

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

# **Programação aplicada à Ciência de Dados**

## **Estrutura de controle de fluxo**

### **Aula 1**

**Código da aula: [DADOS]ANO1C2B2S10A1**



## Objetivo da aula

- Relembrar a estrutura de controle de loop.



## 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 baseadas em evidências.



## 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.

## Exposição

# Motivação: par ou ímpar?

Formem duplas e joguem par ou ímpar.

Caso alguém não saiba como jogar, como podemos ensinar as regras dessa brincadeira?

## Instruções do Jogo "Par ou Ímpar":

### 1. Participantes:

Este jogo requer **dois** jogadores: um será designado como o "Jogador 1", e o outro como o "Jogador 2".

### 2. Rodada inicial:

O Jogador 1 faz a escolha entre "**Par**" ou "**Ímpar**". Em seguida, ao Jogador 2 atribui-se automaticamente a outra opção.

### 3. Números escolhidos:

Cada jogador escolhe um número inteiro, geralmente de **1 a 10**.

### 4. Soma dos números:

Os números escolhidos pelos dois jogadores são **somados**.

### 5. Determinar Par ou Ímpar:

O Jogador 1 (aquele que escolheu "Par") ganha se a soma for par. O Jogador 2 (aquele que escolheu "Ímpar") ganha se a soma for ímpar.

### 6. Anunciar o vencedor:

O jogador, cuja escolha (Par ou Ímpar) corresponde à **paridade** da soma, é declarado o vencedor.

### 7. Jogar novamente (opcional):

Os jogadores podem optar por jogar várias rodadas, alternando as escolhas entre "Par" e "Ímpar".



# Controle de loop em Python

Os **loops** permitem a execução repetida de um bloco de código até que uma condição seja atendida. Em Python, existem diferentes estruturas de controle de loop que oferecem flexibilidade e eficiência na implementação de iterações.

Nesta aula, vamos relembrar os principais tipos de loops em Python e explorar como controlar o fluxo de execução dentro deles.

## Estruturas básicas de loop

O loop **for** é utilizado para iterar sobre uma sequência (como uma lista, tupla, *string* ou *range*). Sua sintaxe básica é:

```
for elemento in sequencia:  
    # Código a ser repetido
```

### Exemplo:

```
1 frutas = ["maçã", "banana", "laranja"]  
2 for fruta in frutas:  
3     print(fruta)
```

```
maçã  
banana  
laranja
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

## Estruturas básicas de loop

O loop **while** executa um bloco de código, enquanto uma condição específica for verdadeira. A estrutura básica é:

```
while condição:  
    # Código a ser repetido  
    # enquanto condição é verdadeira
```

### Exemplo:

```
1 contador = 0  
2 while contador < 5:  
3     print(contador)  
4     contador += 1
```

```
0  
1  
2  
3  
4
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



Vamos  
fazer uma  
**atividade**

# Programar o jogo Par ou Ímpar

Seguindo as instruções da brincadeira Par ou Ímpar, crie em grupo um programa que simule a partida entre você e o computador.

**Obs.:** para simular a jogada do computador, você pode usar o código abaixo.

```
1 import random
2 random.randint(1, 10)
3
```

3

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.



### Dica

A biblioteca **random** é útil para gerar números aleatórios. Exemplo: o comando `random.randint(1,10)` escolhe um número aleatório entre 1 e 10.

Vamos  
fazer uma  
**atividade**

# Programar o jogo Par ou Ímpar

Veja abaixo uma sugestão para a saída do programa:

```
Bem-vindo ao Jogo Par ou Ímpar!

Escolha [P]ar ou [Í]mpar: P
Escolha um número entre 1 e 10: 3

Você escolheu 3 e o computador escolheu 3.
Você ganhou! A soma é 6

Deseja jogar novamente? (s/n): S

Escolha [P]ar ou [Í]mpar: Í
Escolha um número entre 1 e 10: 10

Você escolheu 10 e o computador escolheu 10.
Você perdeu! A soma é 20

Deseja jogar novamente? (s/n): n
Obrigado por jogar! Até a próxima.
```



© Getty Images

O que nós  
**aprendemos  
hoje?**

## Hoje desenvolvemos:

- 1** Conhecimento em estrutura de loop em Python.
- 2** Aplicação do controle de fluxo.
- 3** Aplicação e revisão das estruturas de loop (for e while).



# Saiba mais

**Veja outras formas de programar o jogo Par ou Ímpar em Python:**

CÓDIGO PYTHON. *Simples jogo do par ou ímpar*, 1 out. 2022. Disponível em: <https://codigopython.blogs.sapo.pt/simples-jogo-do-par-ou-impair-210770>. Acesso em: 28 fev. 2024.

**Aprenda a criar um jogo de adivinhação em Python:**

ALURA. *Python: começando com a linguagem*. Disponível em: <https://cursos.alura.com.br/course/python-introducao-a-linguagem/task/24543>. Acesso em: 28 fev. 2024.

**Aprenda a fazer o Jogo da Forca em Python:**

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



# Referências da aula

Identidade visual: Imagens © Getty Images.

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

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