

Курс:

**«Создание web-приложений, исполняемых на стороне сервера при помощи языка программирования PHP и технологии AJAX»**

Встреча 6

ТЕМА: ПАТТЕРНЫ MVC И DI

## Задание 1

В этом задании вам надо будет создать собственную упрощенную реализацию MVC паттерна.

Создайте страницу `view.php` на которой создайте кнопку (или ссылку) с подписью All Records.

Затем создайте два класса с именами `MyController` и `MyModel`.

В классе `MyController` создайте метод с именем `getAllRecords()`. Этот метод должны стать обработчиками кнопки All Records.

В классе `MyModel` создайте метод с именем `getAllRecordsModel()`. Этот метод должен выполнять запрос `"select * from Pictures"` и возвращать результат выполнения этого запроса (для этого задания вместо таблицы `Pictures` можно использовать любую другую таблицу).

Вам надо использовать паттерн `Dependency Injection`, чтобы в методе `getAllRecords()` класса `MyController` можно было вызывать метод `getAllRecordsModel()` класса `MyModel`. Результат вызова метода `getAllRecordsModel()` должен отображаться на странице `view.php` в табличном виде.

Курс:

**«Создание web-приложений, исполняемых на стороне сервера при помощи языка программирования PHP и технологии AJAX»**

Встреча 6

ТЕМА: СЕРИАЛИЗАЦИЯ И ДЕСЕРИАЛИЗАЦИЯ

### **Задание 2**

Для выполнения задания вам понадобится класс `Picture` со свойствами `$id`, `$name`, `$size` и `$imagepath`, созданный ранее. Если этого класса у вас нет, можно использовать любой другой.

Создайте два файла с именами `serialize.php` и `unserialize.php`.

В файле `serialize.php` создайте объект класса `Picture` (или другого класса). Сериализованный объект запишите в текстовый файл.

Теперь в файле `unserialize.php` прочитайте из текстового файла строку и восстановите из нее исходный объект.

Убедитесь, что восстановленный объект полностью функционален (вызовите от имени этого объекта какой-либо метод класса).