

AULA 01

LINGUAGEM DE PROGRAMAÇÃO II - LP212

TIPOS DE DADOS BÁSICOS

PROFA. ANA PAULA MÜLLER GIANCOLI

Instituto Federal de São Paulo - IFSP

Campus Bragança Paulista

AGENDA

Codificação

Comentários

Tipos de Dados

Variáveis

Atribuição

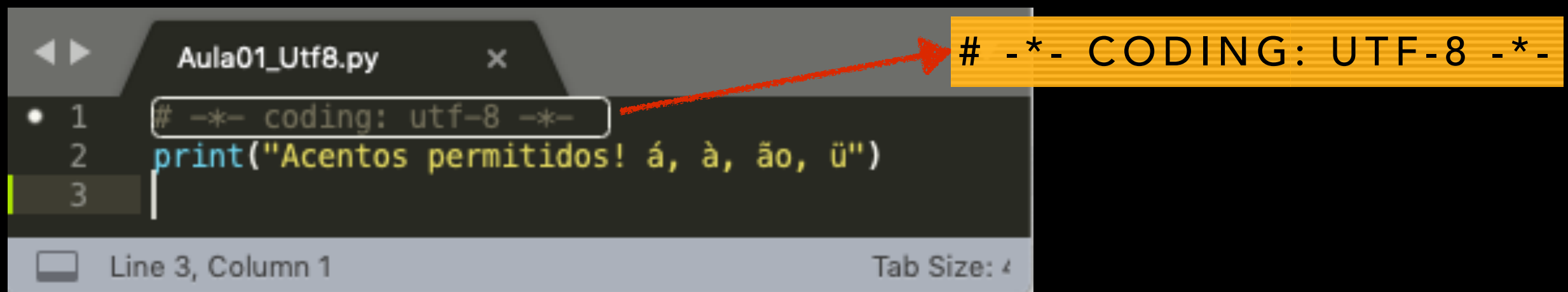
Função de saída print()

CODIFICAÇÃO

CODIFICAÇÃO

PADRÃO

- A codificação **UTF-8** é utilizada por padrão em python.
- Porém, certifique-se do padrão utilizado pelo seu sistema operacional.
- O padrão brasileiro é configurado como UTF-8 ou iso-8859-1.
- Para o modo de script, utilizaremos na primeira linha do arquivo a seguinte estrutura que garantirá a codificação utf-8 ou iso-8859-1.



The image shows a code editor window titled 'Aula01_Utf8.py'. The editor contains three lines of code: line 1 is '# -*- coding: utf-8 -*-', line 2 is 'print("Acentos permitidos! á, à, ão, ü")', and line 3 is empty. A red arrow points from the encoding declaration on line 1 to a yellow callout box on the right that contains the text '# -*- CODING: UTF-8 -*-'. The status bar at the bottom indicates 'Line 3, Column 1' and 'Tab Size: 4'.

```
# -*- coding: utf-8 -*-  
print("Acentos permitidos! á, à, ão, ü")  

```

MAIORES INFORMAÇÕES: [HTTPS://DOCS.PYTHON.ORG/3.7/REFERENCE/LEXICAL_ANALYSIS.HTML#ENCODING-DECLARATIONS](https://docs.python.org/3.7/reference/lexical_analysis.html#encoding-declarations)

COMENTÁRIOS

COMENTÁRIOS

PADRÃO

- Um **comentário** em um programa de computador é um texto destinado apenas ao leitor humano.
- É completamente ignorado pelo interpretador.
- Utilizamos o **símbolo #**, no início da linha para indicar um comentário, assim, o resto da linha é ignorado.

COMENTÁRIOS

PADRÃO

- **Comentários em diversas linhas**, poderemos utilizar 3 aspas duplas seguidas no início do comentário, e ao finalizar, inserir, mais 3 aspas duplas.
- Utilizamos o **símbolo** `"""`, no início da primeira linha para indicar um comentário e no final, `"""`.

TIPOS DE DADOS

TIPOS DE DADOS

VALOR

- Um **valor** pode ser qualquer informação, seja: uma letra, um número, um vídeo, uma música que será manipulado pelo programa seguindo as instruções passadas. Os **valores** são classificados em diferentes **tipos de dados**.
- Para identificarmos o **tipo de dado** de um determinado valor, utilizamos o seguinte comando no modo interativo.

```
>>> type(<valor a ser testado>)\n>>> |
```

TIPOS DE DADOS

TYPE()

```
>>> type(4)
<class 'int'>
>>> type("Ana")
<class 'str'>
>>> type(True)
<class 'bool'>
>>> type([123, 'bletch', 'xyzzzy', 1234])
<class 'list'>
>>> type(("x", "y", 10, 20, "c"))
<class 'tuple'>
>>> type({1: 'Mínimo', 10: 'Máximo', 5: 'Meio'})
<class 'dict'>
>>> |
```

TIPOS DE DADOS

TIPOS DE DADOS

- **int**: números inteiros
- **str**: conjunto de caracteres (strings)
- **bool**: armazenam os valores True, False
- **list**: agrupa um conjunto de elementos
- **tuple**: semelhante ao list, porém, imutável
- **dict**: agrupa elementos que somente serão recuperados por meio de uma chave.

VARIÁVEIS

VARIÁVEIS

- É um espaço de memória que reservamos para armazenar valores temporários que estão sendo processados ou manipulados.
- Seu valor poderá ser alterado a qualquer momento.
- O nome da variável, referência, é como iremos nos referir a um determinado espaço de memória.

ATRIBUIÇÃO

ATRIBUIÇÃO

- Utilizamos o **símbolo** de igual **=** para atribuírmos valores às variáveis.

x = 20

B = 10.6

nome = "Ana"

INDENTAÇÃO

INDENTAÇÃO

- Em Python, a **indentação possui uma função muito importante.**
- Os **blocos de instruções** são delimitados pela **indentação**. Sem ela, os códigos serão interpretados como se estivessem dentro de um único bloco.
- Os **blocos** são uma ou mais instruções que devem ser executadas uma após a outra, de cima para baixo da esquerda para a direita.

INDENTAÇÃO

- Primeiro nível hierárquico, código na margem esquerda.
- Segundo nível, indentado em 4 espaços ou 1 tabulação.
- Terceiro nível, indentado em 8 espaços ou 2 tabulações.

FUNÇÃO DE SAÍDA PRINT()

FUNÇÃO DE SAÍDA

PRINT()

- Existem várias maneiras de apresentar a saída de um programa.
- Os dados podem ser impressos em um formato legível por humanos ou gravados em um arquivo para uso futuro.
- Para exibirmos informações na tela, utilizamos a **função `print()`**.

FUNÇÃO DE SAÍDA

PRINT() - VERSÃO PYTHON 2.X

```
>>> nome = "Ana Paula"
>>> ano = 2008
>>> # versão Python 2.x
>>> print('A profa. %s iniciou no Instituto no ano de %s.' %(nome, ano))
A profa. Ana Paula iniciou no Instituto no ano de 2008.
>>> |
```

FUNÇÃO DE SAÍDA

PRINT() - VERSÃO PYTHON 3.X

```
>>> nome = "Ana Paula"
>>> ano = 2008
>>> # versão Python 3.x
>>> print('A profa. {0} iniciou no Instituto no ano de {1}.'.format(nome, ano))
A profa. Ana Paula iniciou no Instituto no ano de 2008.
>>> |
```

FUNÇÃO DE SAÍDA

PRINT() - VERSÃO PYTHON 3.7

```
>>> nome = "Ana Paula"
>>> ano = 2008
>>> # versão Python 3.7
>>> print(f'A profa. {nome} iniciou no Instituto no ano de {ano}.')
A profa. Ana Paula iniciou no Instituto no ano de 2008.
>>> |
```

PERGUNTAS ?

REFERÊNCIAS

<http://python.org.br>