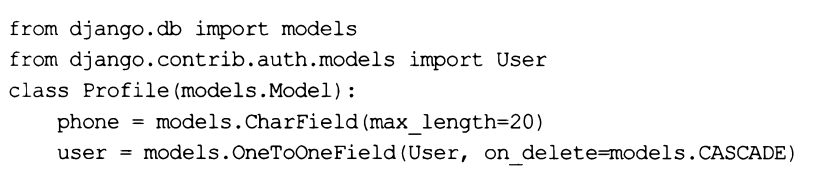
**Тема занятия № 46: Модуль 28. Создание своей модели пользователя**

1. Создание своей модели пользователя

Для хранения списка пользователей в подсистеме разграничения доступа Django предусмотрена стандартная модель user, объявленная в модуле django.contrib.

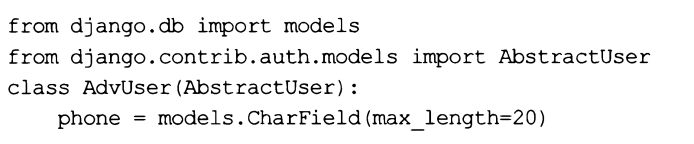
**Auth.models.** Эта модель хранит объем сведений о пользователе, вполне достаточный для многих случаев. Однако часто приходится сохранять в составе сведений о пользователе дополнительные данные: номер телефона, интернет-адрес сайта, признак, хочет ли пользователь получать по электронной почте уведомления о новых сообщениях, и т. п.

Можно объявить дополнительную модель, поместить в нее поля для хранения всех нужных данных и добавить поле, устанавливающее связь ”один-с-одним” со стандартной моделью пользователя. Вот пример создания подобной дополнительной модели:



Разумеется, при создании нового пользователя придется явно создавать связанную с ним запись модели, хранящую дополнительные сведения. Зато не будет никаких проблем с подсистемой разграничения доступа и старыми дополнительными библиотеками, поскольку для хранения основных сведений о пользователях будет использоваться стандартная модель user.

Другой подход заключается в написании своей собственной модели пользователя. Такую модель следует сделать производной от класса Abstractuser, который объявлен в модуле django.contrib.auth.models, реализует всю функциональность по хранению пользователей и представляет собой абстрактную модель — собственно, класс стандартной модели пользователей user также является производным от класса Abstractuser. Пример:

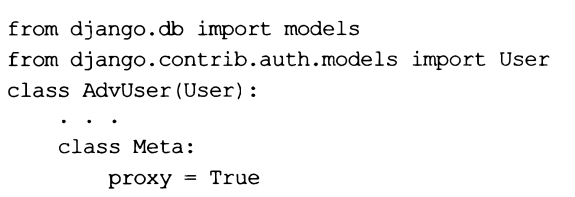
****

Новую модель пользователя следует указать в параметре auth user model настроек проекта.



В таком случае не придется самостоятельно создавать связанные записи, хранящие дополнительные сведения о пользователе — это сделает Django. Однако нужно быть готовым к тому, что некоторые дополнительные библиотеки, в особенности старые, не считывают имя модели пользователя из параметра auth user model, а обращаются напрямую к модели user. Если такая библиотека добавит в список нового пользователя, то он будет сохранен без дополнительных сведений, и код, использующий эти сведения, не будет работать.

Если нужно лишь расширить или изменить функциональность модели пользователя, то можно создать на его основе прокси-модель, также не забыв занести ее в параметр

****

И наконец, можно написать полностью свой класс модели. Однако такой подход применяется весьма редко из-за его трудоемкости. Интересующиеся могут обратиться к странице <https://docs.djangoproject.com/eii/3.0/topics/aiith/custoimzing/>, где приводятся все нужные инструкции.