

# Exercício 1

Integrante: Wesley Bernardes (020321)

**1) Qual tipo de teste é responsável por verificar unidades individuais de código (como funções ou métodos)?**

R: Teste de Unidade

**2) Qual tipo de teste avalia a interface entre diferentes módulos ou serviços para garantir que funcionem corretamente juntos?**

R: Teste de Integração

**3) O que é teste de caixa branca (white box)?**

R: Teste que analisa a estrutura interna e lógica do código.

**4) Qual é um exemplo de teste de caixa preta?**

R: Testar se um campo de login rejeita senhas incorretas.

**5) O que é o ciclo de vida do software?**

R: As etapas desde o planejamento até a manutenção do software.

**6) Em qual fase do ciclo de vida os testes de regressão são mais usados?**

R: Manutenção

**7) O que é um teste de software e por que ele é importante?**

R: Teste de software é o processo de verificar se um sistema ou componente atende aos requisitos especificados. Ele é importante para garantir a qualidade e confiabilidade do software.

**8) Qual símbolo usamos para marcar uma função como teste no pytest?**

R: test\_

**9) Qual comando usamos para executar um arquivo de teste chamado test\_login.py?**

R: pytest test\_login.py

**10) Qual comando usamos no terminal para executar todos os testes de um projeto com pytest?**

R: pytest

**11) O que faz o comando pytest -v?**

R: Executa os testes em modo verboso (mostra detalhes).

**12) Explique a diferença entre teste de unidade e teste de integração, dando um exemplo de cada usando pytest.**

R: **Teste de unidade** verifica partes isoladas do código, como funções ou métodos individuais, sem dependências externas.

```
# arquivo: test_math.py

def soma(a, b):
    return a + b

def test_soma():
    assert soma(2, 3) == 5
```

**Teste de integração** verifica se diferentes partes do sistema funcionam corretamente em conjunto, como o fluxo entre banco de dados e regras de negócio.

```
# arquivo: test_login.py

class FakeDB:
    def __init__(self):
```

```
self.users = {"user@example.com": "senha123"}

def autenticar(self, email, senha):
    return self.users.get(email) == senha

def login(db, email, senha):
    if db.autenticar(email, senha):
        return "Login bem-sucedido"
    return "Credenciais inválidas"

def test_login_sucesso():
    db = FakeDB()
    assert login(db, "user@example.com", "senha123") == "Login bem-suce
dido"

def test_login_falha():
    db = FakeDB()
    assert login(db, "user@example.com", "senhaErrada") == "Credenciais i
nválidas"
```

### **13) Qual é um objetivo do teste de caixa preta?**

R: Validar se o software atende aos requisitos funcionais.