

## ЗАДАНИЕ 3

Составьте HTML-форму с полями:

- имя (текстовое поле);
- e-mail (текстовое поле);
- год рождения (выбор из списка);
- пол (радиокнопки);
- количество конечностей (радиокнопки);
- сверхспособности: бессмертие, прохождение сквозь стены, левитация (множественный выбор из списка);
- биография (многострочное текстовое поле);
- с контрактом ознакомлен (чекбокс);
- кнопка «Отправить».

Оформить страницу красиво с использованием CSS. Предполагается использование верстки из предыдущего семестра.

Реализуйте скрипт на веб-сервере на PHP или другом языке программирования, сохраняющий в базу данных MySQL (MariaDB) заполненную форму.

До записи необходимо проверить корректность заполнения всех полей на веб-сервере. В случае некорректного заполнения отобразить ошибку с описанием проблемы. В случае успешной записи отобразить сообщение о том, что данные успешно сохранены.

Для каждой отправки формы необходимо создать отдельную строку с данными в таблице, использовать поле автоинкремент для выдачи уникального идентификатора записи. Выбранные способности хранить в отдельной таблице используя связь один ко многим. База данных должна быть в нормальной форме (см. методические указания).

Для записи в базу данных использовать подготовленные запросы (prepared statements).

## СДАЧА РАБОТЫ И ОЦЕНИВАНИЕ

В Moodle на проверку отправляется ссылка на веб-страницу с формой и ссылка на репозиторий GIT.

Время выполнения задания до 8 часов, включая самостоятельную работу.

За задание выставляется **до 8 баллов**, по одному на каждое корректно проверяемое и сохраняемое в базу данных поле.

Оценка снижается в 2 раза если база данных не находится в 3й нормальной форме.

Допускается не полное выполнение задания. Не допускается просрочка выполнения задания (просроченное задание оценивается в 0 баллов).

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Ознакомьтесь с содержанием лекций 4 и 5. Используйте учебный или самостоятельно настроенный вами веб-сервер. Для студентов 25 группы (администрирование информационных систем) рекомендуется настроить веб-сервер самостоятельно.

На учебном сервере настроена СУБД MySQL (MariaDB) и для вас создана база данных, название которой совпадает с вашим логином.

Пароль и логин MySQL совпадает с вашим логином и паролем на учебном сервере. До выполнения задачи создайте таблицу необходимой структуры. Для этого подключитесь по SSH и запустите клиент MySQL в командной строке:

```
mysql -u<login> -p<pass>
```

Выполните команду для подключения к вашей базе данных:

```
use <login>
```

Создайте таблицу для сохранения форм:

```
CREATE TABLE application (  
    id int(10) unsigned NOT NULL AUTO_INCREMENT,  
    name varchar(128) NOT NULL DEFAULT '',  
    PRIMARY KEY (id)  
);
```

Потренируйтесь работать с таблицей с помощью запросов INSERT, SELECT, UPDATE и DELETE.

Используйте официальную документацию MariaDB или MySQL для изучения SQL:

<https://mariadb.com/kb/en/data-manipulation/>

<https://dev.mysql.com/doc/refman/8.0/en/sql-data-manipulation-statements.html>

Удалите таблицу:

```
DROP TABLE application;
```

Модифицируйте запрос CREATE TABLE для создания всех требуемых в задаче столбцов таблицы. См. пример db.sql, документацию на CREATE TABLE и типы данных чтобы подобрать подходящие типы колонок БД:

<https://mariadb.com/kb/en/data-types/>

<https://dev.mysql.com/doc/refman/8.0/en/data-types.html>

Для хранения выбранных сверхспособностей используйте отдельную таблицу, используйте 3-ю нормальную форму при проектировании структуры БД так как в дальнейших заданиях потребуются изменение данных в БД и построение отчетов. В полном решении должно быть три таблицы – таблица сущности заявка, таблица сущности способность и таблица связи один ко многим заявка-способность.

Составьте SQL-запросы для вставки данных в таблицы.

Поправьте пример index.php для соединения с вашей базой данных. Залейте на учебный сервер index.php и form.php и проверьте, что при отправке формы строка вставляется в базу данных.

Модифицируйте index.php и form.php для ввода, валидации и вставки остальных требуемых полей.

Для получения ID последней вставленной AUTOINCREMENT записи используйте PDO `lastInsertId()`:

<https://www.php.net/manual/en/pdo.lastinsertid.php>

В HTML форме поле способностей реализуйте через select с множественным выбором: `<select name="abilities[]" multiple="multiple">`

Используйте цикл `foreach()` в PHP для перебора значений поля множественного выбора и вставки выбранных способностей в БД:

```
foreach ($_POST['abilities'] as $ability) {  
    // Вставляем $ability в БД  
}
```