

Vetores


Aula 3: Acesso a Elementos de Vetores

Nesta terceira aula sobre vetores, vamos aprofundar nosso conhecimento sobre o acesso a elementos de vetores em Python. Além disso, exploraremos técnicas avançadas de iteração através de vetores.

Acesso a Elementos de Vetores

Como mencionado anteriormente, você pode acessar elementos de vetores usando índices. Os índices em Python começam em 0. Veja como acessar elementos e realizar algumas operações:

python

 Copy code

```
frutas = ["maçã", "banana", "laranja", "morango", "uva"]

# Acessando o primeiro elemento (maçã)
primeira_fruta = frutas[0]

# Acessando o segundo elemento (banana)
segunda_fruta = frutas[1]


# Modificando um elemento (alterando "maçã" para "pêra")
frutas[0] = "pêra"

# Obtendo o último elemento (uva)
ultimo_elemento = frutas[-1]
```

Slicing (Fatiamento)

Além de acessar elementos individuais, você pode realizar fatiamento (slicing) em vetores em Python. Isso permite extrair partes específicas de um vetor. Veja como fazer isso:

python

 Copy code

```
numeros = [1, 2, 3, 4, 5]

# Fatiando o vetor para obter os três primeiros elementos
tres_primeiros = numeros[0:3] # Isso retorna [1, 2, 3]


# Fatiando o vetor para obter uma parte específica
parte_do_meio = numeros[1:4] # Isso retorna [2, 3, 4]

# Fatiando o vetor para obter elementos do terceiro em diante
do_terceiro_em_diante = numeros[2:] # Isso retorna [3, 4, 5]
```

Comprimento de Vetores

Para obter o comprimento de um vetor (ou seja, quantos elementos ele contém), você pode usar a função `len()` em Python:

python

 Copy code

```
frutas = ["maçã", "banana", "laranja", "morango", "uva"]
tamanho = len(frutas) # Isso retornará 5
```