

Посвящается моей сестре

Содержание

От издательства	17
Вступительное слово	18
Об авторе	20
О рецензенте	21
Предисловие	22
Глава 1. Разработка приложения для ведения блога	28
Установка языка Python	29
Создание виртуальной среды Python	30
Установка веб-фреймворка Django	31
Установка Django с помощью pip	31
Новые функциональные возможности Django 4.....	32
Общий обзор веб-фреймворка Django.....	33
Главные компоненты веб-фреймворка	33
Архитектура Django	34
Создание первого проекта.....	35
Применение первоначальных миграций базы данных.....	36
Запуск и выполнение сервера разработки.....	37
Настроечные параметры проекта	39
Проекты и приложения.....	40
Создание приложения.....	41
Создание моделей данных блога	42
Создание модели поста.....	42
Добавление полей даты/времени.....	44
Определение предустановленного порядка сортировки	45
Добавление индекса базы данных	46
Активация приложения	47
Добавление поля статуса	47

Добавление взаимосвязи многие-к-одному	50
Создание и применение миграций.....	51
Создание сайта администрирования для моделей.....	54
Создание суперпользователя.....	54
Сайт администрирования	55
Добавление моделей на сайт администрирования	56
Адаптация внешнего вида моделей под конкретно-прикладную задачу	58
Работа с наборами запросов QuerySet и менеджерами	60
Создание объектов.....	61
Обновление объектов.....	62
Извлечение объектов	63
Применение метода filter()	63
Применение метода exclude().....	64
Применение метода order_by()	64
Удаление объектов.....	64
Когда вычисляются наборы запросов QuerySet	65
Создание модельных менеджеров	65
Разработка представлений списка и детальной информации	67
Создание представлений списка постов и детальной информации о посте	67
Применение функции сокращенного доступа get_object_or_404().....	68
Добавление шаблонов URL-адресов представлений.....	69
Создание шаблонов представлений.....	71
Создание базового шаблона.....	72
Создание шаблона списка постов	73
Доступ к приложению	74
Создание шаблона детальной информации о посте	74
Цикл запроса/ответа.....	75
Дополнительные ресурсы.....	76
Резюме	77
Присоединяйтесь к нам на Discord	78

Глава 2. Усовершенствование блога за счет продвинутых функциональностей

функциональностей	79
Использование канонических URL-адресов для моделей	80
Создание дружественных для поисковой оптимизации URL-адресов постов.....	82
Видоизменение шаблонов URL-адресов	84
Видоизменение представлений	85
Видоизменение канонического URL-адреса постов.....	86
Добавление постраничной разбивки.....	87
Добавление постраничной разбивки в представление списка постов	87
Создание шаблона постраничной разбивки	88
Обработка ошибок постраничной разбивки	91
Разработка представлений на основе классов	94

Зачем использовать представления на основе классов	95
Использование представления на основе класса для отображения списка постов	95
Рекомендация постов по электронной почте	97
Разработка форм с помощью Django	98
Работа с формами в представлениях	99
Отправка электронных писем с помощью Django	101
Отправка электронных писем в представлениях	106
Прорисовка форм в шаблонах	107
Создание системы комментариев	112
Разработка модели комментария	112
Добавление комментариев на сайт администрирования	114
Создание форм из моделей	116
Оперирование формами ModelForm в представлениях	116
Создание шаблонов комментарной формы	119
Добавление комментариев в представление детальной информации о посте	121
Добавление комментариев в шаблон детальной информации о посте	122
Дополнительные ресурсы	129
Резюме	130

Глава 3. Расширение приложения для ведения блога..... 131

Добавление функциональности тегирования	132
Извлечение постов по сходству	141
Создание конкретно-прикладных шаблонных тегов и фильтров	146
Реализация конкретно-прикладных шаблонных тегов	147
Создание простого шаблонного тега	147
Создание шаблонного тега включения	150
Создание шаблонного тега, возвращающего набор запросов	152
Реализация конкретно-прикладных шаблонных фильтров	154
Создание шаблонного фильтра для поддержки синтаксиса Markdown	154
Добавление карты сайта	159
Создание новостных лент для постов блога	164
Добавление полнотекстового поиска в блог	171
Установка базы данных PostgreSQL	172
Создание базы данных PostgreSQL	173
Выгрузка существующих данных	174
Переключение базы данных в проекте	174
Загрузка данных в новую базу данных	176
Простые операции поиска	177
Поиск по нескольким полям	177
Разработка представления поиска	178
Выделение основ слов и ранжирование результатов	182
Выделение основ слов и удаление стоп-слов на разных языках	183
Взвешивание запросов	184
Поиск по триграммному сходству	185

Дополнительные ресурсы.....	186
Резюме	187
Глава 4. Разработка социального веб-сайта.....	188
Создание проекта социального веб-сайта	189
Запуск проекта социального веб-сайта.....	189
Использование поставляемого с Django фреймворка аутентификации.....	191
Создание представления входа в систему	192
Использование встроенных в Django представлений аутентификации.....	199
Представления входа и выхода.....	199
Представления смены пароля.....	205
Представление сброса пароля.....	208
Регистрация пользователей и профили пользователей	216
Регистрация пользователя	216
Расширение модели пользователя.....	223
Установка библиотеки Pillow и раздача медиафайлов	224
Создание миграций для модели профиля	225
Использование конкретно-прикладной модели пользователя	231
Использование фреймворка сообщений	231
Разработка конкретно-прикладного бэкенда аутентификации	235
Предотвращение использования существующего адреса электронной почты	238
Дополнительные ресурсы.....	239
Резюме	240
Глава 5. Реализация социальной аутентификации	241
Добавление социальной аутентификации на сайт	242
Обеспечение работы сервера разработки по протоколу HTTPS.....	245
Аутентификация с учетной записью Facebook.....	248
Аутентификация с учетной записью Twitter	256
Аутентификация с учетной записью Google.....	268
Создание профиля пользователей, регистрирующихся посредством социальной аутентификации	277
Дополнительные ресурсы.....	279
Резюме	280
Глава 6. Распространение контента на веб-сайте	281
Создание веб-сайта для управления визуальными закладками	282
Разработка модели изображения	282
Создание взаимосвязей многие-ко-многим.....	284
Регистрация модели изображения на сайте администрирования	285
Отправка контента с других сайтов	286
Очистка полей формы.....	287
Установка библиотеки requests.....	288

Переопределение метода save() класса ModelForm	288
Разработка букмарклета с помощью JavaScript	293
Создание представления детальной информации об изображениях.....	306
Создание миниатюр изображений с помощью easy-thumbnails.....	309
Добавление асинхронных действий с помощью JavaScript.....	312
Загрузка JavaScript в DOM.....	314
Защита от подделки межсайтовых HTTP-запросов на JavaScript.....	315
Выполнение HTTP-запросов с помощью JavaScript	317
Добавление бесконечной постраничной прокрутки в список изображений.....	323
Дополнительные ресурсы	330
Резюме	331

Глава 7. Отслеживание действий пользователя

Разработка системы подписки.....	333
Формирование взаимосвязей многие-ко-многим с промежуточной моделью.....	333
Создание представлений списка и детальной информации для профилей пользователей	336
Добавление действий пользователя по подписке/отписке с помощью JavaScript	342
Разработка типового приложения для потока активности	345
Применение фреймворка contenttypes	346
Добавление обобщенных отношений в модели	347
Игнорирование повторных действий в потоке активности	351
Добавление действий пользователя в поток активности.....	352
Отображение потока активности.....	355
Оптимизация наборов запросов, предусматривающих связанные объекты	355
Применение метода select_related()	356
Применение метода prefetch_related()	357
Создание шаблонов действий.....	357
Использование сигналов для денормализации количественных данных.....	361
Работа с сигналами.....	361
Конфигурационные классы приложений	364
Использование меню отладочных инструментов Django.....	366
Установка меню отладочных инструментов Django.....	367
Панели меню отладочных инструментов Django	370
Команды меню отладочных инструментов Django	373
Подсчет просмотров изображений с помощью хранилища Redis.....	374
Установка платформы Docker	375
Установка хранилища Redis	375
Использование хранилища Redis вместе с Python	377
Хранение просмотров изображений в хранилище Redis	379
Хранение рейтинга в хранилище Redis.....	381
Следующие шаги с Redis	384
Дополнительные ресурсы	385
Резюме	385

Глава 8. Разработка интернет-магазина	387
Создание проекта интернет-магазина	388
Создание моделей каталога товаров	389
Регистрация моделей каталога на сайте администрирования	393
Формирование представлений каталога	395
Создание шаблонов каталога	397
Разработка корзины покупок	403
Использование сеансов Django	403
Настроечные параметры сеанса	404
Срок истечения сеанса	405
Хранение корзин покупок в сеансах	406
Создание представлений корзины покупок	410
Добавление товаров в корзину	411
Разработка шаблона отображения корзины	413
Добавление товаров в корзину	415
Обновление количества товаров в корзине	417
Создание процессора контекста для текущей корзины	418
Процессоры контекста	418
Установка корзины в контекст запроса	419
Регистрация заказов клиентов	421
Создание моделей заказа	422
Включение моделей заказа на сайт администрирования	424
Создание заказов клиентов	425
Асинхронные задания	431
Работа с асинхронными заданиями	431
Работники, очереди сообщений и брокеры сообщений	432
Использование Django с Celery и RabbitMQ	433
Отслеживание Celery с помощью инструмента Flower	440
Дополнительные ресурсы	443
Резюме	443
Глава 9. Управление платежами и заказами	444
Интеграция платежного шлюза	444
Создание учетной записи Stripe	445
Установка библиотеки Stripe	448
Добавление Stripe в проект	449
Формирование процесса платежа	450
Интеграция платежного инструмента Stripe Checkout	452
Тестирование процесса оформления заказа	459
Использование тестовых кредитных карт	461
Проверка платежной информации в информационной панели Stripe	463
Применение веб-перехватчиков для получения уведомлений о платежах	467
Создание конечной точки веб-перехватчика	467
Тестирование уведомлений веб-перехватчиков	472

Отсылки к платежам Stripe в заказах	475
Выход в прямой эфир.....	479
Экспорт заказов в CSV-файлы.....	480
Добавление конкретно-прикладных действий на сайт администрирования.....	480
Расширение сайта администрирования за счет конкретно-прикладных представлений.....	483
Динамическое генерирование счетов-фактур в формате PDF	488
Установка библиотеки WeasyPrint.....	489
Создание шаблона PDF	489
Прорисовка PDF-файлов.....	490
Отправка PDF-файлов по электронной почте.....	494
Дополнительные ресурсы	497
Резюме	498

Глава 10. Расширение магазина

Создание купонной системы	499
Разработка купонной модели	500
Применение купона к корзине.....	504
Применение купонов к заказам	512
Создание купонов для платежного инструмента Stripe Checkout	517
Добавление купонов в заказы на сайте администрирования и в счета-фактуры в формате PDF.....	520
Разработка рекомендательного механизма	523
Рекомендация товаров на основе предыдущих покупок.....	524
Дополнительные ресурсы	532
Резюме	532

Глава 11. Добавление интернационализации в магазин.....

Интернационализация в Django.....	535
Настроечные параметры интернационализации и локализации.....	535
Команды управления интернационализацией.....	536
Установка инструментария gettext	536
Как добавлять переводы в проект Django	537
Как Django определяет текущий язык	537
Подготовка проекта к интернационализации	538
Перевод исходного кода Python.....	539
Стандартные переводы.....	540
Ленивые переводы	540
Переводы с переменными.....	540
Формы множественного числа в переводах	541
Перевод собственного исходного кода.....	541
Перевод шаблонов	545
Шаблонный тег <code>{% trans %}</code>	546

Шаблонный тег {% blocktrans %}.....	546
Перевод шаблонов магазина.....	547
Использование интерфейса перевода Rosetta.....	551
Нечеткие переводы.....	554
Шаблоны URL-адресов для интернационализации.....	554
Добавление префикса языка в шаблоны URL-адресов.....	555
Перевод шаблонов URL-адресов.....	556
Переключение языка сайта.....	560
Перевод моделей с помощью модуля django-parler.....	562
Установка модуля django-parler.....	562
Перевод полей моделей.....	563
Интеграция переводов на сайт администрирования.....	565
Создание миграций для переводов моделей.....	566
Использование переводов с ORM-преобразователем.....	569
Адаптация представлений под переводы.....	570
Локализация формата.....	572
Использование модуля django-localflavor для валидации полей формы.....	573
Дополнительные ресурсы.....	575
Резюме.....	576

Глава 12. Разработка платформы электронного обучения.....

Настройка проекта электронного обучения.....	578
Раздача медиафайлов.....	579
Разработка моделей курса.....	580
Регистрация моделей на сайте администрирования.....	582
Использование фикстур с целью предоставления моделям первоначальных данных.....	583
Создание моделей полиморфного содержимого.....	586
Использование модельного наследования.....	587
Абстрактные модели.....	588
Наследование многотабличной модели.....	588
Прокси-модели.....	589
Создание моделей Content.....	589
Создание конкретно-прикладных модельных полей.....	592
Добавление упорядочивания в модули и объекты содержимого.....	594
Добавление представлений аутентификации.....	598
Добавление системы аутентификации.....	598
Создание шаблонов аутентификации.....	599
Дополнительные ресурсы.....	602
Резюме.....	603

Глава 13. Создание системы управления контентом.....

Создание CMS.....	604
Создание представлений на основе классов.....	605

Использование примесей для представлений на основе классов.....	605
Работа с группами и разрешениями.....	608
Ограничение доступа к представлениям, основанным на классах.....	610
Управление модулями курса и их содержимым	616
Использование наборов форм для модулей курса	616
Добавление содержимого в модули курса	621
Управление модулями и их содержимым.....	627
Переупорядочивание модулей и их содержимого.....	632
Использование примесей из модуля <code>django-braces</code>	633
Дополнительные ресурсы	641
Резюме	641

Глава 14. Прорисовка и кеширование контента..... 642

Отображение курсов.....	643
Добавление регистрации студентов	648
Создание представления регистрации студентов	649
Зачисление на курсы	651
Доступ к содержимому курсов	655
Прорисовка разных типов содержимого.....	659
Использование кеш-фреймворка.....	661
Доступные кеш-бэкенды	662
Установка резидентного кеш-сервера Memcached.....	663
Установка образа Memcached платформы Docker	663
Установка привязки Python к Memcached.....	663
Настроечные параметры кеша	664
Добавление кеш-сервера Memcached в проект	664
Уровни кеша	665
Использование низкоуровневого API кеша.....	665
Проверка запросов к кешу с помощью меню отладочных инструментов Django Debug Toolbar.....	667
Кеширование на основе динамических данных.....	671
Кеширование фрагментов шаблона	672
Кеширование представлений	673
Использование сайтового кеша	674
Использование кеш-бэкенда Redis	675
Отслеживание сервера Redis с помощью приложения Django Redisboard.....	676
Дополнительные ресурсы	678
Резюме	679

Глава 15. Разработка API..... 680

Разработка RESTful API	681
Установка фреймворка Django REST framework.....	681
Определение сериализаторов.....	682

Что такое парсер и рендерер.....	683
Разработка представлений списка и детальной информации	684
Потребление API	686
Создание вложенных сериализаторов	688
Разработка конкретно-прикладных представлений API	690
Обработка аутентификации.....	691
Добавление разрешений в представления	692
Создание наборов представлений и маршрутизаторов	694
Добавление дополнительных действий в наборы представлений	696
Создание конкретно-прикладных разрешений	697
Сериализация содержимого курса.....	697
Потребление RESTful API.....	700
Дополнительные ресурсы.....	703
Резюме	704

Глава 16. Разработка чат-сервера..... 705

Создание приложения для ведения чата.....	705
Реализация представления чат-комнаты	706
Реально-временной Django на основе Channels	709
Асинхронные приложения с использованием ASGI.....	710
Цикл запроса/ответа с использованием приложения Channels.....	710
Установка приложения-обертки Channels	712
Написание потребителя.....	714
Маршрутизация.....	716
Реализация WebSocket-клиента.....	717
Активирование канального слоя	723
Каналы и группы.....	724
Установка канального слоя с использованием Redis	724
Обновление потребителя с целью широковещательной рассылки сообщений	725
Добавление контекста в сообщения	730
Видоизменение потребителя с целью обеспечения полной асинхронности	733
Интеграция приложения для ведения чата с существующими представлениями	735
Дополнительные ресурсы.....	736
Резюме	737

Глава 17. Выход в прямой эфир..... 738

Создание производственной среды.....	739
Управление настроечными параметрами для нескольких сред	739
Настрочные параметры локальной среды.....	740
Запуск локальной среды.....	741
Настройки производственной среды	741

Использование инструмента Docker Compose.....	743
Установка инструмента Docker Compose	743
Создание файла Dockerfile.....	744
Добавление требующихся пакетов Python.....	745
Создание файла Compose платформы Docker	746
Конфигурирование службы PostgreSQL	749
Применение миграции базы данных и создание суперпользователя	752
Конфигурирование службы Redis.....	753
Раздача Django через WSGI и NGINX	754
Использование сервера приложений uWSGI.....	755
Конфигурирование сервера приложений uWSGI.....	756
Использование веб-сервера NGINX.....	757
Конфигурирование веб-сервера NGINX.....	758
Использование хост-имени.....	760
Раздача статических и мультимедийных ресурсов.....	761
Сбор статических файлов.....	761
Раздача статических файлов с помощью веб-сервера NGINX	762
Обеспечение защиты сайта с помощью SSL/TLS	764
Проверка готовности проекта к работе в производственной среде	764
Конфигурирование проекта Django под SSL/TLS.....	765
Создание SSL/TLS-сертификата	767
Конфигурирование веб-сервера NGINX под использование SSL/TLS	767
Перенаправление HTTP-трафика на HTTPS.....	770
Использование Daphne для приложения Django Channels.....	771
Использование безопасных соединений для веб-сокетов	773
Включение веб-сервера Daphne в конфигурацию веб-сервера NGINX.....	773
Создание конкретно-прикладных промежуточных программных	
компонентов.....	777
Создание поддоменного промежуточного компонента.....	778
Раздача нескольких поддоменов с помощью веб-сервера NGINX	780
Реализация конкретно-прикладных команд управления.....	780
Дополнительные ресурсы	783
Резюме	784
Предметный указатель.....	786

От издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг, мы будем очень благодарны, если вы сообщите о ней главному редактору по адресу dmkpress@gmail.com. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательство «ДМК Пресс» очень серьезно относится к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу электронной почты dmkpress@gmail.com.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Вступительное слово

Django: веб-фреймворк для перфекционистов, которые стараются придерживаться дедлайнов.

Мне нравится этот слоган, потому что бывает, что разработчики легко становятся жертвами перфекционизма, когда им приходится доставлять работоспособный исходный код точно в срок.

Есть много отличных веб-фреймворков, но иногда они требуют от разработчика слишком многого, например правильно структурировать проект, отыскивать нужные плагины и элегантно использовать существующие абстракции.

Django снимает бóльшую часть этой усталости от принятия решений и предоставляет вам гораздо больше. Но этот фреймворк также и большой, так что изучение его с нуля может оказаться непосильным.

Я изучил Django в 2017 году, в лоб, из необходимости, когда мы решили, что он будет ключевой технологией для нашей платформы программирования на Python (CodeChalleng.es). Я заставил себя изучить все тонкости, разрабатывая крупное практическое технологическое решение, которое с момента своего создания служило тысячам начинающих и опытных разработчиков Python.

Где-то в этом путешествии я подобрал раннюю редакцию данной книги. И она оказалось настоящей сокровищницей. Очень близкая нашим сердцам в Pybytes, она обучает фреймворку Django, помогая **разрабатывать** интересные приложения для решения практических задач. Мало того, Антонио привносит в работу много реального опыта и знаний, что проявляется в том, как он реализует эти проекты.

И Антонио никогда не упускает возможности представить менее известные функциональности, например оптимизацию запросов к базе данных с помощью Postgres, полезные пакеты, такие как `django-taggit`, социальную аутентификацию с использованием различных платформ, (модельные) менеджеры, шаблонные теги включения и многое другое.

В нескольких главах этого нового издания он даже добавил дополнительные схемы, изображения и примечания и перешел с jQuery на ванильный JavaScript (ну, не приятно ли?!).

Данная книга не только подробно описывает Django, используя чистые примеры исходного кода, которые хорошо объяснены, но и освещает смежные технологии, которые необходимы любому разработчику Django: Django REST framework, `django-debug-toolbar`, frontend/JS и, последнее, но не менее важное, Docker.

Что еще более важно, вы найдете целый ряд нюансов, с которыми столкнетесь, и лучших образцов практики, которые вам понадобятся, чтобы стать эффективным разработчиком Django в профессиональной среде.

Найти такой многогранный ресурс, как этот, непросто, и я хочу поблагодарить Антонио за всю ту тяжелую работу, которую он постоянно прилагает, чтобы поддерживать его в актуальном состоянии.

Для меня как разработчика Python, который часто использует Django, книга «Django в примерах» стала моим путеводителем, незаменимым ресурсом, который я хочу иметь под рукой. Всякий раз, когда я возвращаюсь к этой книге, я узнаю что-то новое, даже после того, как прочитал ее несколько раз и использую Django уже целых пять лет.

Если вы отправитесь в это путешествие, то будьте готовы к тяжелой практической работе. Ведь это практическое руководство, так что заварите себе хороший кофе и приготовьтесь полностью погрузиться в кучу исходного кода Django! Но именно так нам лучше всего учиться, верно? :)

– Боб Белдербос, соучредитель Pybytes

Об авторе

Антонио Меле – соучредитель и технический директор Numero, финтех-платформы, которая позволяет финансовым учреждениям строить, автоматизировать и масштабировать цифровые продукты для управления состояниями. Антонио также является техническим директором Echo Investing, цифровой инвестиционной платформы на базе искусственного интеллекта для рынка Великобритании.

Антонио разрабатывает проекты Django с 2006 года для клиентов из нескольких отраслей. В 2009 году Антонио основал Zenx IT, компанию-разработчик, специализирующуюся на разработке цифровых продуктов. Он работал техническим директором и технологическим консультантом во многих технологических стартапах и руководил коллективами разработчиков, создающими проекты для крупных цифровых компаний. Антонио получил степень магистра компьютерных наук в ICAI – Университете Понтифиии Комильяс (Pontificia Comillas), в котором он руководит стартапами, находящимися на ранней стадии. Его отец вдохновил его на увлечение компьютерами и программированием.

О рецензенте

Асиф Сайфуддин – разработчик программного обеспечения из Бангладеш. У него десятилетний профессиональный опыт работы с Python и Django. Помимо работы с различными стартапами и клиентами, Асиф также вносит свой вклад в некоторые часто используемые пакеты Python и Django. За его заслуженный вклад в разработку с открытым исходным кодом ныне он является основным сопровождающим такого ПО, как Celery, oAuthLib, PyJWT и auditwheel. Он также является сосопровождающим нескольких пакетов расширений Django Extensions и инструментария фреймворка Django REST framework. Кроме того, он является членом фонда **Django Software Foundation (DSF)** с правом голоса и членом фонда **Python Software Foundation (PSF)** с правом участия в разработке/управлении. Для многих молодых людей он является наставником в изучении Python и Django как в профессиональном, так и в личном плане.

*Особая благодарность **Карен Стингел** и **Исмиру Куллолли** за чтение и предоставление отзывов о книге с целью дальнейшего улучшения ее содержания. Мы очень ценим вашу помощь!*

Предисловие

Django – это веб-фреймворк Python с открытым исходным кодом, который способствует быстрой разработке и чистому, прагматичному дизайну. Он снимает большую часть хлопот, связанных с веб-разработкой, и обеспечивает относительно плавную кривую обучения для начинающих программистов. Django следует философии Python «батарейки включены в комплект», поставляя богатый и разнообразный набор модулей, которые решают распространенные задачи веб-разработки. Простота Django в сочетании с его мощными функциональными возможностями делает его привлекательным как для начинающих, так и для опытных программистов. Django был разработан с учетом простоты, гибкости, надежности и масштабируемости.

В настоящее время Django используется бесчисленными стартапами и крупными организациями, такими как Instagram, Spotify, Pinterest, Udemy, Robinhood и Coursera. Не случайно, что в течение последних нескольких лет в ежегодном опросе разработчиков Stack Overflow разработчики по всему миру неизменно выбирали Django в качестве одного из самых любимых веб-фреймворков.

Эта книга проведет вас через весь процесс разработки профессиональных веб-приложений с помощью Django. Книга посвящена объяснению механизмов работы веб-фреймворка Django путем написания нескольких проектов с нуля. В данной книге содержатся не только наиболее важные аспекты веб-фреймворка, но и объясняется, как применять Django к самым разнообразным реальным ситуациям.

В ней не только рассказывается о Django, но и представлены другие популярные технологии, такие как база данных PostgreSQL, резидентное хранилище Redis, очередь заданий Celery, брокер сообщений RabbitMQ и кеш-сервер Memcached. По ходу чтения книги вы научитесь интегрировать указанные технологии в свои проекты Django, чтобы создавать продвинутые функциональности и разрабатывать сложные веб-приложения.

Книга «*Django 4 в примерах*» проведет вас по всему процессу разработки практических приложений, по ходу дела решая распространенные задачи и внедряя лучшие образцы практики, используя пошаговый подход, которому легко следовать.

Прочитав эту книгу, вы получите хорошее представление о том, как работает Django и как разрабатывать полноценные веб-приложения на Python.

Для кого эта книга предназначена

Данная книга должна послужить руководством для программистов, недавно приступивших к работе с Django. Она предназначена для разработчиков со

знанием Python, которые хотят изучать Django прагматичным образом. Вы можете быть асолютным новичком в Django либо вы уже его немного знаете, но хотите извлечь из него максимальную пользу. Так или иначе, эта книга поможет вам освоить наиболее актуальные области веб-фреймворка, разрабатывая практические проекты с нуля. Вы должны быть знакомы с концепциями программирования, чтобы при чтении понимать излагаемый материал. В дополнение к базовым знаниям Python подразумевается некоторое предварительное знание HTML и JavaScript.

О чем эта книга рассказывает

Данная книга охватывает целый ряд тем разработки веб-приложений с помощью Django. Она поможет вам разработать четыре разных полнофункциональных веб-приложения, работа над которыми ведется на протяжении 17 глав:

- приложение для ведения блога (главы 1–3);
- веб-сайт по управлению визуальными закладками (главы 4–7);
- интернет-магазин (главы с 8 по 11);
- платформу электронного обучения (главы 12–17).

Каждая глава охватывает несколько функциональных возможностей Django.

Глава 1 «*Разработка приложения для ведения блога*» ознакомит с веб-фреймворком Django посредством создания приложения для ведения блога. Вы создадите базовые модели, представления, шаблоны и URL-адреса блога, чтобы отображать посты блога на страницах. Вы научитесь формировать наборы запросов QuerySet с помощью объектно-реляционного преобразователя Django (ORM) и сконфигурируете встроенный в Django сайт администрирования.

Глава 2 «*Усовершенствование блога за счет продвинутых функциональностей*» научит добавлять в свой блог постраничную разбивку и реализовывать представления на основе классов Django. Вы научитесь отправлять электронные письма с помощью Django, а также обрабатывать и моделировать формы. Вы также реализуете систему комментариев к постам блога.

Глава 3 «*Расширение приложения для ведения блога*» посвящена технике интегрирования сторонних приложений. В этой главе вы ознакомитесь с процессом создания системы тегирования и научитесь формировать сложные наборы запросов QuerySet, чтобы рекомендовать схожие посты. Здесь вы научитесь создавать собственные шаблонные теги и фильтры. Вы также узнаете, как использовать фреймворк карт веб-сайтов и создавать новостную RSS-ленту для своих постов. Вы завершите свое приложение для ведения блога, разработав поисковый механизм, в котором используются возможности полнотекстового поиска PostgreSQL.

Глава 4 «*Разработка социального веб-сайта*» посвящена объяснению техники разработки социального веб-сайта. Вы научитесь использовать встроенный в Django фреймворк аутентификации и расширите модель пользователя конкретно-прикладной моделью профиля. В этой главе вы научитесь

использовать фреймворк сообщений и разработаете конкретно-прикладной бэкенд аутентификации.

Глава 5 «*Реализация социальной аутентификации*» посвящена реализации социальной аутентификации с использованием учетных записей Google, Facebook и Twitter по стандарту OAuth 2 с помощью механизма Python Social Auth. Вы научитесь использовать расширения Django для работы сервера разработки по протоколу HTTPS и адаптировать конвейер социальной аутентификации под конкретно-прикладную задачу автоматизации создания профиля пользователя.

Глава 6 «*Распространение контента на веб-сайте*» научит технике трансформации социального приложения в веб-сайт по управлению визуальными закладками (CMS). Вы определите взаимосвязи многие-ко-многим в моделях и создадите букмарклет JavaScript, который будет интегрирован в ваш проект. В этой главе будет показано, как создавать миниатюры изображений. Вы также научитесь реализовывать асинхронные HTTP-запросы с использованием JavaScript и Django, и вы реализуете бесконечную постраничную прокрутку контента.

Глава 7 «*Отслеживание действий пользователя*» продемонстрирует технику разработки системы подписки для пользователей. Вы дополните свой веб-сайт по управлению визуальными закладками, создав приложение для слежения за потоками активности пользователей. Вы научитесь создавать обобщенные отношения между моделями и оптимизировать наборы запросов QuerySet, поработаете с сигналами и реализуете денормализацию. Вы научитесь использовать меню отладочных инструментов Django Debug Toolbar, чтобы получать соответствующую отладочную информацию. Наконец, интегрируете в свой проект быстрое хранилище данных Redis, чтобы вести подсчет просмотров изображений, и с помощью него создадите рейтинг наиболее просматриваемых изображений.

Глава 8 «*Разработка интернет-магазина*» посвящена обследованию техники создания интернет-магазина. Вы разработаете модели для каталога товаров и создадите корзину покупок, используя сеансы Django. Вы разработаете процессор контекста для корзины покупок и научитесь управлять заказами клиентов. В этой главе вы научитесь отправлять асинхронные уведомления с помощью очереди заданий Celery и брокера сообщений RabbitMQ. Вы также научитесь отслеживать Celery с помощью мониторингового инструмента Flower.

Глава 9 «*Управление платежами и заказами*» посвящена технике интеграции платежного шлюза в интернет-магазин. Вы выполните интеграцию платежного инструмента Stripe Checkout и будете получать асинхронные уведомления о платежах в своем приложении. Вы реализуете конкретно-прикладные представления на сайте администрирования, а также адаптируете сайт администрирования под конкретно-прикладную задачу экспорта заказов в CSV-файлы. Вы также научитесь динамически создавать счета-фактуры в формате PDF.

Глава 10 «*Расширение магазина*» научит технике создания купонной системы для применения скидок к корзине покупок. Вы обновите интеграцию платежного инструмента Stripe Checkout, чтобы имплементировать купон-

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru