

Curso de Automação de Testes de Frontend Web com Python e Selenium

...

Aula 04

Paulo Oliveira

Agenda

- Plugins Python: pytest e unittest
- Projeto Prático 1

Frameworks de Teste Python



Conhecendo o Unittest

- Introdução ao Unittest
 - Unittest é um framework de testes unitários incluído na biblioteca padrão do Python.
 - Permite a criação de testes seguindo a abordagem de testes unitários clássicos.
- Estrutura de um teste típico
 - Setup
 - Preparação do ambiente de teste. Executado antes de cada método de teste.
 - Test Cases
 - Métodos que realmente testam partes do seu código.
 - Cada método testa uma funcionalidade específica.
 - Teardown
 - Limpeza após cada método de teste.
 - Utilizado para fechar conexões de banco de dados, excluir arquivos temporários, etc.

Conhecendo o Pytest

- Introdução ao Pytest
 - Pytest é um framework moderno para testes em Python que se destaca por sua simplicidade e poderosa funcionalidade.
 - Não requer uma classe para definir testes, permitindo a escrita de códigos de teste mais limpos e concisos.
- Estrutura de um teste típico
 - Setup
 - Utiliza fixtures para preparação do ambiente de teste.
 - Funções que iniciam com `test_` para realizar testes.
 - Teardown
 - Utiliza fixtures para limpeza após execução dos testes.

Instalação

- unittest
 - Não é necessário instalar nada adicional se você já tem o Python instalado, pois o unittest faz parte da biblioteca padrão do Python.
- pytest
 - Você precisa instalar o pytest se ainda não o fez. Você pode fazer isso executando o seguinte comando no terminal ou prompt de comando: *pip install pytest*

Projeto Prático 1

- Site: <https://www.saucedemo.com/>
- Automatizar o seguinte fluxo:
 - Logar no site com o usuário standard
 - Adicionar os 6 produtos no carrinho
 - Conferir que no carrinho tem a badge com 6 produtos
 - Entrar no carrinho
 - Remover um dos produtos do carrinho
 - Conferir que no carrinho tem a badge com 5 produtos
 - Clicar no botão Checkout
 - Preencher os dados solicitados e clicar em Continue
 - Clicar no botão Finish
 - Conferir a mensagem “Thank you for your order!”
- Automatizar mais 3 fluxos desejados

Projeto Prático 1

- No final, o objetivo é que vocês tenham:
 - Pelo menos 4 testes criados (um dos testes deve ser o fluxo que está detalhado no slide anterior + 3 a sua escolha)
 - Todos os testes devem ser construídos e executados usando o pytest
 - Tentem usar o máximo possível de locators diferentes, não façam apenas usando os locators mais simples, como id, classes etc, tentem usar mais css selector e xpath para praticarem
 - Comunicação:
 - Quem tiver dúvidas sobre o projeto, pode enviar no grupo
 - Quem concluir o projeto, manda um print do console com o resultado dos testes no grupo
 - Quem concluir o projeto, pode me enviar o repositório ou o arquivo (quem não tiver no github ainda) no privado, que posso dar uma olhada

Perguntas?

- Momento aberto para dúvidas



Próximos Passos

- O que esperar da próxima aula
 - Dúvidas sobre o projeto
 - Implementação do Projeto Prático 1
- Ganhe visibilidade todos os dias

Curso de Automação de Testes de Frontend Web com Python e Selenium

...

Aula 04

Paulo Oliveira