

Aluno: Danilo Saulo Gomes Sampaio
Curso: Sistema de Informação
Período: 2º período
Disciplina: Programação II
Atividade: Lista - Funções em Python

1. Função de Login

```
def login(email, senha):  
    if email == 'admin' and senha == 'admin':  
        return True  
    else:  
        return False
```

2. Função de Desconto

```
def aplicar_desconto(preco):  
    return preco - (preco * 0.10)
```

3. Sistema de Bolsa

```
# funcoes_bolsa.py  
def verificar_acao(valor):  
    if valor > 150.99:  
        return "Ação está cara! Compre com cuidado."  
    else:  
        return "Ação está barata!"  
  
# main.py  
from funcoes_bolsa import verificar_acao  
  
valor = float(input("Digite o valor da ação: "))  
print(verificar_acao(valor))
```

4. Maior ou Menor de Idade

```
def verificar_idade(idade):  
    if idade >= 18:  
        return "Maior de idade"  
    else:  
        return "Menor de idade"
```

5. Cálculo e Classificação de IMC

```
def calcular_imc(peso, altura):  
    return peso / (altura ** 2)  
  
def classificar_imc(imc):  
    if imc < 18.5:  
        return "Abaixo do peso"  
    elif imc < 25:  
        return "Peso normal"  
    else:  
        return "Acima do peso"
```

6. Preço com Imposto

```
def adicionar_imposto(preco):  
    return preco + (preco * 0.15)
```

Try ... Except

1. Divisão com tratamento de erro

```
try:  
    a = int(input("Digite o primeiro número: "))  
    b = int(input("Digite o segundo número: "))
```

```
    print("Resultado:", a / b)
except ZeroDivisionError:
    print("Erro: divisão por zero!")
```

2. Entrada de número com verificação

```
try:
    numero = int(input("Digite um número: "))
    print("Você digitou:", numero)
except ValueError:
    print("Erro: valor inválido!")
```

3. Dois tipos de erro

```
try:
    a = int(input("Digite um número: "))
    b = int(input("Digite outro número: "))
    print(a / b)
except ValueError:
    print("Erro: valor inválido!")
except ZeroDivisionError:
    print("Erro: divisão por zero!")
```

4. Função dividir(a, b)

```
def dividir(a, b):
    try:
        return a / b
    except ZeroDivisionError:
        return "Erro: divisão por zero"
```