

## re\_S22\_A2\_SL10 desenvolvimento software

### Entrega no AVA

Nome: \_\_\_\_\_ Turma: \_\_\_\_\_

## Atividade 1: Análise de código e proposição de melhorias

**Objetivo:** Analisar o código de referência de uma aplicação responsável pelo sistema de controle de acesso em um escritório e propor melhorias que possam ser aplicadas.

### Código em Python

```
def pode_acessar(cargo, dia_semana):
    dias uteis = ["segunda-feira", "terça-feira", "quarta-feira", "quinta-feira",
"sexta-feira"]
    acesso_liberado = False

    # Gerentes têm acesso irrestrito
    if cargo.lower() == "gerente":
        acesso_liberado = True
    # Analistas têm acesso somente em dias úteis
    elif cargo.lower() == "analista" and dia_semana.lower() in dias_uteis:
        acesso_liberado = True
    # Estagiários têm acesso somente em dias úteis
    elif cargo.lower() == "estagiário" and dia_semana.lower() in dias_uteis:
        acesso_liberado = True

    return acesso_liberado

# Teste do programa
cargo = input("Digite o cargo do funcionário: ")
dia_semana = input("Digite o dia da semana: ")

if pode_acessar(cargo, dia_semana):
    print("Acesso Permitido.")
else:
    print("Acesso Negado.")
```

## Detalhes do exercício

### Estrutura lógica

- Entradas: cargo do funcionário e dia da semana.
- Processamento:
  - Define os dias úteis.
  - Verifica se o cargo é "gerente". Se for, o acesso será liberado. Se não for "gerente", verifica se o cargo é "analista" ou "estagiário" e se o dia é útil. Se ambas as condições forem verdadeiras, o acesso será liberado.
  - Em outros casos, o acesso será negado.
- Saída: informa se o acesso será permitido ou negado.

Tempo estimado: 30 minutos

## Lista de materiais

- Computador com internet
- Caderno para anotações
- Uma caneta

## Procedimento experimental

1. Após testar o código, anote os resultados e discuta com o grupo possíveis melhorias ou alternativas.
2. Anote a resolução abaixo e envie pelo AVA.

---

---

---

---

---