

Desafio Final

```
def calcular_total(valor, tipo_cliente):
    if tipo_cliente == "vip":
        desconto = valor * 0.15
    else:
        desconto = valor * 0.05

    valor = valor - desconto

    total = valor + (valor * 0.07)
    print("O total da compra é:", total)
    print("Obrigado por comprar!")
    print("Volte sempre!")
    print("-----")

calcular_total(1000, "vip")
calcular_total(500, "normal")
```

Resultado

```
O total da compra é: 909.5
Obrigado por comprar!
Volte sempre!
-----
O total da compra é: 508.25
Obrigado por comprar!
Volte sempre!
-----
```

Problemas identificados

1. **Número mágico:** `0.15`, `0.05` e `0.07`
2. **Código repetido:** vários `print()` iguais em todas as chamadas.
3. **Falta de clareza e reutilização:** cálculo de desconto e imposto dentro da mesma função.

Refatorado

```
TAXA_IMPOSTO = 0.07
DESCONTO_VIP = 0.15
DESCONTO_NORMAL = 0.05

def obter_desconto(tipo_cliente):
    descontos = {"vip": DESCONTO_VIP, "normal": DESCONTO_NORMAL}
    return descontos.get(tipo_cliente, 0)

def calcular_total(valor, tipo_cliente):
    desconto = valor * obter_desconto(tipo_cliente)
    valor_liquido = valor - desconto
    total = valor_liquido * (1 + TAXA_IMPOSTO)
    return total

def exibir_mensagem(total):
    mensagens = [
        f"O total da compra é: R${total:.2f}",
        "Obrigado por comprar!",
        "Volte sempre!",
        "-----"
    ]
    for msg in mensagens:
        print(msg)

# Teste prático
clientes = [
    (4657, "vip"),
    (2657, "normal"),
    (1349, "outro")
]

for valor, tipo in clientes:
    total = calcular_total(valor, tipo)
    exibir_mensagem(total)
```

Benefícios

- Código **mais claro**, **reutilizável** e **fácil de manter**.
- Segue o padrão **PEP8**.
- Pronto para **adicionar testes automatizados (TDD)** depois.

