

Django Starter

Модели (часть 2)

Django Starter

Introduction



Лазорык Михаил

Software developer, 3 года опыта

 mykhailo.lazoryk

 mykhailo-lazoryk



Модели (часть 2)

Django Starter

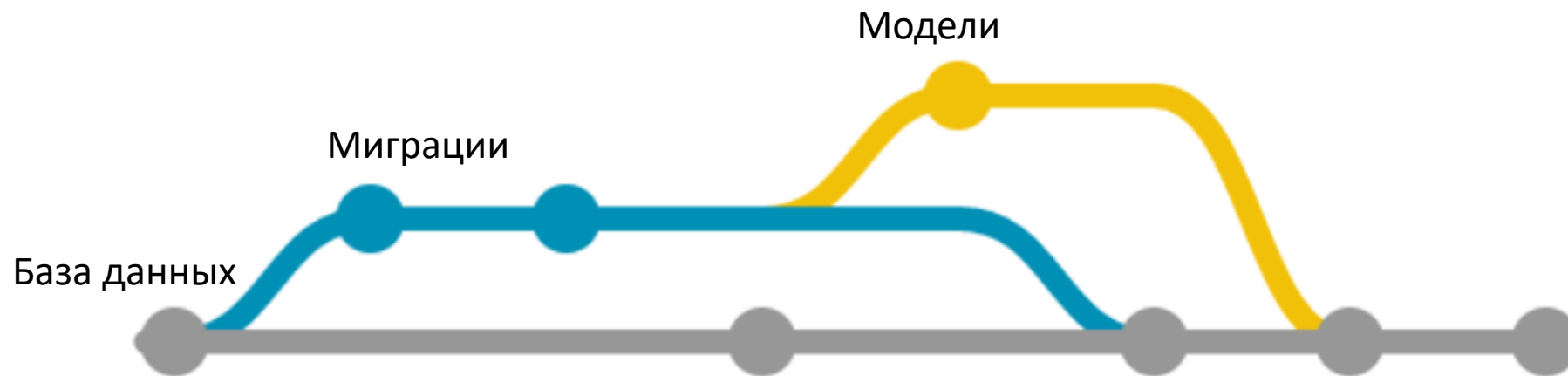
План урока

1. Миграции
2. Использование Django ORM
3. Менеджеры

Django Starter

Миграции

- **Миграции** - это система контроля версий для базы данных, где **makemigrations** отвечает за сохранение состояния моделей в файле миграции - аналог коммита - а **migrate** отвечает за их применение к базе данных.
- Файлы с миграциями находятся в каталоге **"migrations"** приложения.



Django Starter

Миграции

Django использует миграции для переноса изменений в моделях (добавление поля, удаление модели и т.д.) на структуру базы данных. Миграции создавались в основном для автоматической работы, но необходимо знать когда их создавать, запускать и как решать различные проблемы.

Django предоставляет три команды для работы с миграциями и структурой базы данных:

- **migrate**, которая отвечает за применение миграций, за откат миграций и за вывод статуса миграций.
- **makemigrations**, которая отвечает за создание новых миграций на основе изменений в моделях.
- **sqlmigrate**, которая выводит SQL запросы для миграции.

Django Starter

Миграции в SQLite и PostgreSQL

SQLite очень плохо поддерживает изменения в структуре базы данных, но Django пытается эмулировать их следующим образом:

- Создание новой таблицы для новой структуры
- Копирование данных в новую таблицу
- Удаление старой таблицы
- Переименование новой таблицы

Этот процесс, как правило, хорошо работает, но может быть медленным и иногда глючит.

Не рекомендуется использовать и мигрировать SQLite на "боевом" сервере, если вы не очень осведомлены о рисках и его ограничениях.

Django поддерживает SQLite, чтобы позволить разработчикам использовать SQLite для разработки простых проектов.

VS

PostgreSQL предоставляет больше всего возможностей для миграций структуры данных. Единственное ограничение в том, что добавление столбцов со значениями по умолчанию вызывает полную перезапись таблицы и требует времени, пропорционально ее размеру.

По этой причине рекомендуется всегда создавать новые столбцы с `null=True`, т.к. таким образом они будут добавлены сразу.

Django Starter

Использование Django ORM

Вытягивание данных из базы данных происходит через Django ORM которая возвращает QuerySet.

- ORM (англ. Object-Relational Mapping, рус. объектно-реляционное отображение, или преобразование) — технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных».
- QuerySet, по сути, — список объектов заданной модели. QuerySet позволяет читать данные из базы данных, фильтровать и изменять их порядок.

QuerySet может быть создан, отфильтрован, нарезан и, как правило, передан без фактического запроса к базе данных.

Django Starter

Менеджеры

Менеджер (Manager) - это интерфейс, через который создаются запросы к моделям Django. Каждая модель имеет хотя бы один менеджер.

По умолчанию Django добавляет Manager с именем objects для каждого класса модели. Однако, если вы хотите использовать objects, как имя поля, или хотите использовать название, отличное от objects для Manager, вы можете переименовать его для модели.

Чтобы переименовать Manager добавьте в класс атрибут, значение которого - экземпляр `models.Manager()`.

Django Starter

План урока

1. Миграции
2. Использование Django ORM
3. Менеджеры

Проверка знаний

TestProvider.com



Проверьте как Вы усвоили данный материал на TestProvider.com

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.

Django Starter

Спасибо за внимание! До новых встреч!



Лазорык Михаил
Software developer



Информационный видеосервис для разработчиков программного обеспечения

