Практическая работа Реализация записи данных в БД

Цель: научиться отправлять данные в БД

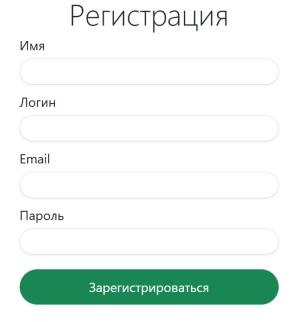
Используемые технологии: HTML, CSS, Bootstrap 5.8, PHP 8.1, MySQL 5.7, PHPMyAdmin.

Задание: создайте страницу регистрации с формой для добавления пользователя в систему. Форма должна состоять из следующих полей: имя, логин, email, пароль. По кнопке «Зарегистрироваться» данные должны заносится в базу данных в таблицу users. Интерфейс оформить с помощью фреймворка bootstrap5.

Порядок выполнения работы:

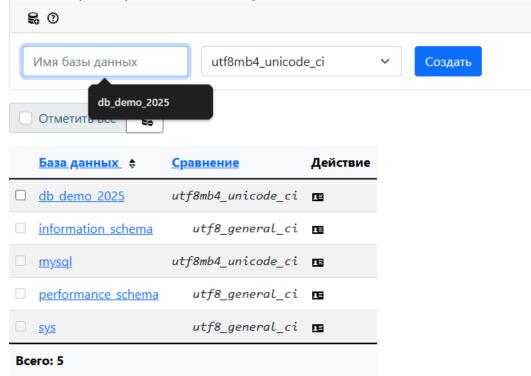
- 1. Создайте в папке проекта файл index.php, внутри файла создайте стандартную структуру HTML.
- 2. Добавляем теги header, section, footer.
- 3. В тег header копируем nav.navbar со всем содержимым из Zeal: Bootstrap5 Guides Navbar раздел Supported content.
- 4. Создаем папку assets, сбрасываем туда содержимое архива с bootstrap5.
- 5. С помощью тега link подключаем файл bootstrap.min.css и проверяем его работу на странице.
- 6. В тег section копируем form со всем содержимым из Zeal: Bootstrap5 Guides forms– раздел Overview.
- 7. Редактируем форму и навигационное меню в соответствии с заданием.
- 8. Далее с помощью классов bootstrap наводим красоту. Помните, что наша задача получить приятный дизайн страницы быстро и с наименьшим количеством кода.
- 9. Далее рекомендации по использованию классов, вы можете использовать другие по желанию. Меняеме нав. панели класс bg-body-tertiary на bg-primary и добавляем navbar-dark. У контейнеров на странице выставить внутренние отступы классом p-3, размер шрифта fs-4. Для тега section прописать дополнительное css свойство минимальная высота около 80% от высоты окна браузера. Круглые поля ввода можно получить классами rounded-pill shadow-sm px3. Для позиционирования тег form положить в див с классом container py-3, самой форме прописать w-50 mx-auto. Остальное подбираем по вкусу самостоятельно, интерфейс страницы должен получится примерно как на изображении ниже.

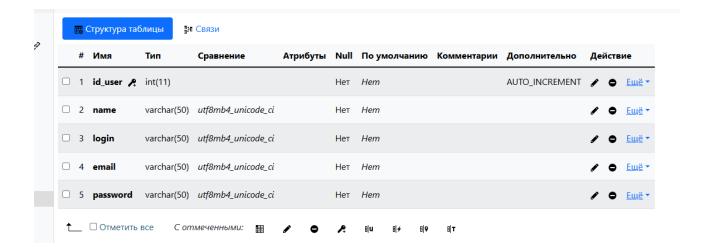
Пример записи данных в БД



Демоэкзамен близко Готовимся

- 10. Интерфейс готов, переходим к созданию БД. Заходим в PHPMyAdmin через вкладку дополнительно в Open Server Panel. По умолчанию логин: root, пароль пустой.
- 11. Создаем новую базу данных и таблицу users в ней.





- 12. Теперь создадим обработчик для нашей формы на странице, который будет записывать данные из формы в таблицу users, находится он пока будет в том же файле index.php.
- 13. Для начала добавьте атрибуты тегу form, action будет пустой(потому что данные обрабатываться будут на этой странице) и атрибут method со значением post.
- 14. Далее создаем блок php в начале страницы и подключаем нашу базу данных с помощью глобальной переменной connect и метода mysqli. Параметры подключения: "localhost", "root", "", "имя вашей базы данных".
- 15. Далее с помощью переменной sql формируем SQL-запрос для вставки новых данных пользователя (имя, логин, электронная почта, пароль) в таблицу users. Функцией sprintf форматируем строку для запроса в ней будет все остальное. Для ускорения работы, можно получить строку INSERT INTO... через PhpMyAdmin. Выбираем таблицу переходим на вкладку SQL нажимаем кнопку Insert. Вставляем внутрь sprintf. После строки INSERT INTO через запятую перечисляем ключи ассоциативного массива данных \$_POST,переданных через HTTP методом POST от формы. У полей формы должны быть прописаны соответствующие пате.
- 16. Далее выполняем SQL-запрос с помощью переменной \$result, в которую записывается результат выполнения запроса с помощью метода ->query(\$sql) у переменной \$connect
- 17. Полученный код обернем в условную конструкцию для проверки, был ли запрос методом POST (т.е. отправлена ли форма), если этого не сделать сервер будет ругаться на необъявленные переменные, пока форма первый раз не будет отправлена. От данного костыля мы позднее откажемся при дальнейшем изучении PHP, а пока поздравляю, если вы все сделали правильно, записи в базу будут добавляться.

```
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
}
```