Algumas operações em Listas com Python

capitais = ["Salvador", "Recife", "Fortaleza", "Natal", "São Luís", "Maceió"]

A partir dessa lista, podemos realizar várias operações comuns em listas.

**Adicionar um elemento**: Podemos adicionar outra capital à lista usando o método append.

capitais.append("Aracaju")

# Lista atualizada: ['Salvador', 'Recife', 'Fortaleza', 'Natal', 'São Luís', 'Maceió', 'Aracaju']

1. **Inserir um elemento em uma posição específica**: Podemos usar o método insert para inserir uma capital em uma posição específica.

capitais.insert(3, "João Pessoa")

# Lista atualizada: ['Salvador', 'Recife', 'Fortaleza', 'João Pessoa', 'Natal', 'São Luís', 'Maceió']

1. **Remover um elemento**: Podemos remover uma capital específica usando o método remove.

capitais.remove("São Luís")

# Lista atualizada: ['Salvador', 'Recife', 'Fortaleza', 'João Pessoa', 'Natal', 'Maceió']

1. **Remover um elemento por índice**: Podemos usar o método pop para remover uma capital por índice.

capitais.pop(2)

# Lista atualizada: ['Salvador', 'Recife', 'João Pessoa', 'Natal', 'Maceió']

1. **Acessar um elemento por índice**: Podemos acessar uma capital específica usando seu índice.

capital = capitais[1]

# Resultado: 'Recife'

1. **Substituir um elemento**: Podemos substituir uma capital em um índice específico.

capitais[1] = "Teresina"

# Lista atualizada: ['Salvador', 'Teresina', 'João Pessoa', 'Natal', 'Maceió']

1. **Obter o comprimento da lista**: Podemos usar a função len para obter o número de capitais na lista.

tamanho = len(capitais)

# Resultado: 5

1. **Ordenar a lista**: Podemos ordenar a lista de capitais em ordem alfabética usando o método sort.

capitais.sort()

# Lista atualizada: ['João Pessoa', 'Maceió', 'Natal', 'Salvador', 'Teresina']

1. **Reverter a lista**: Podemos reverter a ordem dos elementos na lista usando o método reverse.

capitais.reverse()

# Lista atualizada: ['Teresina', 'Salvador', 'Natal', 'Maceió', 'João Pessoa']

1. **Verificar se um elemento está na lista**: Podemos usar o operador in para verificar se uma capital está na lista.

existe = "Fortaleza" in capitais

# Resultado: False

Essas operações são básicas e úteis para manipular listas em Python.

operações básicas que você pode realizar em uma lista (array) em Python, usando a lista de capitais como exemplo.

**Lista de capitais**

capitais = ["Salvador", "Recife", "Fortaleza", "Natal", "São Luís", "Maceió"]

**1. Verificar se a lista está vazia**

Para verificar se a lista está vazia, podemos simplesmente usar uma comparação com uma lista vazia ou usar a função len.

# Método 1: Comparação com lista vazia

if not capitais:

print("A lista está vazia")

else:

print("A lista não está vazia")

# Método 2: Usando len

if len(capitais) == 0:

print("A lista está vazia")

else:

print("A lista não está vazia")

**2. Obter o tamanho da lista**

Para obter o tamanho da lista, usamos a função len.

tamanho = len(capitais)

print("O tamanho da lista é:", tamanho)

**3. Acessar o primeiro elemento**

Para acessar o primeiro elemento da lista, usamos o índice 0.

python

Copiar código

primeira\_capital = capitais[0]

print("A primeira capital é:", primeira\_capital)

**4. Acessar o último elemento**

Para acessar o último elemento da lista, podemos usar o índice -1.

ultima\_capital = capitais[-1]

print("A última capital é:", ultima\_capital)

**5. Adicionar um elemento ao final da lista**

Para adicionar um elemento ao final da lista, usamos o método append.

capitais.append("Aracaju")

print("Lista atualizada:", capitais)

**6. Inserir um elemento em uma posição específica**

Para inserir um elemento em uma posição específica, usamos o método insert.

capitais.insert(2, "Teresina")

print("Lista atualizada:", capitais)

**7. Remover um elemento específico**

Para remover um elemento específico, usamos o método remove.

capitais.remove("Recife")

print("Lista atualizada:", capitais)

**8. Remover um elemento por índice**

Para remover um elemento por índice, usamos o método pop.

elemento\_removido = capitais.pop(3)

print("Elemento removido:", elemento\_removido)

print("Lista atualizada:", capitais)

**9. Limpar a lista**

Para remover todos os elementos da lista, usamos o método clear.

capitais.clear()

print("Lista após limpar:", capitais)

**10. Verificar se um elemento está na lista**

Para verificar se um elemento está na lista, usamos o operador in.

existe = "Fortaleza" in capitais

print("Fortaleza está na lista?", existe)

**11. Contar quantas vezes um elemento aparece na lista**

Para contar quantas vezes um elemento aparece na lista, usamos o método count.

capitais.append("Salvador")

contagem = capitais.count("Salvador")

print("Salvador aparece", contagem, "vezes na lista")

**12. Obter o índice de um elemento**

Para obter o índice da primeira ocorrência de um elemento, usamos o método index.

indice = capitais.index("Natal")

print("O índice de Natal é:", indice)

Essas são algumas das operações básicas que você pode realizar com listas em Python. Elas cobrem desde a verificação de elementos até a manipulação direta dos itens na lista.