



escola
britânica de
artes criativas
& tecnologia

Python para análise de dados



PROGRAMAÇÃO FUNCIONAL



GUIA DA AULA 1



Explore a função lambda

Paradigmas de
programação

Definição

Funções de
alta ordem



Acompanhe aqui
os temas que
serão tratados
na videoaula



1. Paradigmas de programação

Estilos de programação.

- Imperativa;
- Funcional;
- Orientada a objetos.

O Python é uma linguagem multiparadigma:



Este módulo é sobre programação funcional, que é o estilo que busca manipular dados com muitas e pequenas funções.



2. Definição

Função lambda é uma função anônima (sem nome) com bloco de código super enxuto e que pode ser salva em uma variável. Em geral é utilizada com outros métodos funcionais como `map`, `filter` e `reduce`.

```
variavel = lambda params: expressão
```



2. Definição

Exemplo: Função lambda para extrair provedor de e-mail:

```
In [ ]: extrair_provedor_email = lambda email: email.split(sep='@')[-1]
```

```
In [ ]: email = 'andre.perez@gmail.com'
print(email)

provedor_email = extrair_provedor_email(email)
print(provedor_email)
```



2. Definição

Exemplo: Função lambda com estruturas condicionais:

```
In [ ]: numero_e_par = lambda numero: True if numero % 2 == 0 else False
```

```
In [ ]: numeros = range(0, 10)
```

```
In [ ]: for numero in numeros:
    if numero_e_par(numero) == True:
        print(f'O número {numero} é par!')
```



3. Funções de alta ordem

São funções que recebem outras funções para parâmetro ou retornam outra função.

Exemplo: Juros compostos dinâmico.

Definição

```
In [ ]: def retorno(juros: float):
        return lambda investimento: investimento * (1 + juros)
```



3. Funções de alta ordem

Instanciação

```

In [ ]:
retorno_5_porcento = retorno(juros=0.05)
retorno_10_porcento = retorno(juros=0.10)
    
```

```

In [ ]:
valor_final = retorno_5_porcento(investimento=1000)
print(valor_final)

valor_final = retorno_10_porcento(investimento=1000)
print(valor_final)
    
```



3. Funções de alta ordem

Uso

```

In [ ]: anos = 10
        valor_inicial = 1000
        valor_final = valor_inicial

        for ano in range(1, anos+1):
            valor_final = retorno_5_porcento(investimento=valor_final)

        valor_final = round(valor_final, 2)
        print(valor_final)
  
```

```

In [ ]: anos = 10
        valor_inicial = 1000
        valor_final = valor_inicial

        for ano in range(1, anos+1):
            valor_final = retorno_10_porcento(investimento=valor_final)

        valor_final = round(valor_final, 2)
        print(valor_final)
  
```

