

目录

第 1 章 入门准备

01 开篇词：你为什么要学 Python ？

02 我会怎样带你学 Python ？

03 让 Python 在你的电脑上安家落户

04 如何运行 Python 代码？

第 2 章 通用语言特性

05 数据的名字和种类—变量和类型

06 一串数据怎么存—列表和字符串

07 不只有一条路—分支和循环

08 将代码放进盒子—函数

09 知错能改—错误处理、异常机制

10 定制一个模子—类

11 更大的代码盒子—模块和包

12 练习—密码生成器

第 3 章 Python 进阶语言特性

13 这么多的数据结构（一）：列表、元祖、字符串

14 这么多的数据结构（二）：字典、集合

15 Python大法初体验：内置函数

16 深入理解下迭代器和生成器

17 生成器表达式和列表生成式

18 把盒子升级为豪宅：函数进阶

19 让你的模子更好用：类进阶

20 从小独栋升级为别墅区：函数式编程

32 自动化运维和测试

更新时间：2019-11-11 14:34:43



“

天才就是长期劳动的结果。

——牛顿”

6.5 自动化运维和测试路线图

自动化运维

我们说 Python 可以做自动化运维，但其实在这个领域中，Python 开发所占的比重并不是特别高。

对于一个成熟的可自动化运维的系统，主要的关注点有以下这些：

- CMDB（Configuration Management Database），设备数据
- 监控
- 批量运维
- 日志管理
- 持续集成与发布
- 安全漏洞扫描

这其中的每个点都是可以被部分甚至全部自动化的，可以看出所涉及的工作还是很多的，同时技术栈也五花八门。

一些比较典型的技术栈/工具如下：

<div>← 慕课专栏</div>		<div>≡ 你的第一本Python基础入门书 / 32 自动化运维和测试</div>	
<div>目录</div>		<div><div>• 监控：Zabbix</div><div>• 日志收集分析：ELK（Elasticsearch、Logstash、Kibana）</div><div>• 代码版本管理：Git</div><div>• 持续集成：Jenkins</div></div>	
<div>第 1 章 入门准备</div>		<div>对于上述工具的学习使用，可以学习实战课程《中小型企业通用自动化运维架构》，下图是其课程案例，都很实用。</div>	
<div>01 开篇词：你为什么要学 Python ？</div>		<div><div><div>✓ Ansible完成批量部署管理服务器</div><div>✓ Zabbix自定义监控报警</div><div>✓ Zabbix API自动获取存储信息</div><div>✓ 导入Ansible日志到ELK中</div><div>✓ 导入Zabbix日志到ELK中</div><div>✓ 自动化管理资产、跳板机</div><div>✓ Python Web项目自动部署</div></div></div>	
<div>02 我会怎样带你学 Python ？</div>		<div></div>	
<div>03 让 Python 在你的电脑上安家落户</div>		<div></div>	
<div>04 如何运行 Python 代码？</div>		<div></div>	
<div>第 2 章 通用语言特性</div>		<div></div>	
<div>05 数据的名字和种类—变量和类型</div>		<div></div>	
<div>06 一串数据怎么存—列表和字符串</div>		<div></div>	
<div>07 不只有一条路—分支和循环</div>		<div></div>	
<div>08 将代码放进盒子—函数</div>		<div></div>	
<div>09 知错能改—错误处理、异常机制</div>		<div></div>	
<div>10 定制一个模子—类</div>		<div></div>	
<div>11 更大的代码盒子—模块和包</div>		<div></div>	
<div>12 练习—密码生成器</div>		<div></div>	
<div>第 3 章 Python 进阶语言特性</div>		<div></div>	
<div>13 这么多的数据结构（一）：列表、元祖、字符串</div>		<div></div>	
<div>14 这么多的数据结构（二）：字典、集合</div>		<div></div>	
<div>15 Python大法初体验：内置函数</div>		<div></div>	
<div>16 深入理解下迭代器和生成器</div>		<div></div>	
<div>17 生成器表达式和列表生成式</div>		<div></div>	
<div>18 把盒子升级为豪宅：函数进阶</div>		<div></div>	
<div>19 让你的模子更好用：类进阶</div>		<div></div>	
<div>20 从小独栋升级为别墅区：函数式编程</div>		<div></div>	
<div></div>		<div>自动化测试</div>	
<div></div>		<div>对于测试而言，常规的测试方法是通过手工和肉眼来验证功能的正确性，当然我们也有更高效的方法，通过程序自动化地完成一部分这样的工作。比如，对于 Web 页面的测试，可以使用 Selenium 来模拟页面上的点击等操作，并自动进行正确性验证，以此达到自动化的目的。</div>	
<div></div>		<div>对于接口的测试而言，可以使用 requests 库和 Fiddler 抓包工具来验证分析。</div>	
<div></div>		<div>以下是一些常用的自动化测试知识：</div>	
<div></div>		<div><div>• HTML</div><div>• Selenium</div><div>• HTTP</div><div>• Python 的 requests 库</div><div>• Fiddler</div></div>	
<div></div>		<div>想要系统地进行学习，可以考虑这两个实战课程，对自动化测试的很多知识都有覆盖：</div>	
<div></div>		<div><div>• 《零基础入门 Python Web 自动化测试》</div><div>• 《Python接口自动化测试框架实战 从设计到开发》</div></div>	
<div></div>		<div><div>← 31 机器学习</div><div>33 写在最后 →</div></div>	
<div></div>		<div>精选留言 0</div>	

目录	
第 1 章 入门准备	
01 开篇词：你为什么要学 Python ？	
02 我会怎样带你学 Python ？	
03 让 Python 在你的电脑上安家落户	
04 如何运行 Python 代码？	
第 2 章 通用语言特性	
05 数据的名字和种类—变量和类型	
06 一串数据怎么存—列表和字符串	
07 不只有一条路—分支和循环	
08 将代码放进盒子—函数	
09 知错能改—错误处理、异常机制	
10 定制一个模子—类	
11 更大的代码盒子—模块和包	
12 练习—密码生成器	
第 3 章 Python 进阶语言特性	
13 这么多的数据结构（一）：列表、元祖、字符串	
14 这么多的数据结构（二）：字典、集合	
15 Python大法初体验：内置函数	
16 深入理解下迭代器和生成器	
17 生成器表达式和列表生成式	
18 把盒子升级为豪宅：函数进阶	
19 让你的模子更好用：类进阶	
20 从小独栋升级为别墅区：函数式编程	



目前暂无任何讨论

千学不如一看，千看不如一练