

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное
учреждение высшего образования
“Национальный исследовательский университет ИТМО”

Факультет инфокоммуникационных технологий

ЛАБОРАТОРНАЯ РАБОТА №4

РАЗРАБОТКА ИНТЕРФЕЙСОВ ДЛЯ ВЫПОЛНЕНИЯ CRUD-ОПЕРАЦИЙ НАД
БАЗОЙ ДАННЫХ СРЕДСТВАМИ PHP

по дисциплине:

«Проектирование и реализация баз данных»

Выполнил студент:

Старовойтова Елизавета Анатольевна

Группа №К32402

Преподаватель:

Говорова Марина Михайловна

Санкт-Петербург

2023

Цель работы: овладеть практическими навыками разработки форм для вставки, выборки и редактирования данных.

Программное обеспечение: среда разработки XAMPP 8 (PHP 8) или аналогичная, СУБД PostgreSQL 1X, pgadmin 4.

Практическое задание:

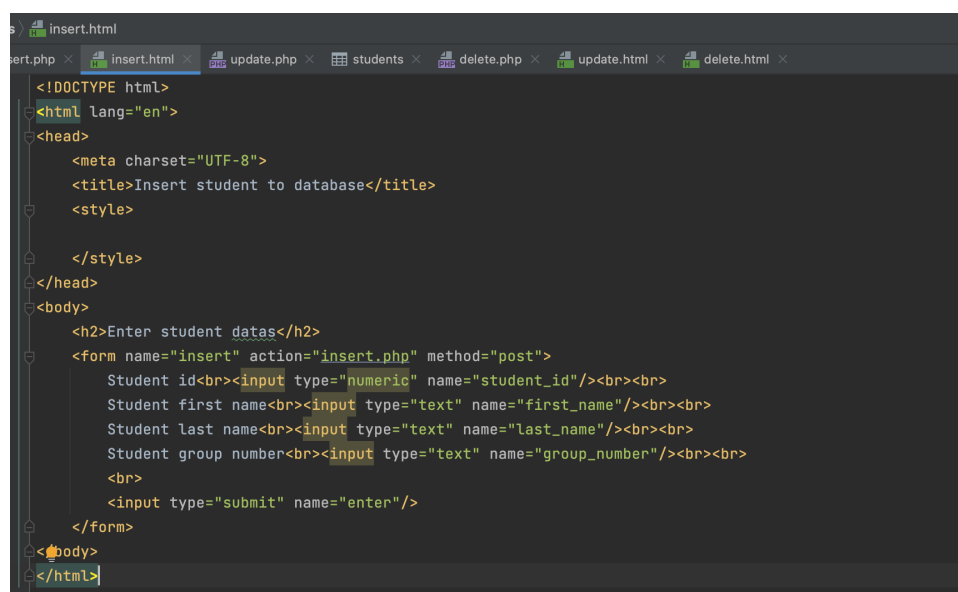
1. Изучить функции открытия соединения к базе данных средствами PHP.
2. Изучить основные функции для создания php-скрипта (на базе видео-уроков 1-7).
3. Создать сайт с использованием базовых возможностей PHP (в соответствии с содержанием видео-уроков 1-7).

Повышенный уровень: создать формы для вставки, редактирования, удаления и выборки данных с использованием библиотеки PDO.

Выполнение:

Таблица **students**, которая содержит student_id(PK), first_name, last_name, group_number.

INSERT:



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Insert student to database</title>
  <style>

</style>
</head>
<body>
  <h2>Enter student datas</h2>
  <form name="insert" action="insert.php" method="post">
    Student id<br><input type="numeric" name="student_id"/><br><br>
    Student first name<br><input type="text" name="first_name"/><br><br>
    Student last name<br><input type="text" name="last_name"/><br><br>
    Student group number<br><input type="text" name="group_number"/><br><br>
    <br>
    <input type="submit" name="enter"/>
  </form>
</body>
</html>
```

```
1 <?php
2 $dbuser = 'postgres';
3 $dbpass = 'Listar17102003';
4 $host = 'localhost';
5 $dbname = 'students';
6 $pdo = new PDO( dsn: "pgsql:host=$host;dbname=$dbname", $dbuser, $dbpass);
7
8 if ($_POST['enter']) {
9     echo "
10         student_id: $_POST[student_id]
11         <br>
12         first_name: $_POST[first_name]
13         <br>
14         last_name: $_POST[last_name]
15         <br>
16         group_number: $_POST[group_number]
17         <br>
18         <br>
19     ";
20 }
21 try {
22     $pgsql = "INSERT INTO students.students (student_id, first_name, last_name, group_number) VALUES (:student_id,:first_name,:last_name,:group_number)";
23     $stmt = $pdo->prepare($pgsql);
24     $stmt->bindValue( param: ':student_id', $_POST['student_id']);
25     $stmt->bindValue( param: ':first_name', $_POST['first_name']);
26     $stmt->bindValue( param: ':last_name', $_POST['last_name']);
27     $stmt->bindValue( param: ':group_number', $_POST['group_number']);
28     $stmt->execute();
29 } catch (PDOException $e) {
30     echo 'Ошибка выполнения запроса: ' . $e->getMessage();
31 }
32
33 echo "<b> Student was inserted!</b><br>";
34
35 <?>
```

В html файле реализована простая форма, куда нужно ввести данные. Нажав на кнопку Отправить, мы получаем значения введенных полей, с которыми работаем в php файле. Первым делом подключаемся к БД, а потом получаем данные, отправленные на сервер с помощью HTTP POST-запроса.

Мы создаем запрос на вставку студента в базу данных с помощью подготовленного выражения, связываем именованные параметры с переменными и выполняем запрос.

UPDATE:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Update Student Group</title>
6     <style>
7
8     </style>
9 </head>
10 <body>
11     <h2>Update Student Group</h2>
12     <form name="update" action="update.php" method="post">
13         Student ID: <br><input type="number" name="student_id"/><br><br>
14         New Group Number: <br><input type="text" name="group_number"/><br><br>
15         <br>
16         <input type="submit" name="enter"/>
17     </form>
18 </body>
19 </html>
```

```
insert.php x insert.html x update.php x students x delete.php x update.html x delete.html x
2 $dbuser = 'postgres';
3 $dbpass = 'listar17102003';
4 $host = 'localhost';
5 $dbname = 'students';
6 $pdo = new PDO( dsn: "pgsql:host=$host;dbname=$dbname", $dbuser, $dbpass);
7
8 if (isset($_POST['enter'])) {
9     $studentID = $_POST['student_id'];
10    $newGroupNumber = $_POST['group_number'];
11
12    // Получение старого значения group_number
13    $selectQuery = "SELECT group_number FROM students.students WHERE student_id = :student_id";
14    $stmt = $pdo->prepare($selectQuery);
15    $stmt->bindValue( param: ':student_id', $studentID);
16    $stmt->execute();
17    $oldGroupNumber = $stmt->fetchColumn();
18
19    echo "
20        student_id: $studentID
21        <br>
22        group_number (old): $oldGroupNumber
23        <br>
24        group_number (new): $newGroupNumber
25        <br>
26        <br>
27    ";
28
29    // Обновление значения group_number
30    $updateQuery = "UPDATE students.students SET group_number = :new_group_number WHERE student_id = :student_id";
31    $stmt = $pdo->prepare($updateQuery);
32    $stmt->bindValue( param: ':new_group_number', $newGroupNumber);
33    $stmt->bindValue( param: ':student_id', $studentID);
34    $stmt->execute();
35
36    echo "<b>Student group was updated!</b><br>";
37
38    // ... (rest of the code) ...
39}
```

По аналогии с insert все, единственное что, перед тем, как обновить номер группы я получаю старый.

Методом `fetchColumn()` я получаю значение старой группы.

DELETE:

```
websites delete.html
insert.php x insert.html x update.php x students x delete.php x update.html x delete.html x
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Delete student</title>
6 <style>
7
8 </style>
9 </head>
10 <body>
11 <h2>Delete student</h2>
12 <form name="delete" action="delete.php" method="post">
13   Enter student id<br><br>
14   <input type="numeric" name="student_id"><br><br><br>
15   <input type="submit" name="enter">
16 </form>
17 </body>
18 </html>
```

```
websites delete.php
insert.php x insert.html x update.php x students x delete.php x update.html x delete.html x
2 $dbuser = 'postgres';
3 $dbpass = 'Listar17102003';
4 $host = 'localhost';
5 $dbname = 'students';
6 $pdo = new PDO( "dsn: 'pgsql:host=$host;dbname=$dbname'", $dbuser, $dbpass);
7
8 if (isset($_POST['enter'])) {
9     $studentID = $_POST['student_id'];
10    $firstName = $_POST['first_name'];
11    $lastName = $_POST['last_name'];
12
13    // Получение студента до удаления
14    $selectQuery = "SELECT student_id, first_name, last_name FROM students.students WHERE student_id = :student_id";
15    $stmt = $pdo->prepare($selectQuery);
16    $stmt->bindValue( param: ':student_id', $studentID);
17    $stmt->execute();
18    $student = $stmt->fetch( mode: PDO::FETCH_ASSOC);
19
20    echo "
21        student_id: {$student['student_id']}
22        <br>
23        first_name: {$student['first_name']}
24        <br>
25        last_name: {$student['last_name']}
26        <br>
27        <br>
28    ";
29
30    $deleteQuery = "DELETE FROM students.students WHERE student_id = :student_id";
31    $stmt = $pdo->prepare($deleteQuery);
32    $stmt->bindValue( param: ':student_id', $studentID);
33    $stmt->execute();
34
35    echo "<b>Student {$student['first_name']} {$student['last_name']} was deleted!</b><br>";
36 }
37 }
```

Version Control | TODO | Problems | Terminal | Services

Все то же самое. Методом `fetch` извлекаю результат запроса. Режим `PDO::FETCH_ASSOC` возвращает ассоциативный массив, где ключи соответствуют именам столбцов.

Видео демонстрация

<https://drive.google.com/file/d/1GLPEIBSAUoTd5FwxuvvgBe-qIGS9W4YtW/view?usp=sharing>

Выводы:

В данной лабораторной работе я овладела практическими навыками создания форм вставки, редактирования и удаления данных.