

# ДЗ №4 - “Проектирование модели данных”

Целью домашнего задания является проектирование модели базы данных, наполнение ее тестовыми данными и отображение и отображение этих данных на сайте. Т.е. создание read-only сайта.

## Проектирование модели

Используя Django Models спроектировать схему вашего приложения. Создать модели для основных сущностей: вопрос, ответ, тэг, профиль пользователя, лайк. Важно использовать правильные типы данных и проставить необходимые связи (ForeignKey) в моделях. Для вопросов создать свой Model Manager, в котором определить типичные выборки (“лучшие” и “новые” вопросы). Поскольку мы будем использовать встроенную авторизацию Django (django.contrib.auth) - у нас будет готовая модель пользователя (django.contrib.auth.models.User), но необходимо создать дополнительную модель (Profile) в которую разместить дополнительные поля, например avatar. Модель Profile и User должны быть связаны 1 к 1.

## ModelManager и методы моделей

В коде вьюшек не должно быть кода бизнес логики (анти-паттерн fat controller). Все методы по выборке и сортировке данных - в ModelManager. Вся логика по выводу данных, например построение полного пути к вопросу - в методы модели. Пагинация - в отдельную функцию. Вьюшки должны состоять из 4-6 строк.

## Наполнение данными

Для удобства дальнейшей разработки нужно наполнить базу тестовыми данными. Для этого нужно создать скрипт (а лучше это сделать в виде команды приложения - Management Command), в котором добавить необходимые объекты в базу. При добавлении удобно использовать уже созданные модели, а не писать сырой SQL.

Обратите внимание, что при указании связи в моделях можно использовать как id, так и объекты на которые вы ссылаетесь:

```
u = User.objects.get(pk = 1)
q = Question(author = u, text = 'wut?') # ссылка на объект
q.save()
a = Answer(question_id = 1, text = 'blabla') # ссылка по id
```

```
a.save()
```

В базе данных в любом случае будут храниться поля вида `author_id` и `question_id`.

## Отображение данных

Разработать view для отображения следующих страниц:

- Список новых вопросов
- Список “лучших” вопросов
- Список вопросов по тэгу
- Страница 1 вопроса со списком ответов

Для страниц не существующих объектов нужно возвращать 404 страницу. В коде view (отображений) не должно быть дублирующегося кода. Использовать функцию пагинации созданную на предыдущем занятии.

## Полезные ссылки

1. Тьюториал по проектированию моделей в Django  
<http://djbook.ru/rel1.7/intro/tutorial01.html#creating-models>
2. Подробная документация по моделям  
<http://djbook.ru/rel1.7/topics/db/models.html>
3. Скрипты - Management Commands  
<http://djbook.ru/rel1.7/howto/custom-management-commands.html>