

УДК 004.438Python
ББК 32.973.22
Л33

Ленц М.

Л33 Python: Непрерывная интеграция и доставка / пер. с англ. А. Е. Мамонова, Д. А. Беликова. – М.: ДМК Пресс, 2020. – 168 с.: ил.

ISBN 978-5-97060-797-8

Язык Python используется во многих областях – веб-разработке, науке о данных и машинном обучении, интернете вещей (IoT), автоматизации систем. Морис Ленц, блогер, архитектор программного обеспечения с большим опытом работы, досконально рассматривает возможности Python, упрощающие и повышающие эффективность разработки ПО. В книге представлены различные виды тестирования; показано, как настроить автоматизированные системы, которые выполняют эти тесты, и устанавливать приложения в различных средах контролируемым способом. Представленный материал позволит разработчику успешно решать технические проблемы, которые обычно скрываются в программном коде.

Издание предназначено для технических специалистов, занимающихся доставкой программного обеспечения: разработчиков, архитекторов, инженеров по релизу и DevOps-специалистов.

УДК 004.438Python
ББК 32.973.22

Authorized Russian translation of the English edition of Python Continuous Integration and Delivery: A Concise Guide with Examples ISBN 978-1-4842-4280-3 © 2019 by Moritz Lenz.

This translation is published and sold by permission of Packt Publishing, which owns or controls all rights to publish and sell the same.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN 978-1-4842-4280-3 (анг.)
ISBN 978-5-97060-797-8 (рус.)

© 2019 by Moritz Lenz
© Оформление, издание, перевод,
ДМК Пресс, 2020

Содержание

Об авторе	10
О техническом рецензенте	11
Благодарность	12
Введение	13
Глава 1. Автоматическое тестирование	17
1.1. Что же мы хотим от тестов	17
Быстрая обратная связь	17
Уверенность	18
Помощь в отладке	19
Справка по проектированию	19
Спецификация продукта	20
1.2. Недостатки тестов	20
Усилия	20
Дополнительный код для поддержки	21
Хрупкость	21
Ложное чувство безопасности	22
1.3. Характеристики хорошего теста	22
1.4. Виды тестов	22
Модульные тесты (Unit Tests)	23
Интеграционные тесты (Integration Tests)	24
Системные тесты (System Tests)	24
Дымовые тесты (Smoke Test)	25
Тесты производительности	26
1.5. Резюме	27
Глава 2. Модульное тестирование в Python	28
2.1. Отступление: виртуальное окружение	29
2.2. Начало работы с модульными тестами	29
Первый тест	30
Пишем больше тестов	32
Тестируем неудачный случай	33
2.3. Работа с зависимостями	34
Отделение логики от внешних зависимостей	34
Внедрение зависимостей для тестирования	37
Поддельные объекты (Мок-объекты)	39

Исправление	41
2.4. Разделение кода и тестов	42
Настройка Python Path	42
2.5. Подробнее о модульном тестировании и Pytest	43
2.6. Запуск юнит-тестов в чистой среде	44
2.7. Другой пример проекта: matheval	45
Логика приложения	46
2.8. Резюме	48

Глава 3. Непрерывная интеграция с Jenkins

3.1. Серверы непрерывной интеграции	50
3.2. Начало работы с Jenkins	51
Запуск Jenkins в Docker	51
Настройка исходного кода репозитория	52
Создание первого задания Jenkins	53
3.3. Экспорт дополнительных сведений о тесте в Jenkins	56
3.4. Шаблоны для работы с Jenkins	57
Ответственности	58
Уведомления	58
Ветви функций и пул-запросы (pull requests)	59
3.5. Другие показатели в Jenkins	59
Покрытие кода	59
Сложность	60
Стиль кода	60
Проверка архитектурных ограничений	60
3.6. Резюме	61

Глава 4. Непрерывная доставка

4.1. Причины для CD и автоматизированных развертываний	63
Экономия времени	63
Сокращение цикла релиза	63
Сокращение цикла обратной связи	64
Надежность релизов	65
Меньшие приращения облегчают торговлю	65
Больше архитектурной свободы	66
Передовые методы обеспечения качества	66
4.2. План для CD	67
Архитектура конвейера	67
Антишаблон: отдельные сборки для каждой среды	69
Все зависит от формата упаковки	70
Технология управления репозиториями Debian	71
Инструменты для установки пакетов	72
Управление конвейером	73
4.3. Резюме	74

Глава 5. Сборка пакетов	75
5.1. Создание tar-архива с исходным кодом	75
5.2. Упаковка с помощью dh-virtualenv	76
Начало работы с упаковкой	77
5.3. Файл debian/control	77
Направление процесса сборки	78
Объявление зависимостей Python	78
Сборка пакета	79
Создание пакета python-matheval	79
Компромиссы dh-virtualenv	80
5.4. Резюме	81
Глава 6. Распространение пакетов Debian	82
6.1. Сигнатуры	82
6.2. Подготовка репозитория	83
6.3. Автоматизация создания репозитория и добавления пакета	84
6.4. Обслуживание репозитория	86
Настройка компьютера для использования репозитория	87
6.5. Резюме	87
Глава 7. Развертывание пакетов	89
7.1. Ansible: основы	89
Соединения и файл инвентаризации	90
Модули	91
Модуль shell	92
Модуль copy	92
Модуль template	93
Модуль file	94
Модуль apt	94
Модули yum и zypper	95
Модуль package	95
Специализированные модули	95
Плейбуки	95
Переменные	98
Роли	100
7.2. Развертывание с помощью Ansible	102
7.3. Резюме	103
Глава 8. Виртуальная площадка для автоматизации развертываний	104
8.1. Требования и использование ресурсов	104
8.2. Знакомство с Vagrant	105
Настройка сети и Vagrant	106

8.3. Настройка машин.....	109
8.4. Резюме	114

Глава 9. Сборка в конвейере с помощью

Go Continuous Delivery	115
9.1. О Go Continuous Delivery	115
Устройство конвейера.....	116
Соответствие заданий агентам	116
Одно слово по поводу среды	117
Материалы	118
Артефакты	118
9.2. Установка	119
Установка сервера GoCD в Debian	119
Установка агента GoCD в Debian	120
Первый контакт с XML-конфигурацией GoCD.....	121
Создание SSH-ключа.....	122
9.3. Сборка в конвейере.....	123
Макет каталога	124
Этапы, задания, задачи и артефакты.....	124
Конвейер в действии.....	126
Старый номер версии – это не полезно.....	126
Создание уникальных номеров версий.....	127
Еще кое-что по поводу сборки.....	128
Подключение к GoCD	129
9.4. Резюме	130

Глава 10. Распространение и развертывание

пакетов в конвейере	131
10.1. Загрузка в конвейер	131
Учетные записи пользователей и безопасность	132
10.2. Развертывание в конвейере	134
10.3. Результаты	135
10.4. Проходя весь путь до реальных условий эксплуатации	136
10.5. Достижение разблокировано: базовая непрерывная доставка	138

Глава 11. Улучшаем конвейер.....

11.1. Откаты и установка определенных версий.....	139
Реализация	140
Давайте попробуем!.....	141
11.2. Проведение дымовых тестов в конвейере	142
Когда проводить такой тест?	142
Тестирование белого ящика	143
Образец дымового теста черного ящика.....	144

Добавление дымовых тестов в конвейер и роллинг-релизы	144
11.3. Шаблоны конфигурации	146
11.4. Как избежать шквала повторных сборок.....	148
11.5. Резюме	149

Глава 12. Безопасность

12.1. Опасности централизации	150
12.2. Время до выхода на рынок для исправлений безопасности.....	151
12.3. Аудит и спецификация ПО	152
12.4. Резюме	153

Глава 13. Управление состояниями.....

13.1. Синхронизация кода и версий базы данных	155
13.2. Разделение версий приложения и базы данных.....	155
Пример изменения схемы.....	156
Создание нового столбца, допускающего значение NULL.....	157
Миграция данных	159
Применение ограничений и очистка	159
Предварительные условия.....	160
Инструментарий	161
Структура	161
Единого решения не существует.....	162
13.3. Резюме	162

Глава 14. Выводы и перспективы

14.1. Что дальше?	163
Улучшенное обеспечение качества.....	163
Метрики	164
Автоматизация инфраструктуры.....	165
14.2. Заключение.....	167