

## Практическая работа № 7.2

### «Подготовка к реализации функционала системы. Часть 2»

**Цель работы:** закрепление практических навыков работы с функциональным и объектно-ориентированным подходом.

**Формируемые компетенции:** ПК 5.4 «Производить разработку модулей информационной системы в соответствии с техническим заданием», ПК5.1 «Собирать исходные данные для разработки проектной документации на информационную систему».

**Отчет о работе:** предоставляется в составе отчета по практике. Студент защищает написанный код очно.

Для работы с базой данных будет использоваться класс `mysqli`.

<https://www.php.net/manual/ru/book.mysqli.php>

#### Задачи:

##### Задача 1.

Создать класс `Request` - класс «обертку» для обработки запросов.

В классе реализовать следующие атрибуты:

- 1) `isPost` – признак, что запрос был совершен методом `POST`,
- 2) `isGet` – признак, что запрос был совершен методом `GET`.

Для заполнения атрибутов использовать `$_SERVER["REQUEST_METHOD"]`.

В конструкторе класса проинициализировать атрибуты.

Создать следующие методы:

- 1) метод «очистки» входного параметра;
- 2) метод очистки массива (с рекурсивным вызовом если необходимо)
- 3) метод `post`, данный метод может вызываться с параметром либо без него:
  - а) при вызове без параметра – вернуть очищенный массив `$_POST`
  - б) при вызове с параметром – вернуть очищенный параметр из массива `$_POST`
- 4) метод `get`, данный метод может вызываться с параметром либо без него:
  - а) при вызове без параметра – вернуть очищенный массив `$_GET`
  - б) при вызове с параметром – вернуть очищенный параметр из массива `$_GET`
- 5) метод возвращающий `host` запроса из массива `$_SERVER`
- 6) метод возвращающий параметр `token` из строки запроса. Если токен отсутствует, метод должен возвращать `null`.

В файле инициализации приложения создать экземпляр класса Request.

## Задача 2.

В файле конфигурации приложения создать массив с параметрами, необходимыми для подключения к базе данных.

Создать класс `MySQL` - класс «обертку» для класса `mysqli`.

В конструктор класса, должен принимать массив для подключения к БД.

В классе реализовать следующие атрибуты:

- 1) `isConnected` – признак, что есть соединение сервером БД.

В конструкторе класса вызвать родительский конструктор и проинициализировать атрибут `isConnected`.

Создать следующие методы:

- 1) метод выполняющий запрос к бд и возвращающий результат в виде ассоциативного массива. Метод должен принимать код запроса;

- 2) метод проверки поля на уникальность. Метод должен принимать:

- а) имя таблицы,
- б) поле поиска,
- в) значение поиска.

Создать экземпляр класса в файле инициализации приложения.

Проверить корректность работы приложения.