



{ UNIT
TEST }

Introdução ao Unittest em Python



unittest





O que é unittest?

O unittest é uma biblioteca do Python usada para testar partes do seu código automaticamente, para garantir que tudo está funcionando como esperado.

Com ele, você pode criar testes de unidade — ou seja, testar funções específicas individualmente



- Framework padrão: Incluído na biblioteca padrão do Python, sem dependências externas.
- Baseado em classes: Organiza testes em classes e métodos para melhor estrutura.
- Asserts poderosos: Verificações para validar resultados esperados e falhas.
- Suites e agrupamentos: Permite criar coleções lógicas de testes para execução conjunta.



Resposta:

Por que Testar seu Código?



Real Python

- Detectar bugs precocemente: Reduz custos corrigindo erros no início do desenvolvimento.
- Facilitar refatoração: Permite melhorias no código com segurança e confiança.
- Garantir regressão: Assegura que novas funcionalidades não quebrem o código existente.
- Documentação viva: Testes descrevem o comportamento esperado do software.



Conceitos Chave do Unittest

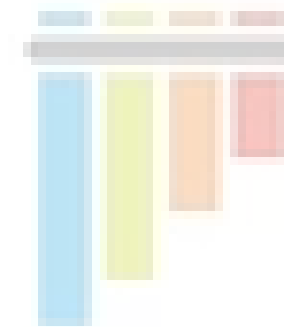
- TestCase: Classe base para criar casos de teste específicos.
- assert: Métodos para validar condições, como assertEquals e assertTrue.
- setUp e tearDown: Preparação e limpeza antes e depois de cada teste.
- TestSuite e TestRunner: Agrupa e executa os testes, exibindo relatórios.



unittest vs pytest: A Batalha dos Testes



unittest



unittest

- Framework padrão do Python, estruturado e baseado em classes.
- Ideal para quem busca estabilidade e integração com demais ferramentas.

pytest

- Framework popular, fácil de usar, com sintaxe simples e poderosa.
- Suporta fixtures e mocks avançados para testes mais flexíveis.

