# PROGRAMAÇÃO EM PYTHON

SENAI

# PROGRAMAÇÃO EM PYTHON

## Plano de Aula

**Conteúdo:**

- Desvio Condicional;
- Laço de Repetição;
- Atividades;

## Inicio:

As informações deste conteúdo visam compreender o conteúdo do curso.

SENAI

# PROGRAMAÇÃO EM PYTHON

## Objetivo

Capacitar profissionais para desenvolver aplicações em linguagem Python, por meio de técnicas de programação, seguindo boas práticas, procedimentos e normas.

SENAI

PROGRAMAÇÃO EM PYTHON

# Desvio Condicional


Um desvio condicional é uma estrutura que permite alterar o fluxo de execução de um programa com base na avaliação de uma condição.

Em vez de seguir uma sequência linear fixa, o programa decide qual conjunto de instruções executar conforme o resultado dessa verificação, possibilitando a implementação de lógicas de tomada de decisão essenciais para a flexibilidade e adaptabilidade do código.

03-Habilitacao.py

```
1   nome = input("Qual é o seu nome? ")
2   idade = int(input("Qual sua idade? "))
3
4   if idade >= 18:
5       print("Maior de idade")
6
```

O aluno pode testar a variável idade sem utilizar a função int(), o que causará um erro de tipo.

```
Qual é o seu nome? Maria
Qual sua idade? 28
Maior de idade
```

SENAI

PROGRAMAÇÃO EM PYTHON

# Desvio Condicional


**Uma observação no nosso código:**

Em Python, o caractere dois-pontos (`:`) indica o início de um novo bloco de código. Ele informa ao interpretador que as instruções que seguem e que estiverem **indentadas**, fazem parte daquele bloco.

A **indentação** (normalmente de 4 espaços) define o escopo desse bloco, ou seja, quais comandos serão executados como parte da estrutura condicional. Sem a indentação correta, o Python não consegue determinar onde o bloco termina, o que pode levar a erros de execução.

**Condicional Simples.**

**Indentação**
Utilizamos a tecla TAB para facilitar a organização e legibilidade do código.

```
if idade >= 18:
    print("Maior de idade")
```

**:**
Início de um bloco de código.

SENAI

PROGRAMAÇÃO EM PYTHON

# Desvio Condicional


## Condicional Composta (if-else)

03-Habilitacao.py

```
1   nome = input("Qual é o seu nome? ")
2   idade = int(input("Qual sua idade? "))
3
4   if idade >= 18:
5       print("Maior de idade")
6   else:
7       print("Menor de idade")
8
```

PROBLEMAS SAÍDA CONSOLE DE DEPURAÇÃO TERMINAL PORTAS COM

```
PS D:\Fabrica de Programadores> python.exe .\03-Habilitacao.py
Qual é o seu nome? Maria
Qual sua idade? 17
Menor de idade
PS D:\Fabrica de Programadores>
```

Permite definir um caminho alternativo com o **else** para quando a condição do if não for satisfeita.

SENAI

PROGRAMAÇÃO EM PYTHON

# Desvio Condicional


## Condicional Aninhada

03.2-Habilitacao.py

```python
nome = input("Qual é o seu nome? ")
idade = int(input("Qual sua idade? "))
possui_carteira = input("Possui carteira de motorista? \n (1-Sim / 2-Não) ")

if idade >= 18:
    if possui_carteira == "1":
        print("Pode dirigir")
    else:
        print("Não pode dirigir")
else:
    print("Menor de idade")

```

PROBLEMAS SAÍDA CONSOLE DE DEPURAÇÃO TERMINAL PORTAS COMENTÁRIOS

```
PS D:\Fabrica de Programadores> python.exe .\03.2-Habilitacao.py
Qual é o seu nome? Maria
Qual sua idade? 28
Possui carteira de motorista?
 (1-Sim / 2-Não) 2
Não pode dirigir
PS D:\Fabrica de Programadores>
```

Consiste em colocar um bloco condicional dentro de outro. Isso é útil quando há necessidade de verificar múltiplas condições sequencialmente.

SENAI

PROGRAMAÇÃO EM PYTHON

# Desvio Condicional


## Condicional Composta com Múltiplos (if-elif-else)

04-Temperatura.py

```python
temperatura = float(input("Digite a temperatura em Celsius: "))

if temperatura >= 30:
    print("Está quente!")
elif temperatura >= 20:
    print("Está agradável.")
elif temperatura >= 10:
    print("Está frio!")
else:
    print("Está muito frio!")

```

PROBLEMAS SAÍDA CONSOLE DE DEPURAÇÃO TERMINAL PORTAS COMENTÁRIOS

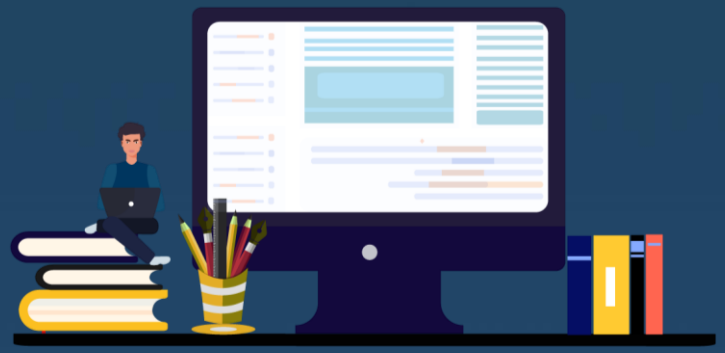
```
PS D:\Fabrica de Programadores> python.exe .\04-Temperatura.py
Digite a temperatura em Celsius: 22
Está agradável.
PS D:\Fabrica de Programadores>
```

Permite testar várias condições sequencialmente usando elif para condições intermediárias.

SENAI

PROGRAMAÇÃO EM PYTHON

# Apresentação da situação de aprendizagem *1*



## Sistema de Avaliação de Desempenho Escolar

Imagine que você foi contratado pelo **ESCOLA-FABPRO** para desenvolver uma ferramenta simples que auxilie os professores na avaliação dos alunos em cálculos nas médias escolares.

O objetivo é criar um programa que receba nome do aluno, três notas, calcule a média aritmética e com base nessa média, informe se o aluno está "Aprovado", "Em Recuperação" ou "Reprovado".

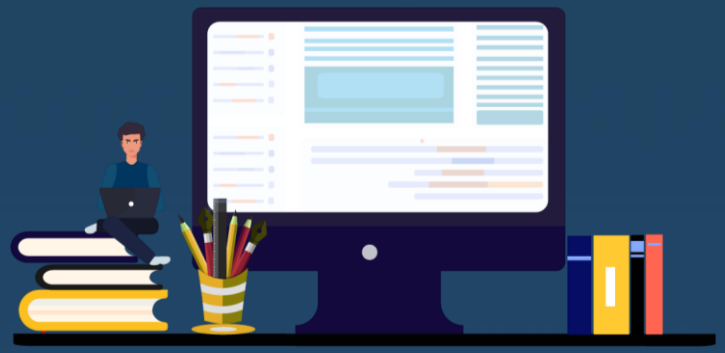
**O programa deve seguir as seguintes regras:**

- Se a média for maior ou igual a 7, o aluno é considerado Aprovado.
- Se a média for maior que 4, o aluno está Em Recuperação.
- Caso contrário, o aluno será Reprovado.

SENAI

PROGRAMAÇÃO EM PYTHON

# Apresentação da situação de aprendizagem *2*



## Monitoramento de Saúde com Cálculo de IMC

Imagine que você foi convidado para desenvolver uma aplicação simples que auxilie as pessoas a monitorarem seu estado de saúde. Neste cenário, o objetivo é criar um programa que receba o peso e a altura do usuário, calcule o Índice de Massa Corporal (IMC) e exiba o valor calculado. Com base no resultado, o programa deve indicar uma mensagem que oriente o usuário quanto à sua condição de saúde:

**O programa deve seguir as seguintes regras:**

- Se o IMC for maior ou igual a 30.0, a mensagem exibida será "Cuidado com a Saúde".
- Caso contrário, a mensagem será "Tudo ok".

---

**Observação:** O aluno pode incrementar o código seguindo a tabela.

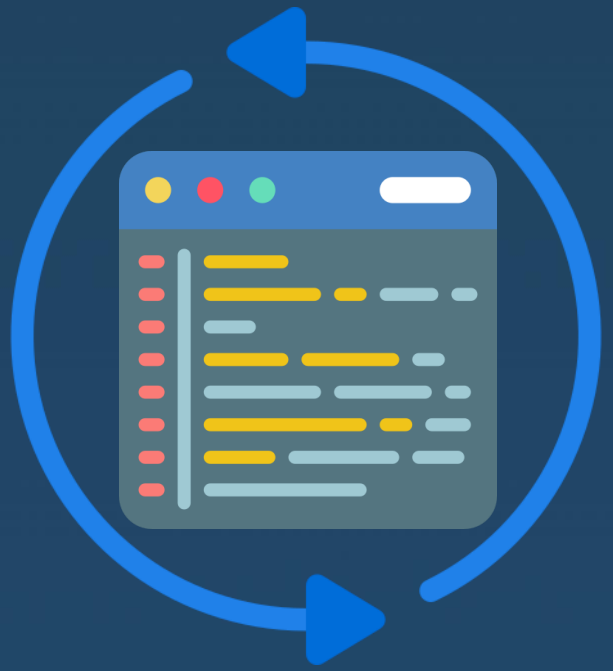
- **Abaixo de 18.5:** Abaixo do peso
- **Abaixo de 24.9:** Peso normal
- **Abaixo de 29.9:** Sobrepeso
- **Abaixo de 34.9:** Obesidade Grau I
- **Abaixo de 39.9:** Obesidade Grau II
- **40.0 ou mais:** Obesidade Grau III (mórbida)

Condicional Composta com Múltiplos (if-elif-else)

SENAI

PROGRAMAÇÃO EM PYTHON

# Laço de Repetição



Laços de repetição permitem executar um bloco de código diversas vezes, facilitando a automatização de tarefas e o processamento de coleções de dados.

**Laço for:**

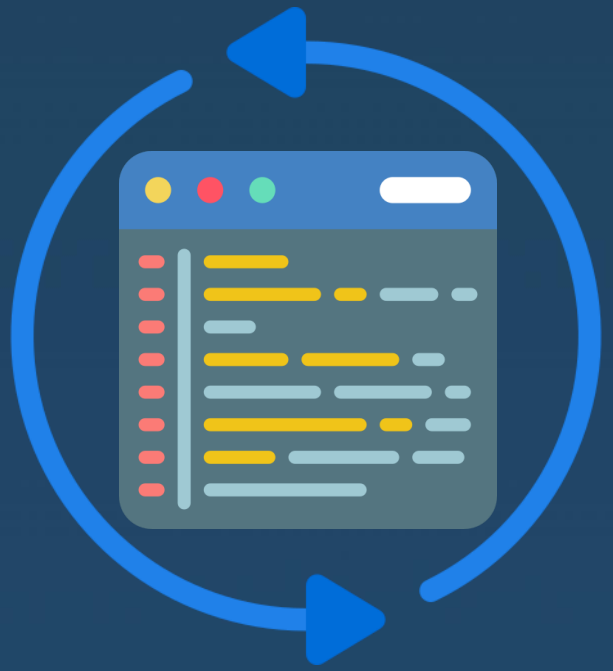
Utilizado para iterar sobre elementos de uma sequência. Em cada **iteração**, um elemento da coleção é atribuído a uma variável temporária e o bloco de código é executado para esse elemento. Essa estrutura é ideal quando se sabe a quantidade de elementos a serem percorridos ou quando se deseja processar todos os itens de uma coleção.

**OBS:** Iteração é o processo de repetir um conjunto de instruções. Permitindo percorrer elementos do seu código enquanto uma condição for verdadeira, facilitando tarefas repetitivas.

SENAI

PROGRAMAÇÃO EM PYTHON

# Laço de Repetição



**Laço while:**

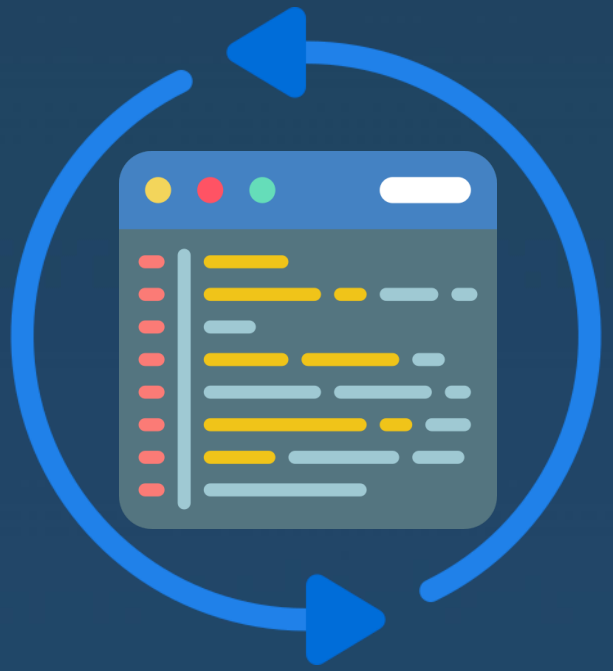
Executa o bloco de código enquanto uma condição lógica permanecer verdadeira. Esse laço é indicado quando o número de repetições não é conhecido de antemão, e a continuidade da execução depende de uma condição que pode mudar durante o processo.

É importante garantir que a condição seja eventualmente alterada para evitar loops infinitos.

SENAI

PROGRAMAÇÃO EM PYTHON

# Laço de Repetição



Um exemplo prático de desenvolver um sistema de tabuada sem laço de repetição.

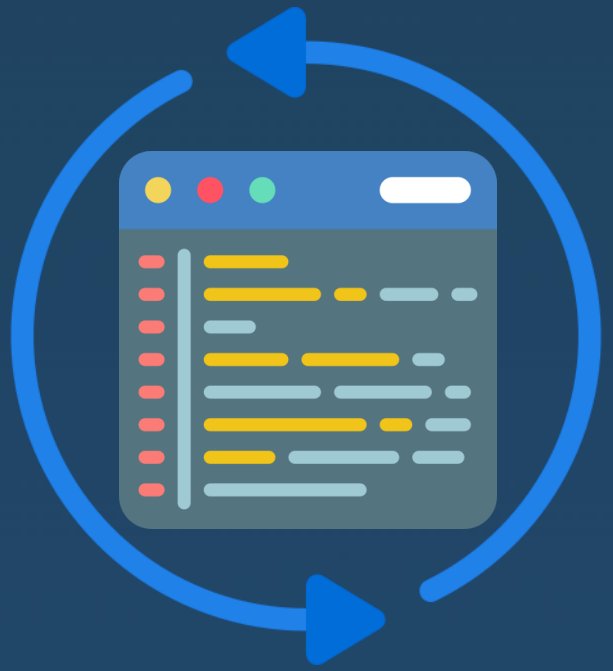
05-Tabuada.py

```
1   numero = 5
2
3   print(f" 1 x {numero} = {1 * numero}")
4   print(f" 2 x {numero} = {2 * numero}")
5   print(f" 3 x {numero} = {3 * numero}")
6   print(f" 4 x {numero} = {4 * numero}")
7   print(f" 5 x {numero} = {5 * numero}")
8   print(f" 6 x {numero} = {6 * numero}")
9   print(f" 7 x {numero} = {7 * numero}")
10  print(f" 8 x {numero} = {8 * numero}")
11  print(f" 9 x {numero} = {9 * numero}")
12  print(f"10 x {numero} = {10 * numero}")
```

SENAI

PROGRAMAÇÃO EM PYTHON

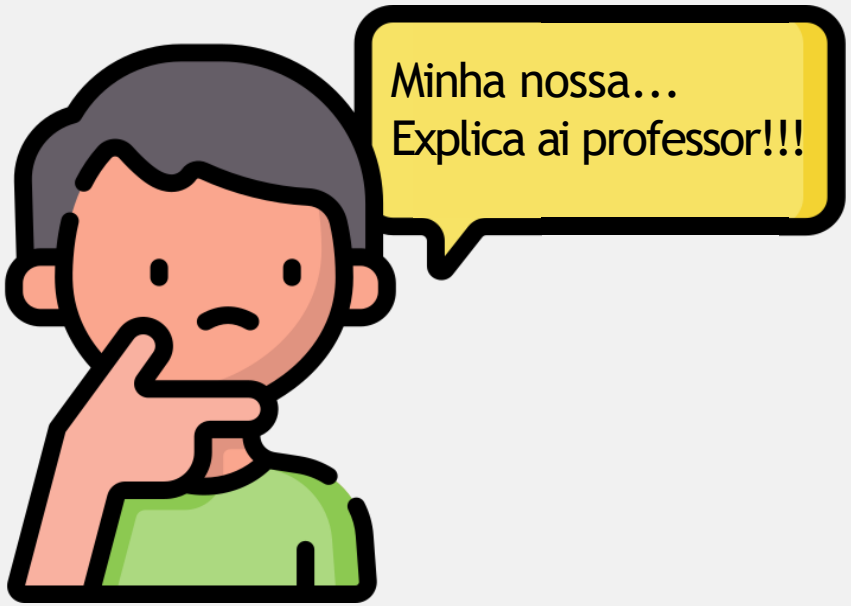
# Laço de Repetição



Um exemplo prático de como desenvolverum sistema de tabuada utilizando o laço de repetição com a função - **for**.

05.1-Tabuada.py

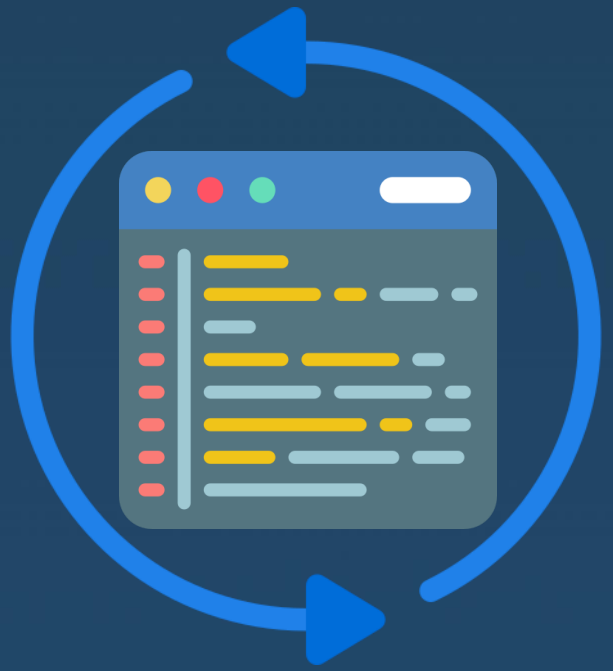
```
1   numero = 5
2
3   for i in range(1, 11):
4       print(f" {i} x {numero} = {i * numero}")
5
```




SENAI

PROGRAMAÇÃO EM PYTHON

# Laço de Repetição



05.1-Tabuada.py

```python
numero = 5

for i in range(1, 11):
    print(f" {i} x {numero} = {i * numero}")

```

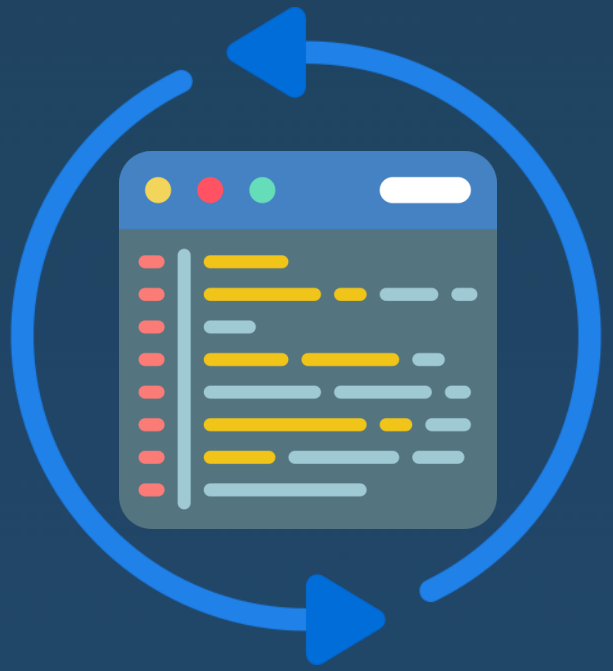
O in é um operador usado para verificar se um valor está presente em uma sequência. Quando usado em um laço de repetição (como no for), ele permite percorrer cada item dentro de uma sequência, neste caso de 1 até 10. Pois estamos usando uma função chamada range(), comumente utilizada para percorrer uma sequencia de números.

i é nossa variável que vai receber a cada iteração do laço valores de 1 até 10.

SENAI

PROGRAMAÇÃO EM PYTHON

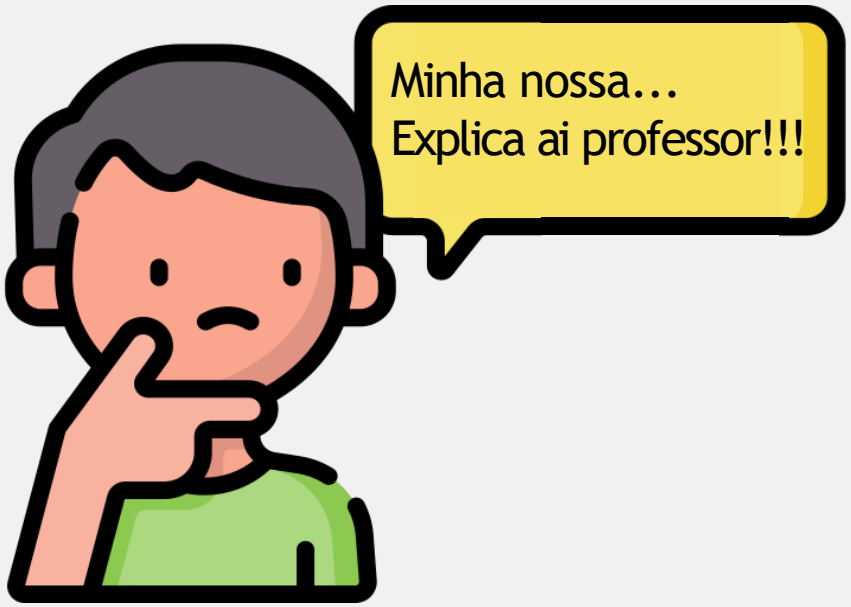
# Laço de Repetição



Um exemplo prático de como desenvolverum sistema de tabuada utilizando o laço de repetição com a função - **while**.

05.2-Tabuada.py

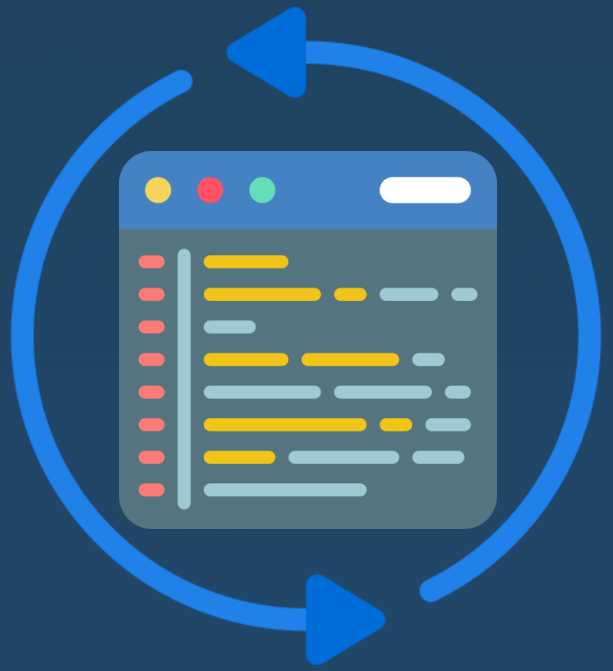
```
1   numero = 5
2   i = 1
3
4   while i <= 10:
5       print(f" {i} x {numero} = {i * numero}")
6       i += 1
```




SENAI

PROGRAMAÇÃO EM PYTHON

# Laço de Repetição



05.2-Tabuada.py

```python
numero = 5
i = 1

while i <= 10:
    print(f" {i} x {numero} = {i * numero}")
    i += 1
```

O laço while vai continuar executando enquanto a condição **i <= 10** for verdadeira.

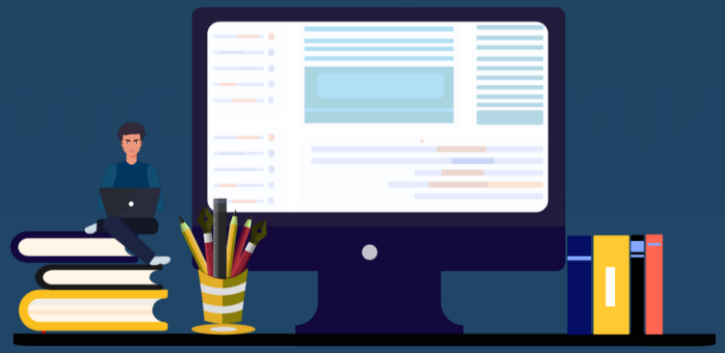
A cada iteração do laço, o valor de i aumenta para +1 (por causa da linha i += 1), isso é chamado de incremento de variável.

**Nota:** Um laço de repetição não é executado ou interrompe sua execução assim que a condição lógica se torna falsa.

SENAI

PROGRAMAÇÃO EM PYTHON

# Apresentação da situação de aprendizagem *3*



## Tabuada Personalizada na Escola FABPROG

Nesta atividade, os alunos desenvolverão um programa que solicita a entrada de um número inteiro e a quantidade de vezes que a tabuada desse número deverá ser exibida. O programa calculará e apresentará a tabuada de 1 até 10 para o número informado, repetindo a exibição conforme a quantidade especificada.

**O programa deve seguir as seguintes regras:**

- Usuário deve digitar o número para a tabuada.
- Deve digitar da onde a tabuada deverá começar.
- Digite até qual número o multiplicador deva ir.

Esta atividade pode ser realizada usando laço de repetição for ou while.

SENAI

# REFERÊNCIAS


ALVES, William Pereira. Lógica de programação de computadores. São Paulo: Érica, 2012.

ASCENCIO, Ana F. G.; CAMPOS, Edilene A. V. Fundamentos da programação de computadores. 3. ed. São Paulo: Pearson, 2012.

FARRELL, Joyce. Lógica e design de programação. São Paulo: Cengage Learning, 2010.

MANZANO, J. A. N. G.; OLIVEIRA, J. F. Algoritmos: Lógica para desenvolvimento de programação de computadores. 27. ed. rev. São Paulo: Érica, 2014.

MEDINA, Marco; FERTIG, Cristina. Algoritmos e programação. Teoria e prática. São Paulo: Novatec, 2005.

PEREIRA, Silvio do Lago. Algoritmos e lógica de programação em C: Uma abordagem didática. São Paulo: Érica, 2010.


## Importante:

Os conteúdos disponibilizados são específicos para este curso/turma, a divulgação ou reprodução do material para outras pessoas/organização não é autorizada.



ffigaro@sp.senai.br


Fernando Oliveira Figaro


Instrutor de Formação Profissional III


SENAI