

Практическая работа № 7.6

«Создание класса Post с функционалом создания поста»

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.

Задачи:

Задача 1.

В классе обработки данных создать методы:

- 1) метод заменяющий все включения «\r\n» на «
». Для реализации метода можно воспользоваться вариантом функции замены: `preg_replace('/\v+|\\r\\n/ui','
','<данные>');`
- 2) метод заменяющий все включения «
» на «\r\n».
- 3) метод преобразующий переданную дату из формата «Y-m-d H:i:s» в формат «d.mY H:i:s». Использовать объектный подход работы с [датой/временем](#).

Задача 2.

Создать класс Post для создания/редактирования и просмотра постов. Данный класс должны наследовать класс, связанный с обработкой данных. Класс должен принимать экземпляр пользователя. Создать экземпляр класса Post в файле инициализации приложения.

Создать следующие методы:

- 1) метод `validate` – метод валидации данных. В методе должен вызываться родительский метод валидации.
- 2) метод `load` – метод загрузки данных из входного массива. Для реализации метода необходимо родительский метод;
- 3) метод `save` – метод сохранения данных. Для реализации метода использовать экземпляр класса БД, содержащийся в атрибуте с экземпляром пользователя. Необходимо преобразовать контент поста – преобразовать «\r\n» в «
»

Задача 3.

В файле страницы просмотра всех постов, создать условие отображения кнопки создания поста. Ссылка должна вести на страницу создания поста.

Задача 4.

В файле страницы создания поста доработать форму аналогично регистрации или авторизации пользователя. При отображении текста поста заменить «
» на «\r\n», используя ранее созданный метод.

Задача 5.

В файле инициализации страницы создания поста произвести проверку, что метод запроса POST, пользователь авторизован и не является администратором. Загрузить данные в экземпляр поста и проверить данные на валидацию. В случае успеха предыдущих проверок, сохранить данные в БД, используя ранее созданный метод. Сделать перенаправление на страницу поста с использованием параметра идентификатора поста. Проверить, что данные в БД сохраняются корректно.

Проверить корректность работы приложения.