

接口自动化测试 持续集成

Postman + Newman +
Git + Jenkins + 钉钉

Storm 编著

详细讲解 HTTP 接口自动化测试方法

深入讲解 Postman 工具使用方法

全面解析 接口自动化测试持续集成实战项目

```
Integer euclidAlgorithm (int A, int B){  
    A=Math.abs(A);  
    B=Math.abs(B);  
    while (B!=0){  
        if (A>B) A=A-B;  
        else B=B-A;  
    }  
    return A;  
}
```

```
Integer euclidAlgorithm (int A, int B){  
    A=Math.abs(A);  
    B=Math.abs(B);  
    while (B!=0){  
        if (A>B) A=A-B;  
        else B=B-A;  
    }  
    return A;  
}
```

```
Integer euclidAlgorithm (int A, int B){  
    A=Math.abs(A);  
    B=Math.abs(B);  
    while (B!=0){  
        if (A>B) A=A-B;  
        else B=B-A;  
    }  
    return A;  
}
```

非
外
借

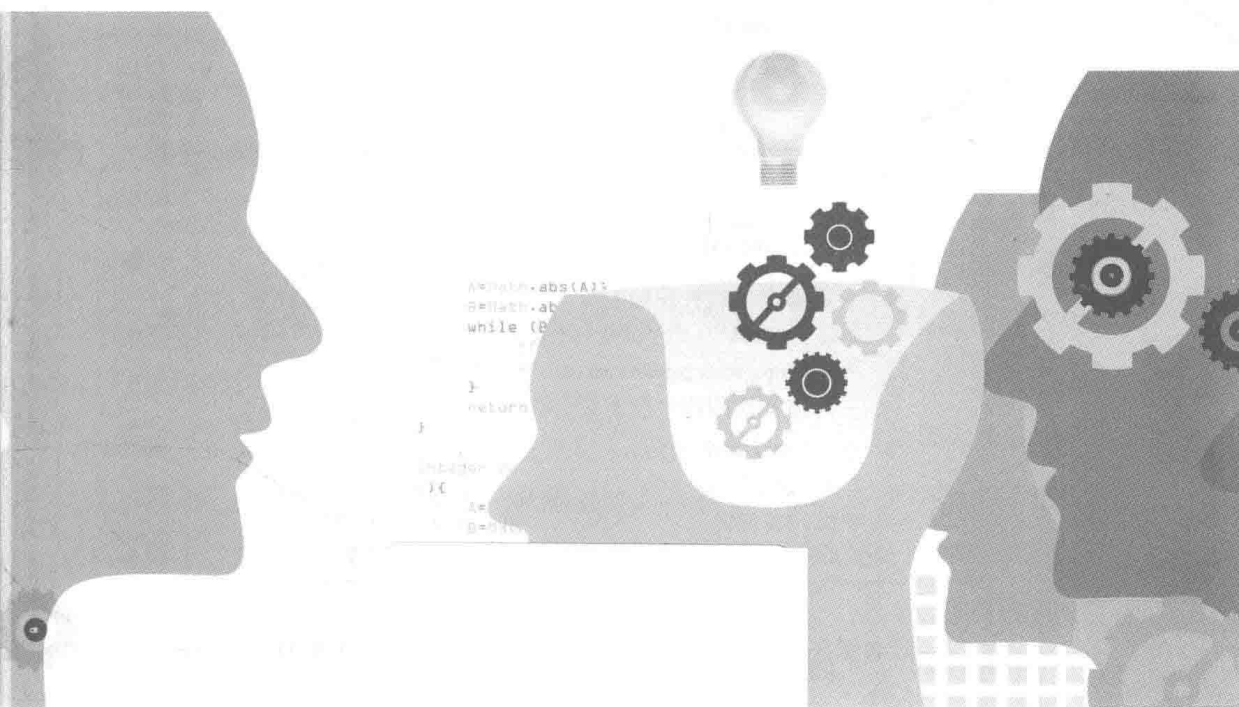
中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS

接口自动化测试 持续集成

**Postman + Newman +
Git + Jenkins + 钉钉**

—— Storm 编著 ——



人民邮电出版社
北京

图书在版编目 (CIP) 数据

接口自动化测试持续集成 : Postman+Newman+Git+
Jenkins+钉钉 / Storm 编著. — 北京 : 人民邮电出版社,
2019.2

ISBN 978-7-115-50341-1

I. ①接… II. ①S… III. ①软件工具—自动检测
IV. ①TP311.561

中国版本图书馆CIP数据核字(2018)第302946号

内 容 提 要

本书主要讲解接口自动化测试以及接口测试工具 Postman 的使用等相关内容。第 1 章介绍了接口测试基础知识,包括接口测试背景、接口基础知识、接口测试流程、接口文档系统等内容;第 2~5 章介绍了 Postman 工具的基本用法和高级用法,包括 Postman 工具介绍、Postman 基本操作、Postman 集合、Postman 脚本的应用等内容,帮助读者学习借助该工具完成接口测试的方法;第 6~7 章介绍了 Jenkins、GitHub 和钉钉等工具的基本用法,以及如何借助这些工具实现接口测试自动化和持续集成;第 8 章通过实际项目复习前 7 章所学内容,帮助读者从整体上深入领会接口自动化测试持续集成的内容精要。

不管是测试工程师、测试经理,还是其他对测试技术感兴趣的人员,均可以通过本书学习相关知识。

-
- ◆ 编 著 Storm
责任编辑 李 莎
责任印制 马振武
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市祥达印刷包装有限公司印刷
 - ◆ 开本: 700×1000 1/16
印张: 20.25
字数: 220 千字
印数: 1—2 400 册
- 2019 年 2 月第 1 版
2019 年 2 月河北第 1 次印刷
-



定价: 65.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

序

PREFACE

子龙前段时间突然找到我，想让我给他写序，我不禁心中感慨万千。想想子龙从我的课堂毕业到现在出版图书，还不到3年的时间！我欣赏子龙这些年在技术方面的学习和钻研精神，也钦佩他百忙之余还能将工作中的经验和知识总结出来，进行分享和传播。

近年来，测试技术飞速发展，公司对测试人员的要求也越来越高。测试已经从原本的纯业务测试发展到接口自动化测试、开发测试、性能测试等，总体呈现出复杂化、多元化的特点。其中，接口测试是走向自动化测试和性能测试的必经之路，也是目前测试人员经常接触到的工作之一。

接口可以分为内部接口和外部接口。内部接口是服务器端内部代码交互时用到的接口，如白盒测试就是测试内部接口；外部接口是客户端与服务器端交互时用到的接口，如HTTP接口。技术人员进行接口测试时，经常采用Postman、JMeter、SoapUI、Insomnia等工具，其中使用较为频繁的工具是Postman。

在百测（BestTest）的测试答疑群里，很多人提出类似“HTTP接口测试怎么做”“接口自动化怎么做”“接口自动化持续集成怎么做”等问题，这也是近年来技术人员在接口测试工作中遇到的亟待详尽解答的问题。子龙根据自己学习和工作的积累和心得，介绍了接口测试的基础知识、接口测试工具Postman的基本使用、Postman脚本的应用、接口自动化测试持续集成等焦点问题，并且通过项目实战，帮助读者融会贯通，学以致用。

我很高兴能为如此优秀的学生作序，也十分期待本书的出版，希望读者能通过学习本书使自己在接口测试及接口自动化测试方面的能力得到提升。同时，也希望技术人员能以本书为平台，更好地探讨接口测试的技术和方法，不断提高软件测试水平。

百测（BestTest）软件测试创始人、网易测试专家
安大叔

前言

FOREWORD

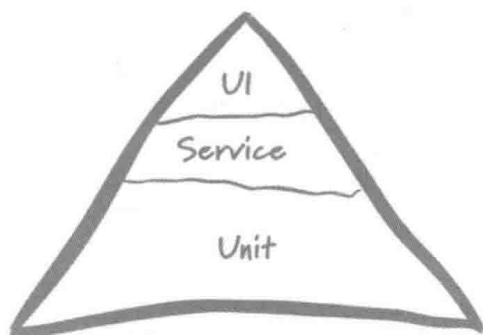
软件测试是软件开发的重要组成部分，是贯穿整个软件生命周期，对软件产品进行验证和确认的活动过程，其目的是尽早发现软件产品中存在的各种问题，如与用户需求、预先定义不一致等问题。随着技术的发展，测试从手工向自动化转变，从用户界面（User Interface, UI）层测试向单元测试靠拢。接下来，先回顾几个概念。

单元测试：对软件中的最小可测试单元进行检查和验证。具体来说就是开发者编写一小段代码，用于检验被测代码的一个很小的、很明确的功能是否正确。通常而言，一个单元测试是用于判断某个特定条件（或者场景）下某个特定函数的行为。

集成测试：它是在单元测试的基础上，将所有的软件单元按照概要设计规格说明的要求组装成模块、子系统或系统，并测试该过程中各部分工作是否达到或实现相应技术指标及要求。也就是说，在集成测试之前，单元测试应该已经完成。这一点很重要，因为如果不经过单元测试，那么集成测试的效果将会受到很大影响，并且会大幅增加软件单元代码纠错的代价。

系统测试：将需测试的软件，作为整个基于计算机系统的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素及环境结合在一起测试。系统测试的目的在于通过与系统的需求定义作比较，发现软件与系统定义不符合或与之矛盾的地方。

再来看看经典的测试分层金字塔图。



其中Unit代表单元测试，Service代表服务集成测试（或接口集成测试），UI代表页面系统测试。单元测试需要强大代码能力，很多测试人员还没有能力去执行，因此目前大多数公司还处于开发自测的阶段；随着开源UI自动化测试框架Selenium的发展，Web UI自动化测试近几年已趋于成熟（Appium是移动端UI自动化测试的代表框架），但其有3个明显的缺点：第一，UI测试介入测试时机较晚，修复发现的漏洞成本较大；第二，UI测试很难发现底层逻辑问题；第三，页面元素经常变换，导致自动化产出、投入比偏低；而这些恰恰是接口自动化测试所能解决的问题。关于接口自动化测试，目前在业内有两大类解决方案，一类是通过代码编写接口测试框架，实现接口自动化测试，其要求测试人员掌握扎实的编程基础；另一类是借助接口测试工具，配合Postman等集成工具实现接口自动化测试持续集成。前者更灵活，但后者的学习成本更低，适合新人上手。接口测试工具有很多，其中Postman安装简单、使用方便、功能强大，另外，这也是开发人员常用的接口调试工具，使用相同的工具测试出来的问题就更有说服力了。本书将借助该工具，带领大家了解接口测试持续集成的流程。

目前，很多项目都需要执行接口测试，很多读者也想了解接口测试方面的知识，但市面上与接口测试相关的书却很少，于是我根据自己的学习经验和工作经验写了这本书。

读者在了解了基本概念，理解了接口测试原理后，下载并安装Postman工具，跟随书中的示例进行练习，并把自己所学知识应用到目前从事的项目中。

由于本人水平有限，广大读者在学习过程中，如发现任何疑问，可发邮件至 apitest100@163.com，期待得到你的真挚反馈，让我们在技术之路上共同进步。

感谢天怡和其他编辑老师的耐心指导；感谢读者的信任；感谢BestTest测试培训机构提供的接口项目示例；感谢安大叔的教导；感谢家人的大力支持。

目录

CONTENTS

Chapter 1 接口测试基础知识

1.1 接口测试背景	2
1.2 接口基础知识	3
1.2.1 接口的定义	4
1.2.2 接口的分类	4
1.2.3 HTTP	5
1.2.4 HTTP请求	6
1.2.5 HTTP响应	9
1.3 接口测试流程	11
1.4 接口文档系统	13
1.4.1 ShowDoc简介	13
1.4.2 ShowDoc部署	14
1.4.3 接口测试文档示例	16

Chapter 2 初识Postman工具

2.1 安装Postman	19
2.2 Postman工具简介	24
2.3 Postman账号	37
2.4 Postman同步	39
2.5 Postman设置	41
2.6 发送第一个请求	45
2.7 将请求保存到集合	47

Chapter 3 Postman基本操作

3.1 API请求与响应	49
3.2 API授权、Cookies和证书	70
3.3 抓取HTTP请求	78
3.4 拦截器	80
3.5 代理	83
3.6 生成代码片段	87
3.7 Postman Echo	89

3.7.1 请求方法	89
3.7.2 Headers	95
3.7.3 认证方法	96
3.7.4 Cookies操作	99

Chapter **4** Postman集合

4.1 变量	102
4.1.1 变量的概念	102
4.1.2 管理环境变量	103
4.1.3 管理和查看全局变量	106
4.2 集合	108
4.2.1 创建集合	108
4.2.2 共享集合	111
4.2.3 管理集合	112
4.2.4 导入/导出文件	114
4.3 集合执行	118
4.3.1 集合运行参数	118
4.3.2 使用环境变量	121
4.3.3 使用数据文件	124
4.3.4 迭代运行集合	128
4.3.5 创建 workflow	130
4.3.6 分享集合运行结果	133
4.3.7 集合运行排错	134

Chapter 5 Postman脚本的应用

5.1 脚本介绍	139
5.2 预请求脚本	143
5.3 测试脚本	150
5.3.1 Tests基础知识	150
5.3.2 脚本示例	152
5.3.3 脚本进阶	162
5.3.4 pm对象	166
5.4 分支和循环	174
5.5 Postman Sandbox	175
5.6 Newman	178
5.6.1 安装Newman	178
5.6.2 Newman选项	180
5.6.3 集合运行排错	182
5.6.4 定制报告	183

Chapter 6 Jenkins、Git与钉钉

6.1 Jenkins	187
6.1.1 部署Jenkins	188
6.1.2 管理插件	194
6.1.3 创建项目	197
6.1.4 配置项目运行频率	201
6.1.5 配置邮件发送	202
6.2 Git	207
6.2.1 什么是Git	207
6.2.2 安装Git	209
6.2.3 Git常用命令	210
6.2.4 GitHub远程仓库	222
6.2.5 搭建GitLab	227
6.3 钉钉	228
6.3.1 钉钉简介	229
6.3.2 集成Jenkins	231

Chapter 7

接口测试持续集成

7.1 整合GitHub	235
7.2 整合Jenkins	238
7.3 整合钉钉	240

Chapter 8

项目接口测试实战

8.1 项目介绍	243
8.1.1 项目部署	243
8.1.2 查看接口文档	245
8.2 编写接口测试文档	250
8.2.1 编写接口测试计划	251
8.2.2 编写接口测试用例	257
8.3 执行接口测试	267
8.3.1 从Postman执行接口测试	267
8.3.2 从Newman执行接口测试	306
8.4 接口自动化测试持续集成实战	307
8.4.1 通过GitHub维护测试文件	308
8.4.2 配置Jenkins自动化测试任务	309
8.4.3 接收自动化测试结果	311

Chapter

1

接口测试基础知识

越来越多的公司在招聘测试人员时，要求应聘的人员从事过接口测试相关工作。那么，什么是接口测试？为什么要进行接口测试？如何进行接口测试？本章将详细讲解接口测试基础知识，相信学完本章内容，读者将会得到这些问题的答案。

1.1 接口测试背景

应用程序编程接口（Application Programming Interface, API）是近年来最流行的技术之一，强大的Web应用程序和领先的移动应用程序都离不开后端强大的API。API技术的应用给系统开发带来了便利，但也对测试人员提出了更高的要求，如何以一种有效的方式测试这些API，并且确保它们按照预期运行，是目前测试人员面临的主要问题。

大多数互联网公司和团队都在实施敏捷开发项目，在敏捷开发产品的生命周期中，测试人员需要一种简单、快捷的方法自动化测试API，甚至希望能够监控生产环境API（线上环境）的实时状况。

本书将介绍一套完整的接口自动化测试解决方案，帮助读者快速了解和掌握该方案，并顺利应用到实际项目中。

近年来项目开展接口测试的比例逐年上升，如图1-1所示。

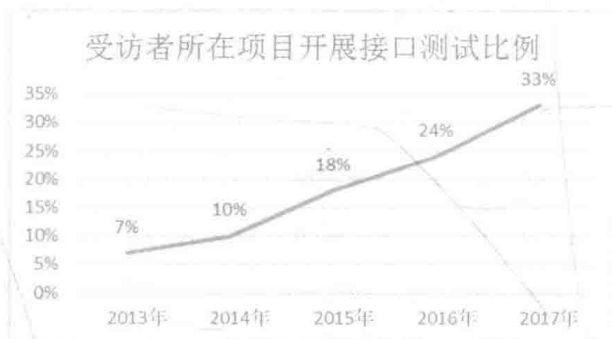


图1-1 受访者所在项目开展接口测试比例统计

图1-1反映出从事接口测试工作的人员越来越多，应用接口测试的项目也越来越多。

1. 接口测试的必要性

目前，软件系统的复杂度不断上升，传统功能测试的片面性及滞后性导致测试成本急剧增加，且测试效率大幅度下降，仅靠功能测试已难以保证项目质量及进度。

而接口测试的应用，恰好能使测试团队更早、更深入地介入项目，这样测试人员在项目初期就能发现系统深层次的问题，降低问题修复的时间成本。同时，由于接口的变更概率远远小于用户界面（User Interface,UI）的变更概率，因此，接口测试自动化维护成本比UI自动化维护成本更低，接口测试相对更容易实现自动化测试持续集成，且可以减少回归测试的人力与时间成本，缩短测试周期，满足后端快速发布版本的需求。持续集成是接口测试低成本、高收益的根源，是接口测试的灵魂。

2. 接口测试的原理

测试人员借助工具模拟客户端向服务器端发送请求报文，服务器端接收请求报文后，对相应的报文做出处理并向客户端返回应答，工具模拟客户端接收应答，然后测试人员检查应答是否准确，这就是接口测试的原理。

3. 接口测试的范围

关于接口测试的范围，主要从以下两方面进行介绍。

（1）是否所有的接口都需要测试？

随着系统复杂度越来越高，接口越来越多，想完全覆盖所有接口是一件很困难的事情。通常情况下，主要测试最外层的两类接口：数据进入系统的接口（调用外部系统的参数为本系统使用）和数据流出系统的接口（验证系统处理后的数据是否正常）。

（2）被测接口需要测试哪些方面？

测试人员需要关注被测接口的功能是否实现、性能是否达标、安全性是否满足，重点关注数据的交换、传递、处理次数以及控制管理过程。

1.2 接口基础知识

进行接口测试，首先需要了解什么是接口，下面将详细讲解接口的基础知识。

1.2.1 接口的定义

两个不同的系统或者一个系统中两个不同的功能，它们之间相互连接的部分称为接口。

在软件测试中，常说的接口一般有两种：图形用户接口（Graphical User Interface, GUI），它是人与程序的接口；应用程序编程接口（Application Program Interface, API），本书中提到的接口特指API。

API是一组定义、程序及协议的集合，API可实现计算机软件之间的相互通信。API的一个主要功能是提供通用功能集。程序员通过使用API函数开发应用程序，从而可以避免编写无用程序，减轻编程任务。很多公司将开发岗位分为前端工程师和后端工程师，他们之间相互配合完成工作。一般来说，他们会协商接口的定义方式，其中一方定义接口（一般由后端工程师定义接口），另一方来调用接口，以实现预期功能。

前后端分离是近年来Web应用开发的一个发展趋势。这种模式具有以下优势。

- ① 后端工程师不用精通前端技术（如HTML、JavaScript或CSS），只专注于数据的处理，对外提供API即可。
- ② 前端工程师的专业性越来越强，其通过API获取数据，并专注于页面设计。
- ③ 前后端分离可扩大接口的应用范围，开发的接口可以应用到Web页面上，也可以应用到App上。

1.2.2 接口的分类

依据所遵循协议的不同，常见接口可以分为以下3类。

（1）HTTP接口，它是基于超文本传输协议（HyperText Transfer Protocol, HTTP）开发的接口，但不能排除没有使用其他协议。

（2）Web Service接口，它是系统对外的接口，比如你要从别的网站或服务上获取资源，一般来说，别人不会把数据库共享给你，他们会提供一个他们写好的方

法，让你用来获取数据，你使用他们写好的方法就能引用他们提供的接口，从而达到同步数据的目的。

(3) RESTful接口，简称为REST，其描述了一个架构样式的网络系统，核心是面向资源。REST专门针对网络应用设计和开发方式，降低开发的复杂性，提高系统的可伸缩性。

基于浏览器/服务器模式 (Browser/Server, B/S) 的软件系统接口大多数为HTTP接口，因此，本书将重点介绍HTTP接口的测试方法。要测试HTTP接口，首先需要了解HTTP、HTTP请求和响应的相关知识。

1.2.3 HTTP

HTTP是应用最为广泛的网络协议之一，所有的万维网文件都必须遵守这个标准。设计HTTP的目的是为了提供一种发布和接收HTML页面的方法。1960年德特·纳尔逊 (Ted Nelson) 构思了一种通过计算机处理文本信息的方法，并称之为超文本 (HyperText)，这成为HTTP标准架构的发展根基。Ted Nelson组织协调万维网联盟 (World Wide Web Consortium, W3C) 和互联网工程任务组 (The Internet Engineering Task Force, IETF) 共同合作研究，最终发布了一系列请求评议 (Request For Comments, RFC)，其中著名的RFC 2616定义了HTTP 1.1。

HTTP的主要特点可概括为如下几点。

(1) 支持客户端/服务器模式。客户端向服务器请求服务时，只需传送请求方法和路径。常用的请求方法有GET、POST。每种方法规定的客户端与服务器联系的类型不同。

(2) 简单。由于HTTP简单，服务器的程序规模小，因而通信速度比较快。

(3) 灵活。HTTP允许传输任意类型的数据对象。正在传输的类型由Content-Type加以标记。

(4) 无连接。限制每次连接，使其只处理一个请求。服务器处理完客户端的请求，并收到客户端的应答后，即断开连接。采用这种方式可以节省传输时间。