功能的自动化测试（GUI、接口）

**课程适合人群：**

软件自动化测试工程师

**本课程重点解答&解决：**

* 掌握单元测试框架
* 学习自动化测试理念
* 介绍自动化测试中的测试数据
* 学习基于GUI的自动化测试技术——Selenium
* 学习基于GUI的自动化测试技术——Cypress
* 学习基于GUI的自动化测试技术——Playwright
* 学习接口自动化测试

**本课程亮点+核心价值介绍：**

* 详细介绍单元测试框
* 详细介绍自动化测试中的测试数据
* 接口测试增加接口安全测试

**课程大纲：**

课长7天

1单元测试框架

 1.1 unittest

 1.1.1 unittest自动化测试框架介绍

 1.1.2 以案例开始

 1.1.3 unittest的装饰器

 1.1.4 通过parameterized的参数化

 1.1.5 测试覆盖率分析工具

 1.1.6 断言

 1.1.7 批量运行并且产生报告

 1.2 pytest

 1.2.1 pytest概述

 1.2.2 pytest安装

 1.2.3 以案例开始

 1.2.4 pytest设计规则

 1.2.5 装饰器

 1.2.6 pytest 常用命令行选项

 1.2.7 pytest 运行并发测试

 1.2.8 pytest特有的参数化功能

 1.2.9 配合allure生成漂亮的pytest测试报

2 软件测试自动化测试方法

 2.1测试自动化的内涵

 2.2测试自动化的基本结构

 2.3测试自动化的原理和方法

 2.4基于脚本的自动化测试

 2.5自动化测试流程

 2.6测试自动化普遍存在的问题

3 软件测试自动化测试实践

 3.1什么类型的功能测试应该采用自动化方法

 3.2如何处理复杂业务流程的自动化测试

 3.4如何准备自动化功能测试数据

 3.4什么原因导致无法识别对象

 3.5如何在运行过程中防止程序中断

 3.6自动测试的选择（案例讨论）

 3.7自动化测试的持续集成对软件测试的重大价值

 3.8自动化测试与回归测试的集成对软件测试的重大价值

4 自动化测试工具的分类和选择

 4.1测试工具的分类和选择

 4.2测试工具的主流产品介绍

 4.3自动测试的选择

 4.4自动化工具原理

 4.5存在的问题分析

5 自动化测试和软件质量分析报告

 5.1软件产品的质量度量

 5.2评估系统自动化测试的覆盖程度

 5.3自动化测试缺陷分析方法

6自动化测试数据准备

 6.1测试数据的创建方法

 6.2测试数据的创建时机

 6.从测试数据1.0到3.0

7 自动化功能测试工具: Selenium/WebDriver的使用

 7.1 Selenium原理

 7.1.1 基本原理

 7.1.2 环境配置

 7.1.3 驱动器版本

 7.2自动化功能测试工具: Selenium/WebDriver的使用

 7.2.1 使用浏览器

 7.2.2获取控件

By.ID

By.NAME

By.CLASS\_NAME

By.TAG\_NAME

By.LINK\_TEXT

By.PARTIAL\_LINK\_TEXT

By.CSS\_SELECTOR

By.XPATH

定位一组元素

 7.2.3 操作

浏览器操作

基本操作

鼠标事件

等待

显示等待

隐私等待

强制等待

多表单切换

多窗口切换

下拉条(Select)操作

单选框(Radio)操作

复选框(Checkbox)操作

警告框处理

cookie操作

调用HTM5 Video

截图

数据驱动

 7.2.4 Selenium Grid

基本Grid

在Docker下运行Grid

 7.2.5程序封装优化

异常优化

Retry优化

 7.2.6 POM介绍

8 Cypress

 8.1安装

 8.2 第一个代码

 8.3定位元素

 8.3.1为Cypress专门提供data-\*属性

 8.3.2基本元素定位

Cypress默认使用CSS Selector元素定位

通过CSS Selector元素定位

使用cypress.$(selector) 元素定位

 8.3.3辅助元素定位

HTML代码

find()

get()

contain()

children()

parents()

parent()

siblings()

first()

last()

next()

nextAll()

nextUtil()

prev()

prevAll()

 8.4操作元素

8.4.2访问网站

8.4.3刷新页面

8.4.4强制刷新页面

8.4.5设置窗口大小

8.4.6前进、后退

8.4.7判断元素是否可见

8.4.8判断元素是否存在

8.4.9判断元素是否不存在

8.4.10条件判断

8.4.11获取元素属性

8.4.12点击元素

8.4.13双击元素

8.4.14右击元素

8.4.15提交表单

8.4.16往输入框输入文本元素

8.4.17聚焦元素

8.4.18失去焦点

8.4.19清空元素

8.4.20检查多选或者单选框被选中

8.4.21检查多选或者单选框未被选中

8.4.22在检查之前忽略错误检查

8.4.23 select 选项框

8.4.24滑内嵌滚动条滑动

8.4.25浏览器自带的滚动条

8.4.26在元素上触发事件

 8.5钩子函数(Hook)

 8.6等待

 8.7断言

8.7.1隐式断言

 should()

 and()

8.7.2显示断言

 expect()

 assert()

8.8iframe切换与Windows切换

8.8.1 iframe切换

8.8.2 Windows切换

8.9参数化

8.9.1变量参数化测试

8.9.2使用fixture测试外部数据

8.9.3使用外部json文件进行测试

8.9.4优化

8.10 POM（Page Object Method）

8.10.1定义页面文件

 登录页面文件

商品页面文件

8.10.2主文件

8.11 命令行运行测试

8.11.1运行

8.11.2 cypress open

8.11.3 cypress run

8.11.4 cypress verify

8.11.5 cypress version

8.12 cypress支持Allure

8.13 cypress 与Jenkins集成

9 Playwright

9.1 Playwright介绍

9.2安装

9.3 Playwright命令

9.4录制

9.4.1开始录制

9.4.2异步代码与同步代码

9.4.3录制脚本进行优化

9.5设置

9.5.1忽略HTTPS错误

9.5.2设置自定义视口大小

* 在conftest.py中配置
* 在测试用例中配置

9.5.3模拟手机浏览器

9.6 定位元素

9.6.1定位函数

* page.get\_by\_role()
* page.locator()
* page.get\_by\_text()
* page.get\_by\_label()
* page.get\_by\_placeholder()
* page.get\_by\_alt\_text()
* page.get\_by\_title()
* page.get\_by\_test\_id()

9.6.2 CSS与xpath

* CSS
* xpath

9.6.3 playwright定位类型

* xpath
* css
* text

9.7操作

9.7.1基本操作

* 跳转到一个新的链接
* 获取元素焦点
* 输入文本
* 设置复选框取消或选中
* 取消已选中复选框取
* 获取元素属性值
* 获取内部文本
* 获取内部HTML
* 获取文本内容
* 设置select下拉选项
* 调度事件
* >>组合表达式
* 执行JavaScript语句

9.7.2 鼠标操作

* 元素定位
* 获取元素位置及大小
* 鼠标移动
* 按下鼠标
* 释放鼠标
* 鼠标双击
* 鼠标悬停
* 鼠标点击
* 案例

9.7.3 键盘操作

* 模拟键盘输入字符操作
* 单元素键盘事件操作
* 上传文件
* 聚焦元素
* 拖拽操作

9.7.4 等待

* 等待事件触发
* 等待JavaScript函数触发
* 等待事件被触发
* 自动等待
* 强制等待

9.7.5其他操作

* iframe操作
* 窗口切换
* 操作video
* 降低执行速度
* 快速截图
* 复用 Cookies 等认证信息
* 监听事件
* 拦截更改网络请求
* 灵活设置代理

9.7.6 批量执行

9.8 断言

9.8.1断言内容

* 页面URL断言
* 页面Title断言
* 获取文本内容断言
* 获取内部文字断言
* 获取属性值断言
* 复选框断言
* js表达式断言
* 内部HTML断言
* 元素可见断言
* 启动状态断言
* 直接对比断言
* 自定义断言

9.8.2对象断言

* page属性
* element属性
* locator属性

9.8.3 from playwright.sync\_api import expect 断言方法

9.9 与pytest结合

9.9.1 安装

9.9.2 代码

9.9.3参数介绍

* 运行测试
* 开启页面运行
* 指定浏览器运行
* 开启页面运行并指定浏览器运行
* 指定两个浏览器运行
* 用慢动作运行测试
* 开启记录每次测试的轨迹
* 开启每次测试录制视频
* 开启每次测试后是否自动截屏

9.9.4通过浏览器跳过测试

9.9.5在特定浏览器上运行

9.10 POM & FOM

9.10.1 POM

* 登录
* 查看商品
* 放入购物车

9.10.2 FOM

* 保留文件
* 添加flowObject.py
* 改写TestEbusiness.py

9.11数据驱动

9.11.1通过mark.parametrize参数化

9.11.2 通过fixtures参数化

9.11.3 yarm格式文件

* 相关插件
* yarm文件
* 代码

9.11.4 Excel格式文件

* 相关插件
* Excel
* 代码

9.11.5 CSV格式文件

* 相关插件
* CSV文件
* 代码

9.11.6 json格式文件

* 相关插件
* json文件
* 代码

9.11.7 XML格式文件

* 相关插件
* XML文件
* 代码

9.12 生成Allure报告

9.12.1改写pageObject.py

9.12.2 改写flowObject.py

9.12.3 改写TestEbusiness.py

9.12.4 查看报告

9.13与Jenkins集成

10 接口自动化测试

 10.1 Requests类介绍

 10.2 基于requests类的Python接口自动化测试

 10.2.1原始阶段

 10.2.2对CSRF的校验

 10.2.3使用excel作为数据驱动

 10.2.4数据驱动优化

 10.2.5相对复杂的接口测试

 10.3 Cypress接口测试

 10.3.1 cy.request参数

 10.3.2 Get方法

 10.3.3 Post方法

 10.3.4 案例

Get接口测试案例

Post接口测试案例

 10.4接口性能测试

 10.5接口安全测试