $sql\_insert\_model = "INSERT INTO models (cooling\_type\_id, model\_name, description) VALUES ('$new\_cooling\_type\_id', '$new\_model\_name', '$new\_model\_description')";      $conn->query($sql\_insert\_model);  }  // Проверка, была ли отправлена форма для добавления нового типа охлаждения  if(isset($\_POST['add\_cooling\_type\_submit'])) {      $new\_cooling\_type\_name = $\_POST['new\_cooling\_type\_name'];      $new\_cooling\_type\_description = $\_POST['new\_cooling\_type\_description'];      $sql\_insert\_cooling\_type = "INSERT INTO cooling\_types (type\_name, description) VALUES ('$new\_cooling\_type\_name', '$new\_cooling\_type\_description')";      $conn->query($sql\_insert\_cooling\_type);  }  // Выполнение SQL запроса для таблицы cooling\_types  $sql\_cooling\_types = "SELECT \* FROM cooling\_types";  $result\_cooling\_types = $conn->query($sql\_cooling\_types);  // Выполнение SQL запроса для таблицы models  $sql\_models = "SELECT models.id, models.model\_name, models.description, cooling\_types.type\_name FROM models INNER JOIN cooling\_types ON models.cooling\_type\_id = cooling\_types.id";  $result\_models = $conn->query($sql\_models);  ?>  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <title>Cooling Types and Models</title>  </head>  <body>      <h2>Типы систем охлаждения</h2>      <table border="1">          <tr>              <th>Тип системы охлаждения</th>              <th>Описание</th>              <th>Редактировать</th>              <th>Удалить</th>          </tr>          <?php          while($row = $result\_cooling\_types->fetch\_assoc()) {              echo "<tr>";              echo "<td>" . $row["type\_name"] . "</td>";              echo "<td>" . $row["description"] . "</td>";              echo "<td><a href='edit\_cooling\_type.php?id=" . $row["id"] . "'>Редактировать</a></td>";              echo "<td><a href='delete\_cooling\_type.php?id=" . $row["id"] . "' onclick='return confirm(\"Вы уверены?\")'>Удалить</a></td>";              echo "</tr>";          }          ?>      </table>      <h2>Добавить новый тип</h2>      <form method="post">          Тип системы: <input type="text" name="new\_cooling\_type\_name"><br>          Описание: <input type="text" name="new\_cooling\_type\_description"><br>          <input type="submit" name="add\_cooling\_type\_submit" value="Добавить">      </form>      <h2>Модели систем охлаждения</h2>      <table border="1">          <tr>              <th>Название модели</th>              <th>Описание</th>              <th>Тип</th>              <th>Редактирование</th>              <th>Удаление</th>          </tr>          <?php          while($row = $result\_models->fetch\_assoc()) {              echo "<tr>";              echo "<td>" . $row["model\_name"] . "</td>";              echo "<td>" . $row["description"] . "</td>";              echo "<td>" . $row["type\_name"] . "</td>";              echo "<td><a href='edit\_model.php?id=" . $row["id"] . "'>Редактировать</a></td>";              echo "<td><a href='delete\_model.php?id=" . $row["id"] . "' onclick='return confirm(\"Вы уверены?\")'>Удалить</a></td>";              echo "</tr>";          }          ?>      </table>      <h2>Добавить новую модель</h2>      <form method="post">          Название модели: <input type="text" name="new\_model\_name"><br>          Описание: <input type="text" name="new\_model\_description"><br>          Тип системы охлаждения:          <select name="new\_cooling\_type\_id">              <?php              $result\_cooling\_types->data\_seek(0); // Возврат к началу результата              while($row = $result\_cooling\_types->fetch\_assoc()) {                  echo "<option value='" . $row["id"] . "'>" . $row["type\_name"] . "</option>";              }              ?>          </select><br>          <input type="submit" name="add\_model\_submit" value="Добавить">      </form>  </body>  </html>  <?php  // Закрытие соединения с базой данных  $conn->close();  ?>  edit\_model.php  <?php  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "laba6";  $conn = new mysqli($servername, $username, $password, $dbname);  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  if(isset($\_POST['edit\_submit'])) {      $edit\_model\_id = $\_POST['edit\_model\_id'];      $edit\_model\_name = $\_POST['edit\_model\_name'];      $edit\_description = $\_POST['edit\_description'];      $sql\_edit = "UPDATE models SET model\_name='$edit\_model\_name', description='$edit\_description' WHERE id='$edit\_model\_id'";      $result\_edit = $conn->query($sql\_edit);      if ($result\_edit === TRUE) {          header("Location: laba6.php");          exit();      }  }  if(isset($\_GET['id'])) {      $edit\_model\_id = $\_GET['id'];      $sql\_get\_model = "SELECT \* FROM models WHERE id='$edit\_model\_id'";      $result\_get\_model = $conn->query($sql\_get\_model);      if ($result\_get\_model->num\_rows > 0) {          $row = $result\_get\_model->fetch\_assoc();          $edit\_model\_name = $row["model\_name"];          $edit\_description = $row["description"];      }  }  $conn->close();  ?>  <h2>Edit Model</h2>  <form method="post">  Model Name: <input type="text" name="edit\_model\_name" value="<?php echo $edit\_model\_name; ?>"><br>  Description: <input type="text" name="edit\_description" value="<?php echo $edit\_description; ?>"><br>  <input type="hidden" name="edit\_model\_id" value="<?php echo $edit\_model\_id; ?>">  <input type="submit" name="edit\_submit" value="Save Changes">  </form>  edit\_cooling\_type.php  <?php  // Подключение к базе данных  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "laba6";  $conn = new mysqli($servername, $username, $password, $dbname);  // Проверка подключения  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  // Проверка, был ли отправлен запрос на редактирование  if(isset($\_POST['edit\_submit'])) {      $edit\_cooling\_type\_id = $\_POST['edit\_cooling\_type\_id'];      $edit\_cooling\_type\_name = $\_POST['edit\_cooling\_type\_name'];      $edit\_cooling\_type\_description = $\_POST['edit\_cooling\_type\_description'];      $sql\_edit = "UPDATE cooling\_types SET type\_name='$edit\_cooling\_type\_name', description='$edit\_cooling\_type\_description' WHERE id='$edit\_cooling\_type\_id'";      $result\_edit = $conn->query($sql\_edit);      // После успешного редактирования перенаправляем пользователя обратно на laba6.php      if ($result\_edit === TRUE) {          header("Location: laba6.php");          exit();      }  }  // Получение ID записи для редактирования  if(isset($\_GET['id'])) {      $edit\_cooling\_type\_id = $\_GET['id'];      // Получение данных о типе охлаждения для редактирования      $sql\_get\_cooling\_type = "SELECT \* FROM cooling\_types WHERE id='$edit\_cooling\_type\_id'";      $result\_get\_cooling\_type = $conn->query($sql\_get\_cooling\_type);      if ($result\_get\_cooling\_type->num\_rows > 0) {          $row = $result\_get\_cooling\_type->fetch\_assoc();          $edit\_cooling\_type\_name = $row["type\_name"];          $edit\_cooling\_type\_description = $row["description"];      }  }  // Закрытие соединения с базой данных  $conn->close();  ?>  <!-- Форма для редактирования типа охлаждения -->  <h2>Edit Cooling Type</h2>  <form method="post">  Type Name: <input type="text" name="edit\_cooling\_type\_name" value="<?php echo $edit\_cooling\_type\_name; ?>"><br>  Description: <input type="text" name="edit\_cooling\_type\_description" value="<?php echo $edit\_cooling\_type\_description; ?>"><br>  <input type="hidden" name="edit\_cooling\_type\_id" value="<?php echo $edit\_cooling\_type\_id; ?>">  <input type="submit" name="edit\_submit" value="Save Changes">  </form>  delete\_model.php  <?php  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "laba6";  $conn = new mysqli($servername, $username, $password, $dbname);  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  if(isset($\_GET['id'])) {      $delete\_model\_id = $\_GET['id'];      $sql\_delete = "DELETE FROM models WHERE id='$delete\_model\_id'";      $result\_delete = $conn->query($sql\_delete);  }  $conn->close();  header("Location: laba6.php");  exit();  ?>  <?php  // Подключение к базе данных  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "laba6";  $conn = new mysqli($servername, $username, $password, $dbname);  // Проверка подключения  if ($conn->connect\_error) {      die("Connection failed: " . $conn->connect\_error);  }  // Получение ID записи для удаления  if(isset($\_GET['id'])) {      $delete\_cooling\_type\_id = $\_GET['id'];      // Удаление записи из таблицы cooling\_types      $sql\_delete = "DELETE FROM cooling\_types WHERE id='$delete\_cooling\_type\_id'";      $result\_delete = $conn->query($sql\_delete);      // После успешного удаления перенаправляем пользователя обратно на laba6.php      if ($result\_delete === TRUE) {          header("Location: laba6.php");          exit();      } else {          echo "Error deleting record: " . $conn->error;      }  }  // Закрытие соединения с базой данных  $conn->close();  ?> |

Рассмотрим результат выполненной работы:

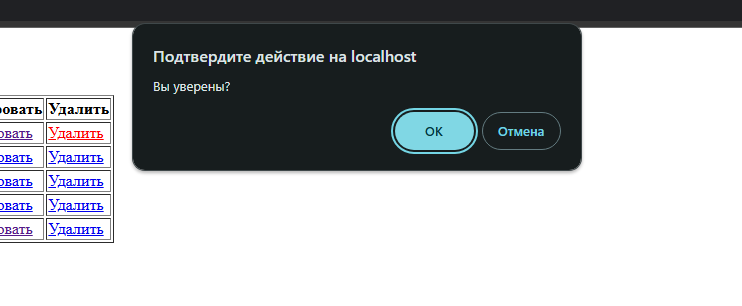


Присутствует удаление/редактирование/добавление записей.

Редактирование:



Удаление:



Вывод: в ходе лабораторной работы изучены трехуровневые архитектуры веб-приложений (Клиент – Веб-сервер – Сервер БД) и языка написания серверного сценария php для построения динамического контента web-сайтов.