

# Aula 5: Estruturas de Repetição

## Necessidade da Repetição

- **Pense na sua rotina:** Escovar os dentes, tomar café, ir para a aula... são ações que você repete diariamente.
- **Pense na programação:** E se você precisasse ler a nota de 40 alunos? Você escreveria 40 linhas de código pedindo `input()` para cada aluno? Muito trabalhoso, né? Você criaria 40 variáveis (`nota1`, `nota2`, ..., `nota40`)? Impraticável em casos assim!

**Solução?** Utilizar **estruturas de repetição** (ou **Laços**), permitindo que você escreva o código UMA VEZ e deixe o computador executando quantas vezes forem necessárias.

Comando	Tradução	Uso Principal
<code>for</code>	Para	Usado quando você sabe o <b>número exato</b> de repetições (ou quer percorrer uma lista/sequência).
<code>while</code>	Enquanto	Usado quando você quer repetir a ação <b>enquanto uma condição for verdadeira</b> (número de repetições é incerto).
<code>do-while</code>	Faça-Enquanto	Como Python <b>não tem</b> o comando <code>do-while</code> nativo, sua funcionalidade é simulada com o <code>while</code> e uma lógica de inicialização/controle.

## Atividade Assistida

Vamos desenvolver um programa que leia 5 números e calcule o dobro, o triplo e o quádruplo de cada um. Faremos isso de três formas, focando nas estruturas:

### Usando `for` (Quando sabemos a contagem: 5 vezes nesse caso)

O `for` em Python usa a função `range()` para determinar a contagem.

```

print("--- Usando FOR (Repetição Contada) ---")

# O range(5) gera os números 0, 1, 2, 3, 4 (5 repetições)
for i in range(5):
    try:
        # i representa o número atual da repetição (0, 1, 2...)
        print(f"Número {i + 1} de 5:")
        num = float(input("Digite um número: "))

        dobro = num * 2
        triplo = num * 3
        quádruplo = num * 4

        print(f" Resultado: Dobro={dobro}, Triplo={triplo}, Quádruplo={quádruplo}\n")

    except ValueError:
        print("Entrada inválida. Tente novamente.")

```

### Usando while (Repetição baseada em Condição)

O `while` repete o bloco de código enquanto a condição inicial for `True`. Precisamos de um contador manual para controlar o número de repetições:

```

print("--- Usando WHILE (Repetição Condicional) ---")

contador = 0 # Inicializamos o contador
limite = 5 # Definimos o limite

while contador < limite: # A condição de parada: Enquanto o contador for menor que 5
    try:
        print(f"Número {contador + 1} de {limite}:")
        num = float(input("Digite um número: "))

        dobro = num * 2
        triplo = num * 3
        quádruplo = num * 4

        print(f" Resultado: Dobro={dobro}, Triplo={triplo}, Quádruplo={quádruplo}\n")

        contador = contador + 1 # IMPORTANTÍSSIMO! Incrementa o contador para evitar loop
                                infinito

    except ValueError:
        print("Entrada inválida. Tente novamente.")
        # Não incrementamos o contador para dar nova chance ao usuário

```

## Usando do-while (Executar primeiro, checar depois)

Como o Python não tem o do-while nativo, nós o simulamos garantindo que o bloco interno execute pelo menos uma vez, ou usando um while True com um if para quebrar (comando break):

```
print("--- Simulação DO-WHILE (Executa 1ª vez, depois checa) ---")

contador = 0
limite = 5

while True: # Loop infinito garantido para executar pelo menos uma vez
    if contador >= limite:
        break # Ponto de DECISÃO: Se o limite for atingido, usamos 'break' para sair

    try:
        print(f"Número {contador + 1} de {limite}:")
        num = float(input("Digite um número: "))

        dobro = num * 2
        triplo = num * 3
        quádruplo = num * 4

        print(f"Resultado: Dobro={dobro}, Triplo={triplo}, Quádruplo={quádruplo}\n")

        contador = contador + 1 # Incremento

    except ValueError:
        print("Entrada inválida. Tente novamente.")
```

## Desafios:

### 1. Cálculo de Média Escolar para Vários Alunos

Use o laço for para repetir a lógica de cálculo de média e status (Aprovado/Reprovado/Recuperação) que você fez na Aula 4, agora para 10 estudantes.

### 2. Cadastro de Candidatos

Desenvolva um programa que colete dados de 12 pessoas, usando a decisão para filtrar candidatos menores de 18 anos.

- O programa deve pedir o **Ano de Nascimento** do candidato.
- **Se for menor de 18**, o programa deve informar que ele não pode participar e **pular** a coleta dos demais dados (telefone, email etc) para esse candidato.
- **Se for maior de 18**, o programa prossegue com o input() para os demais dados.

### 3. Tentativa de Login e Senha

Simule um sistema de login simples onde o usuário tem um número limitado de tentativas para digitar a senha correta.

- Defina um nome de **usuário** e uma **senha** corretos (ex: `admin` e `123456`).
- Dê ao usuário **3 tentativas** para acertar a combinação.
- **Se a senha estiver correta**, imprima uma mensagem de sucesso e use o comando `break` para sair do loop.
- **Se a senha estiver errada**, informe o erro e diminua o número de tentativas restantes.
- **Se as tentativas acabarem**, imprima uma mensagem de bloqueio.