

Ed u c a ç ã o
P r o f i s s i o n a l
P a u l i s t a

Técnico em
Ciência de
Dados

Estrutura de controle de fluxo

Compreensão de listas

Aula 2

Código da aula: [DADOS]ANO1C2B2S13A2



Objetivos da Aula

Revisar o conceito de dicionário e aprender compreensão de dicionário.



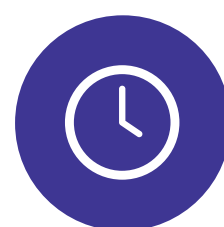
Competências da Unidade (Técnicas e Socioemocionais)

- Ser proficiente em linguagens de programação para manipular e analisar grandes conjuntos de dados.
- Usar técnicas para explorar e analisar dados, aplicar modelos estatísticos, identificar padrões, realizar inferências e tomar decisões fundamentadas em evidências.
- Colaborar efetivamente com outros profissionais, como cientistas de dados e engenheiros de dados; trabalhar em equipes multifuncionais, colaborando com colegas, gestores e clientes.



Recursos Didáticos

- Recurso audiovisual para exibição de vídeos e imagens.
- Acesso ao laboratório de informática e/ou à internet.
- Software Anaconda/Jupyter Notebook instalado ou similar.



Duração da Aula

50 minutos

Relembrando

Dicionário	
Definição	<ul style="list-style-type: none">Dicionários são estruturas de dados em Python que armazenam pares chave-valor.Cada valor é associado a uma chave única, proporcionando acesso rápido e eficiente aos dados.
Sintaxe básica	<ul style="list-style-type: none">Declarados usando chaves {}.Exemplo: meu_dicionario = {'chave1': valor1, 'chave2': valor2}
Acesso aos elementos	<ul style="list-style-type: none">Acesso usando chaves: valor = meu_dicionario['chave'].As chaves podem ser de diferentes tipos (inteiros, strings etc.).
Operações comuns	<ul style="list-style-type: none">Adição de itens: meu_dicionario[nova_chave] = novo_valor.Remoção de itens: del meu_dicionario[chave_a_ser_removida].Verificação de existência: chave_existe = chave in meu_dicionario.
Mutabilidade	<ul style="list-style-type: none">Dicionários são mutáveis, ou seja, podem ser modificados após a criação.
Uso versátil	<ul style="list-style-type: none">Dicionários são úteis para mapear informações complexas e são amplamente utilizados em Python para representar dados estruturados.

Compreensão de dicionários

A compreensão de dicionário em Python é uma construção sintática que permite criar dicionários de maneira concisa e expressiva. Assim como a compreensão de lista, ela simplifica a criação e manipulação de dicionários em uma única linha de código.

Sintaxe básica:

```
{novo_valor: expressao for elemento in lista_original if condicao}
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



Tome nota:

novo_valor: O valor associado a uma nova chave no dicionário.

expressao: A expressão que define o valor associado a cada nova chave.

elemento: A variável que representa cada elemento na lista original.

lista_original: A lista da qual os elementos são retirados para criar as chaves no dicionário.

condicao (opcional): Uma condição que determina se um par chave-valor será incluído no dicionário.

Exemplo

Neste exemplo, a compreensão de dicionário é usada para criar um dicionário em que cada chave é um número de 1 a 5, e cada valor é o quadrado desse número. A construção é eficiente e legível, tornando-se uma alternativa elegante para loops tradicionais na criação de dicionários.

```
quadrados_dict = {x: x**2 for x in range(1, 6)}  
print(quadrados_dict)
```

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
```

Outra forma de criar o mesmo dicionário:

```
quadrados_dict = {}  
for x in range(1, 6):  
    quadrados_dict[x] = x**2  
  
print(quadrados_dict)
```

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Vamos
fazer uma
atividade

Exercício

Acompanhe as orientações ao lado para realizar esta atividade.

 **20 minutos**

 **Em grupo**

- 1** Crie uma lista dos **quadrados** dos números de 1 a 10.
- 2** Gere uma lista com os **cubos** dos números **pares** de 1 a 10.
- 3** Crie uma lista contendo apenas os **números ímpares** de 1 a 20.
- 4** Utilizando compreensão de lista, obtenha os primeiros cinco **múltiplos de 3**.
- 5** Gere uma lista com os números de 1 a 50, **excluindo os divisíveis por 5**.

Vamos
fazer uma
atividade

Exercício

Acompanhe as orientações ao lado para realizar esta atividade.

 **20 minutos**

 **Em grupo**

- 6** Crie um dicionário com **pares** chave-valor, representando o **mapeamento** de números de 1 a 5 para seus **quadrados**.
- 7** Gere um dicionário em que as **chaves** são os números de 1 a 10, e os valores são seus **cubos**.
- 8** Crie um dicionário com os números de 1 a 10 como **chaves** e seus **quadrados** como valores, mas apenas para **números ímpares**.
- 9** Utilizando compreensão de dicionário, **mapeie** os números de 1 a 5 para "**par**" se forem pares, e "**ímpar**" se forem ímpares.
- 10** Crie um dicionário em que as chaves são os números de 1 a 10, e os valores são **True** se o número for par, e **False** se o número for ímpar.



Vamos
fazer um
quiz

O que é compreensão de lista em Python?

Um método para converter listas em *strings*.

Uma técnica para criar listas de maneira mais concisa e legível.

Uma função que retorna o comprimento de uma lista.

Uma maneira de ordenar listas em ordem decrescente.



Vamos
fazer um
quiz

O que é compreensão de lista em Python?



Um método para converter listas em *strings*.

Uma técnica para criar listas de maneira mais concisa e legível.



Uma função que retorna o comprimento de uma lista.

Uma maneira de ordenar listas em ordem decrescente.



FEEDBACK GERAL DA ATIVIDADE

Compreensão de lista é uma construção em Python que permite criar listas de forma mais concisa, substituindo a necessidade de usar loops tradicionais.



Vamos
fazer um
quiz

Como é definida a compreensão de lista no Python?

[elemento for lista_original]

{elemento for lista_original}

[expressao for elemento in
lista_original if condicao]

{expressao for elemento in
lista_original if condicao}



Vamos
fazer um
quiz

Como é definida a compreensão de lista no Python?



[elemento for lista_original]

{elemento for lista_original}



[expressao for elemento in
lista_original if condicao]

{expressao for elemento in
lista_original if condicao}



FEEDBACK GERAL DA ATIVIDADE

A sintaxe básica da compreensão de lista inclui uma expressão que define os elementos da nova lista, um loop que itera sobre uma lista original e uma condição opcional para filtrar os elementos.



Vamos
fazer um
quiz

Como é a sintaxe básica da compreensão de dicionário em Python?

`{chave, valor for elemento in
lista_original}`

`[chave: valor for elemento in
lista_original]`

`{chave: expressao for elemento
in lista_original if condicao}`

`[chave: valor for elemento in
lista_original if condicao]`



Vamos
fazer um
quiz

Como é a sintaxe básica da compreensão de dicionário em Python?



**{chave, valor for elemento in
lista_original}**

**[chave: valor for elemento in
lista_original]**



**{chave: expressao for elemento
in lista_original if condicao}**

**[chave: valor for elemento in
lista_original if condicao]**



FEEDBACK GERAL DA ATIVIDADE

A sintaxe básica da compreensão de dicionário inclui uma expressão para a chave e o valor, um loop que itera sobre uma lista original e uma condição opcional para filtrar os elementos.



© Getty Images

O que nós
aprendemos
hoje?

Hoje desenvolvemos:

- 1** O conhecimento sobre o conceito de compreensão de dicionário;
- 2** A revisão sobre a definição de dicionário;
- 3** A aplicação prática de compreensão de dicionário por meio de exercícios.

Saiba mais

Quer saber mais sobre dicionário e aprender sobre tupla?

ORESTES, Y. *Python: trabalhando com dicionários*. Alura, 16 out. 2018. Disponível em: <https://www.alura.com.br/artigos/trabalhando-com-o-dicionario-no-python>. Acesso em: 24 mar. 2024.

Você acha que está programando demais? Entenda neste artigo se um cientista de dados programa.

SILVA, P. H. C. M. da. *Cientista de dados programa?* Alura, 20 jun. 2023. Disponível em: <https://www.alura.com.br/artigos/cientista-de-dados-programa>. Acesso em: 24 mar. 2024.

Referências da aula

MENEZES, N. N. C. *Introdução à programação com Python: algoritmos e lógica de programação para iniciantes*. São Paulo: Novatec, 2019.

Identidade visual: imagens © Getty Images.

Ed u c a ç ã o
P r o f i s s i o n a l
P a u l i s t a

Técnico em
Ciência de
Dados