PLN na Ciência Política

Professora: Lorena G. Barberia

DCP-USP

lorenabarberia@usp.br

Universidade de São Paulo

Tópicos da Aula

Universidade de São Paulo

Aula de Hoje

- Nesta aula, exploraremos conceitos fundamentais de Python.
- Aprenderemos sobre operações em listas, operadores lógicos e estruturas condicionais e de repetição.

Universidade de São Paulo

O que são Operações em Listas?

- As listas são estruturas de dados em Python que podem armazenar vários elementos.
- Operações em listas envolvem a manipulação, adição, remoção e processamento de elementos em uma lista.
- Essas operações são essenciais para criar programas dinâmicos e interativos.

Universidade de São Paulo

Operações Básicas em Listas

- Adição de elementos: `append()`, `insert()`, `extend`, `+`;
- Remoção de elementos: `remove()`, `pop()`, `del`;
- Acesso a elementos: `index()`, `[índice]`;
- Tamanho da lista: `len()`;
- Ordenação: `sort()`, `sorted()`;
- Fatiamento: `[índice]`, `[índice1:índice2]`

Universidade de São Paulo

Introdução aos Operadores Lógicos

- Operadores lógicos são utilizados para criar expressões lógicas em programação.
- Eles permitem avaliar condições e tomar decisões com base em valores verdadeiros ou falsos.
- Em Python, temos três operadores lógicos principais: `and`, `or` e `not`.

Universidade de São Paulo

Operador Lógico ‘and‘

- O operador lógico ‘and‘ retorna verdadeiro (`True`) se ambas as condições forem verdadeiras.
- Caso contrário, retorna falso (`False`).

- **Exemplo:**

  - `a = 5`
  - `b = 10`
  - `a > 0 and b < 15` → `resultado = True`

Universidade de São Paulo

Operador Lógico ‘or‘

- O operador lógico ‘or‘ retorna verdadeiro (`True`) se pelo menos uma das condições for verdadeira.
- Caso contrário, retorna falso (`False`).

- **Exemplo:**
  - `x = 4`
  - `y = -2`
  - `x < 5 or y > 0` → `resultado = True`

Universidade de São Paulo

Operador Lógico ‘not‘

- O operador lógico ‘not‘ inverte o valor de uma expressão lógica.
- Se a expressão é verdadeira (`True`), o ‘not‘ a torna falsa (`False`) e vice-versa.

- **Exemplo:**
  - `condicao = False`
  - `not condicao` → `resultado = True`

Universidade de São Paulo

A Importância da *Indentation* (Recuo)

- Em Python, o recuo é fundamental para definir blocos de código.
- O recuo é a forma como Python delimita estruturas de controle, como loops e condicionais.
- Diferentemente de outras linguagens que usam chaves, o recuo em Python determina a hierarquia do código.

Universidade de São Paulo

Estruturas de Controle com Indentação

- Loops e condicionais em Python são definidos pelo recuo.
- Exemplo de um loop ”for”:
  - `for i in range(5):`
  - `  print(i)`
- Exemplo de uma estrutura condicional:
  - `if x > 10:`
  - `  print("x é maior que 10")`

Universidade de São Paulo

Evitando Erros de *Indentation*

- Erros de *indentation* são comuns para iniciantes em Python.
- Misturar espaços e `tabs` pode causar erros.
- Mantenha consistência no recuo ao longo do código.
- Use um editor que exibe o recuo corretamente.

Universidade de São Paulo

Introdução às Estruturas Condicionais

- As estruturas condicionais permitem tomar decisões em um programa.
- Em Python, as estruturas condicionais são implementadas com os comandos `if`, `elif` e `else`.
- Elas executam diferentes blocos de código com base em condições específicas.

Universidade de São Paulo

Comando `if` Simples

- O comando `if` permite executar um bloco de código se uma condição for verdadeira.

- **Sintaxe:**

  - `if condição:`
  - `    Código a ser executado se a condição for verdadeira`

Universidade de São Paulo

Comando `if` com `else`

- O comando `else` permite executar um bloco de código quando a condição do `if` não for verdadeira.

- **Sintaxe:**

```
if condição:
    Código a ser executado se a condição for verdadeira
else:
    Código a ser executado se a condição não for verdadeira
```

Universidade de São Paulo

Comando `if` com `elif`

- O comando `elif` é usado para testar condições adicionais após um `if`.
- Pode haver múltiplos `elif` após o `if`, e apenas um bloco será executado.

- **Sintaxe:**

```
if (condição):
    Código a ser executado se a condição for verdadeira
elif (outra condição):
    Código a ser executado se a outra condição for verdadeira
else:
    Código a ser executado se nenhuma condição for
verdadeira
```

Universidade de São Paulo

Introdução às Estruturas de Repetição

- As estruturas de repetição permitem executar um bloco de código várias vezes.
- Em Python, temos duas estruturas de repetição principais: `for` e `while`.
- Elas são usadas quando precisamos executar um conjunto de instruções repetidamente.

Universidade de São Paulo

Estrutura de Repetição `while`

- O `while` é usado para executar um bloco de código enquanto uma condição for verdadeira.

- **Sintaxe:**

  - `while condição:`
  - `    Código a ser executado enquanto a condição for verdadeira`

Universidade de São Paulo

Estrutura de Repetição `for`

- O `for` é usado para iterar sobre uma sequência (lista, tupla, string, etc.).
- **Sintaxe:**
  - `for variável in sequência:`
  - `    Código a ser executado em cada iteração`

Universidade de São Paulo

Considerações finais

- Nesta aula, exploramos operações em listas, operadores lógicos, estruturas de repetição e condicionais em Python;
- Se algo não ficar claro em algum comando ou objeto de Python, reforçamos que é importante sempre procurar a documentação do python no site oficial `https://docs.python.org/3/`;
- Na próxima aula, discutiremos como criar nossas próprias funções em Python, como utilizar os módulos e funções criados por outros, e introduziremos o Numpy, uma das principais *libraries* de matemática da linguagem.

Universidade de São Paulo

Laboratório

Agora, vocês irão se reunir em grupos e trabalharão no laboratório da aula de hoje (Lab_Aula2.ipynb). Os laboratórios estão na nossa pasta do drive. Pedimos que façam uma cópia do arquivo em uma subpasta. Nomeiem essa subpasta de acordo com o grupo (sobrenome1_sobrenome2_lab2) e coloquem as informações dos alunos em uma célula de *Markdown* no início do laboratório (Nome, NUSP, curso e se é da graduação ou pós).

Universidade de São Paulo

Dúvidas?

Universidade de São Paulo