# Лабораторная работа №5. Взаимодействие с реляционными СУБД (MySQL).

**Цель работы:** изучение возможностей языка программирования PHP по взаимодействию с реляционной СУБД MySQL.

**Порядок выполнения работы**

1. Изучить темы 4.1–4.4 лекционного материала.

2. Выполнить задание по лабораторной работе.

3. Представить для проверки результат выполнения работы в виде файлов и демонстрации выполнения на компьютере. БД в отдельном SQL-файле.

**Рекомендации по выполнению заданий**

Одной из наиболее часто встречающихся проблем при работе с базами данных является синхронизация кодировок. Чтобы избежать данной проблемы, необходимо следовать правилу: кодировки должны быть одинаковыми в:

– среде разработки приложения;

– файлах шаблонов HTML-страниц;

– конфигурационных файлах;

– таблицах БД;

– настройках взаимодействия MySQL и PHP.

Поскольку UTF8 становится стандартом де-факто, предлагается использовать именно эту кодировку, а в целях устранения проблем после установления соединения с СУБД и выбора БД следует выполнить два запроса:

SET CHARACTER SET 'UTF8' и SET NAMES 'UTF8'

Данная лабораторная работа не подразумевает разработку сложных SQL-запросов, однако следует уделить внимание корректности работы с СУБД на стороне PHP.

Для взаимодействия с MySQL в PHP предусмотрено три стандартных решения: использование расширения mysql – классический вариант, признанный устаревшим, использование расширения mysqli (MySQL Improved) – наиболее активно развивающийся способ, рекомендованный ныне к использованию большинством специалистов; использование расширения PDO (PHP Data Objects) – одно из наиболее перспективных направлений развития взаимодействия PHP с реляционными СУБД, предоставляющее дополнительный уровень абстракции и повышающий таким образом совместимость написанных на PHP программ с различными СУБД.

В данной лабораторной работе не предъявляется жёстких требований к использованию того или иного варианта взаимодействия PHP и MySQL, однако очень рекомендуется (допустим в расширенном задании, например, решив его иным методом, чем основное или различные методы в общем и основном) попробовать в решении оба современных варианта: mysqli (MySQL Improved) и PDO (PHP Data Objects), причём основной акцент сделать на применение расширения mysqli.

**Общее задание**

Это задание предполагает освоение общей структуры таблиц MySQL, типов полей, атрибутов, индексов (ключей), знакомство с основными SQL-запросами – поэтому рекомендуется применять приложение PhpMyAdmin.

* Создайте БД с кодировкой UTF8 (utf8\_general\_ci). Создайте 2 таблицы (тематика таблиц на усмотрение студента) с полями включающие поле с индексом (ключом) PRIMARY с автоинкрементом, 1-2 поля с уникальным индексом (ключом), 1-2 поля в одной из таблиц которое может быть связано с другой таблицей. Допустим, одна таблица данных о регистрации авторов и содержит, например, поля id, name, password, ip\_registration, data\_registration, а другая, к примеру, содержит тематические статьи с полями id, author\_id (имеется ввиду отсылка на id зарегистрированного автора из 1-й таблицы), title, text, image, data\_publications, hide, opinion и т.п. Подобрать корректные и оптимальные типы данных и длину для соответствующих полей, такие как varchar, text, int, double, bigint, enum, boolean, data, timestamp и т.п.
* Сделать небольшое первичное заполнение/редактирование/выделения через PhpMyAdmin. Изучить соответствующие SQL-команды.
* Изучить импорт/экспорт файла БД в формат SQL.
* Создать скрипт PHP для доступа к созданной БД. Определить константы для доступа к БД (localhost, имя, пароль,имя БД). Подключиться к БД, обработать ошибку подключения с соответствующим сообщением, установить кодировку UTF8, вывести всю БД и (или) часть полей с применением сортировки на экран в табличном виде. Рекомендуется также вывести объединенные данные из двух таблиц (например, имена авторов и соответствующие заголовки размещенных ими статей).

**Задания по вариантам**

**Вариант 1:** написать скрипт, отображающий содержимое некоторой заданной таблицы из БД (например, список компаний и их реквизитов, список товаров и цен, часовых поясов и смещения времени по Гринвичу относительно Минска и т.п.) в графическом виде, добавление, редактирование и удаление, просмотр записей. Удаление записи предусмотреть по различным полям.

**Вариант 2:** написать скрипт, отображающий структуру и данные всех таблиц указанной БД. (таблиц не менее 4-х, отобразить также первичные и вторичные ключи, и типы полей).

**Вариант 3:** написать скрипт, формирующий в виде HTML-страницы календарь, в котором ссылками являются даты, за которые в некоторой таблице, содержащей новости, присутствуют новости. Данные о датах и содержимых новостей и картинки находятся в БД.

**Вариант 4:** написать скрипт, выводящий в случайном порядке заданное количество неповторяющихся записей из произвольной таблицы БД.

**Вариант 5:** написать скрипт, позволяющий добавить в произвольную таблицу БД произвольное количество записей со случайными данными. Скрипт должен получать в качестве входных данных имена и типы ("число" или "текст") полей таблицы, а также количество добавляемых записей.

**Вариант 6:** написать скрипт, получающий через форму e-mail пользователя, проверяющий его корректность и добавляющий его в таблицу БД в случае, если такого e-mail там ещё нет.

**Вариант 7:** написать скрипт, получающий через форму имя пользователя, пароль и подтверждение пароля. Если такого пользователя ещё нет в БД, а пароль и его подтверждение совпадают, необходимо добавить в БД имя пользователя и пароль в виде хэша sha1. А также добавить функцию изменения пароля.

**Вариант 8:** сформировать БД, содержащую информацию о студентах и их оценках по различным предметам (не менее пяти). Написать скрипт, формирующий список студентов с их средними баллами, а также минимальным и максимальным баллом с указанием списка предметов, за которые был получен такой балл. Предусмотреть корректировку по баллам для студента. Отобразить полный список и получившийся.

**Вариант 9:** написать скрипт, позволяющий выполнить произвольный запрос к СУБД или серию произвольных запросов. после чего отображающий результат выполнения запроса и статистику: время выполнения, использованная оперативная память. Запрос или серию запросов получать через веб-формы.

**Вариант 10:** Создать базу городов. Далее участвуют человек и компьютер. Необходимо назвать город, дальше получаем ответ от компьютера с вероятностью в 97.4% название города, чьё название начинается на последнюю букву названного игроком города. Далее ситуация повторяется, игрок должен назвать город у которого название начинается с последней буквы названным опонентом города. Имена не могут повторяться.

**Вариант 11:** Написать скрипт, который бы по заданному году выводит ближайшие пять лет названия по китайскому календарю. Год получать через вебформу.

**Расширенное задание**

На основе результатов выполнения расширенных заданий в лабораторных работах 1–4 доработать программу формирования пользовательской и администраторской частей сайта в контексте взаимодействия с СУБД, а именно:

– реализовать хранение структуры сайта в БД;

– реализовать построение карты сайта и поиска по сайту;

– реализовать протоколирование действий администратора;

– реализовать такие модули сайта как (на выбор) голосование, показ случайного

баннера, подписка на рассылку.

**Полезные ссылки:**

http://jtest.ru/bazyi-dannyix/

https://www.php.net/manual/ru/mysqli.quickstart.php

https://www.php.net/manual/ru/class.mysqli.php

https://www.php.net/manual/ru/book.pdo.php

https://www.php.net/manual/ru/class.pdo.php

Аврамец Дмитрий Владимирович (ПОИТ)

dva@bsuir.by