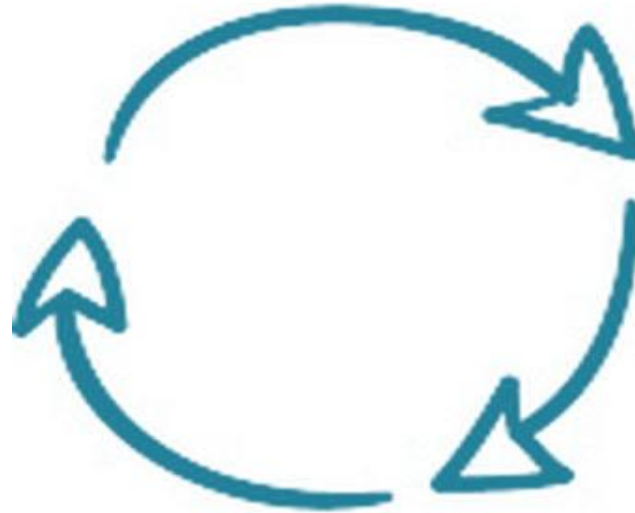




Escola de Engenharia FCI



Estrutura de Repetição (Loops)

Loop (Looping, Laço, Repetição)



Fonte: https://upload.wikimedia.org/wikipedia/commons/thumb/8/8f/Repeat_font_awesome.svg/512px-Repeat_font_awesome.svg.png

Comportamento

Loop



- Comando 1

- Comando 2

- Comando 3

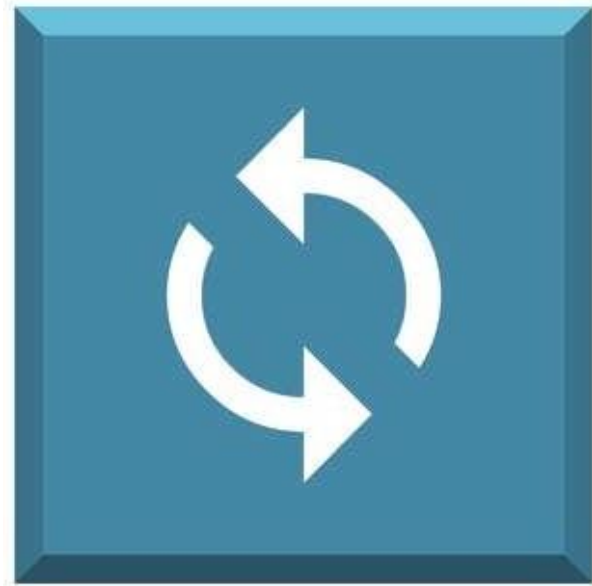
- Comando 4

- Comando 5



Comandos de Looping

- **while ()**
- for ()



Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

n=1 # inicia com o 1º valor

...

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
➤
```

Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1  
print(n) # exibe o valor na tela
```

...

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
➤
```

Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1  
print(n)  
n=n+1 # incrementar a variável
```

...

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
➤
```

Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1  
print(n)  
n=n+1  
print(n) # exibir o novo valor
```

...

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
➤
```


Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1  
print(n)  
n=n+1  
print(n)  
n=n+1 #incrementar novamente  
  
...
```

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
➤
```

Analizando sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1
print(n)
n=n+1
print(n)
n=n+1
print(n)    # exibir o valor
```

...

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
>
```

Analizando sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1      # iniciar a variável
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
...
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
> |
```

Analizando sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1          # iniciar a variável
print(n)     # exibir a variável
n=n+1        # incrementar
print(n)     # exibir a variável
n=n+1        # incrementar
print(n)     # exibir a variável
n=n+1        # incrementar
...
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
> 
```

Analizando sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1      # iniciar a variável
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
...
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
> 
```

Analizando sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

```
n=1      # iniciar a variável
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
print(n) # exibir a variável
n=n+1    # incrementar
...
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
> |
```

Sem Repetição...

Elaborar um programa para exibir os números inteiros entre 1 e 15:

n=1

print(n)

n=n+1

print(n)

n=n+1

print(n)

n=n+1

...

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
>
```

Sintaxe do comando WHILE

Comandos Anteriores

while condição:

comando 1

comando 2

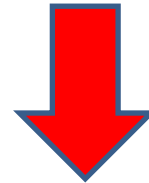
...

comando n

true



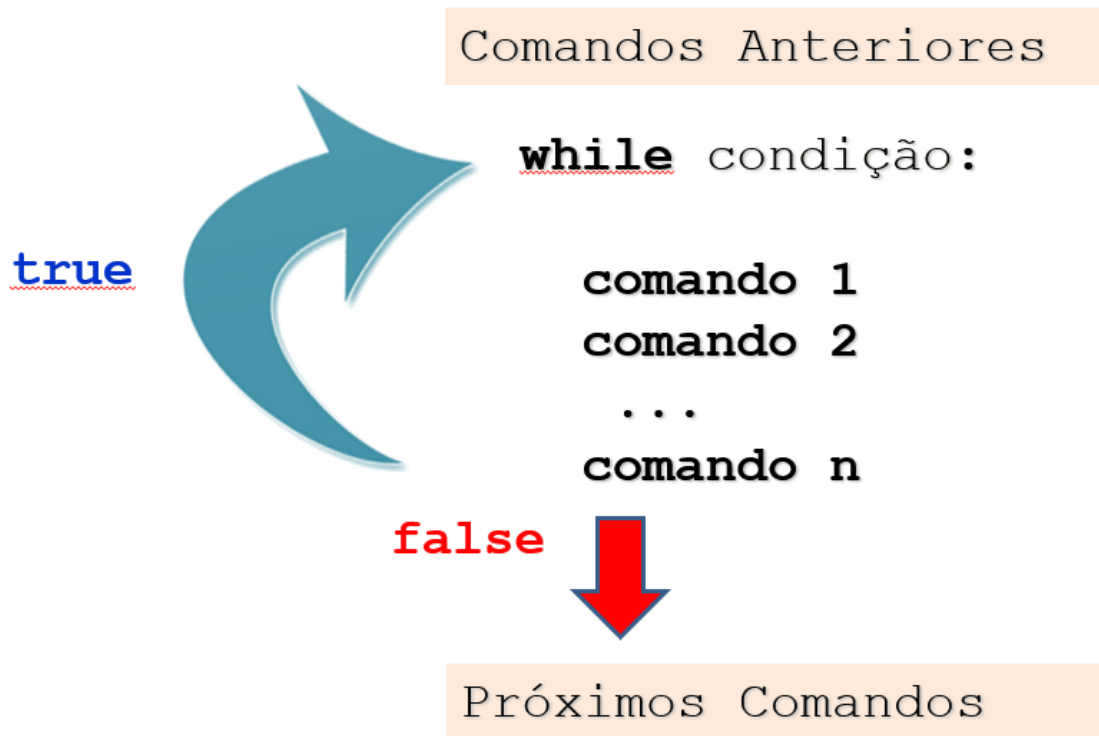
false



Próximos Comandos

Aplicando o loop while...

Sintaxe do comando WHILE



```
n=1
print(n)
n=n+1
print(n)
n=n+1
print(n)
n=n+1
...
```



Resolução



```
n=1
```

```
while n<=15:
```

```
    print(n)
```

```
    n=n+1
```

Com Repetição (Loop)

Elaborar um programa para exibir os **números inteiros entre 1 e 15**. Depois, exibir a palavra “FIM”

Sintaxe do comando WHILE

Comandos Anteriores

while condição:

comando 1

comando 2

...

comando n

true



false



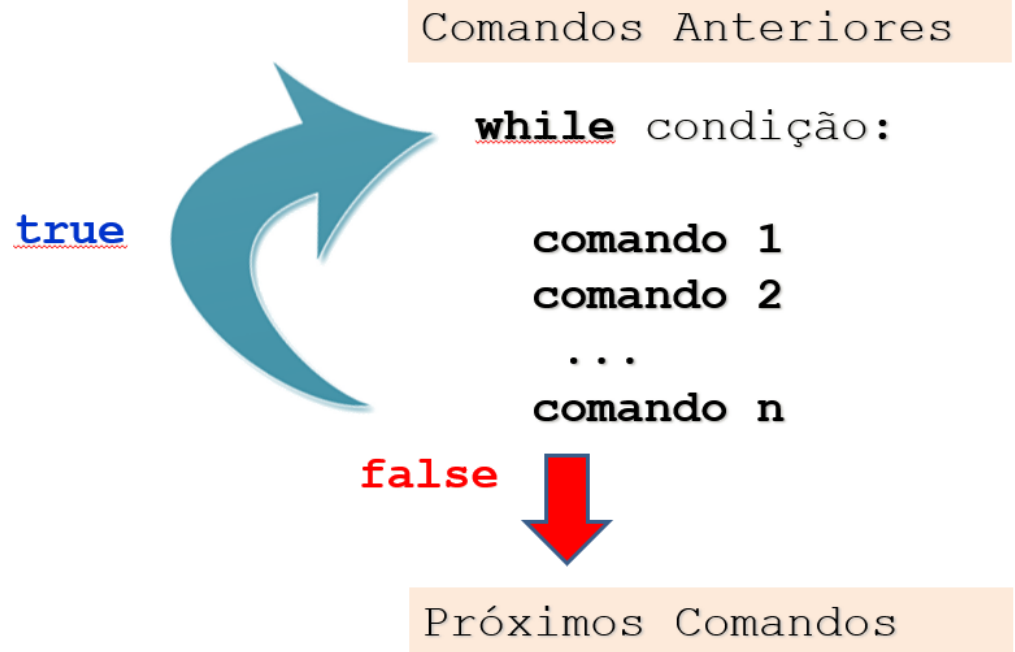
Próximos Comandos

Com Repetição (Loop)

Elaborar um programa para exibir os **números inteiros entre 1 e 15**. Depois, exibir a palavra “FIM”

```
n=1
while n<=15:
    print(n)
    n=n+1
print("FIM")
```

Sintaxe do comando WHILE



Com Repetição (Loop)

Elaborar um programa para exibir os **números inteiros entre 1 e 15**. Depois, exibir a palavra “FIM”

```
n=1
```

```
while n<=15:
```

```
    print(n)
```

```
    n=n+1
```

```
print("FIM")
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
FIM
> 
```

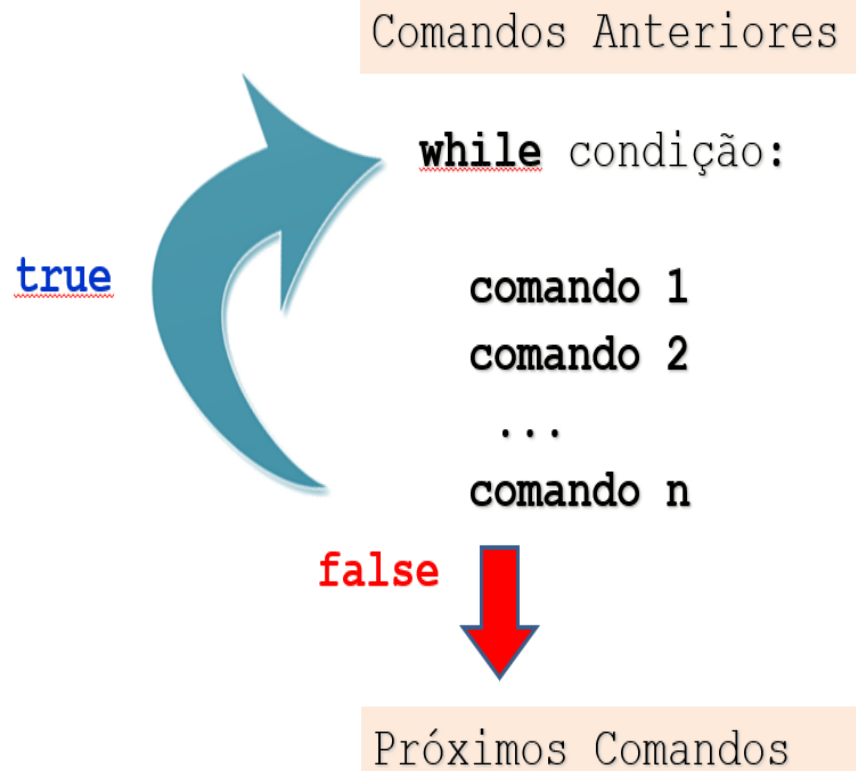



Sua vez...

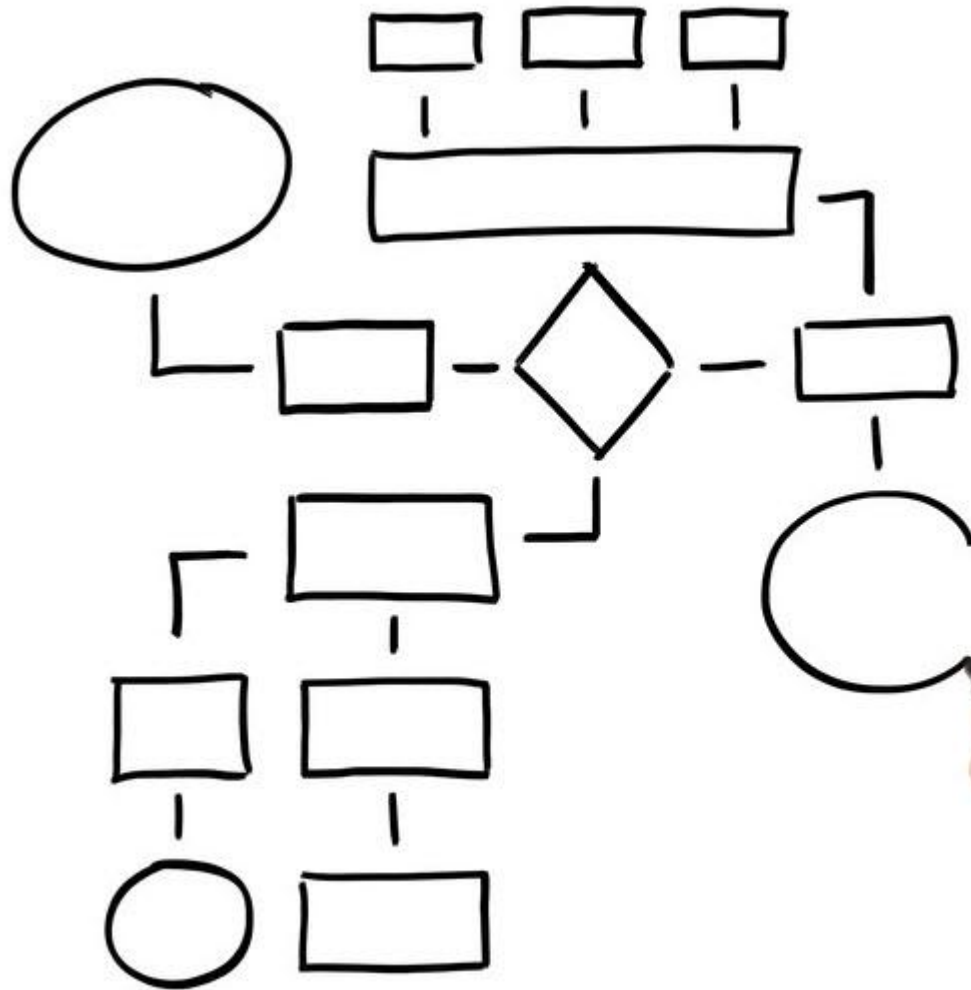
Utilizando o **looping while**, elaborar programas para exibir na tela:

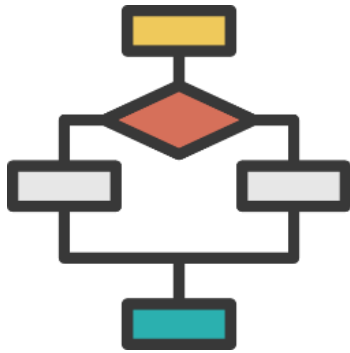
- a) Números ímpares entre 150 e 200
- b) Números em contagem regressiva entre 20 e 0

Sintaxe do comando **WHILE**

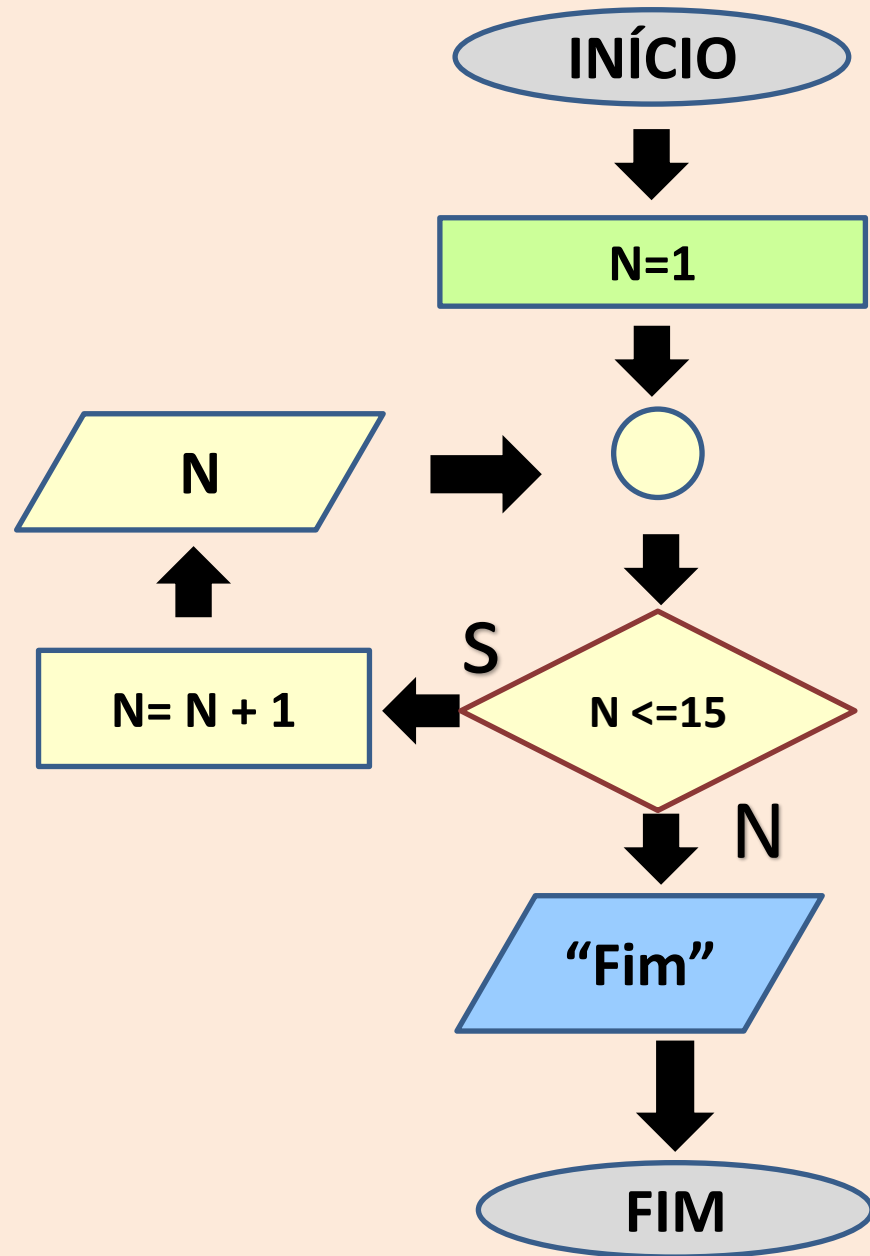


E como ficaria o FLUXOGRAMA com o Looping while?





**Fluxograma
com Looping:**



Aplicação do Looping while

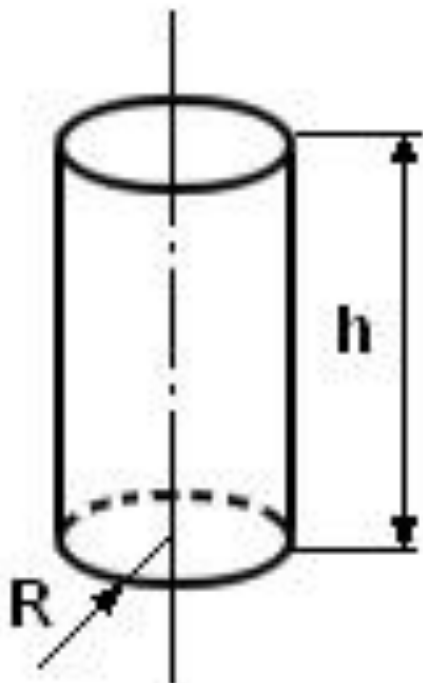
Consistência de Dados (Validação)



<https://img.freepik.com/>

Os dados digitados precisam ser verificados...

Elaborar um programa para calcular a área total (S) e o volume (V) de um cilindro reto.



$$S = 2\pi R(R + h)$$

$$V = \pi R^2 h$$



Codificação

$$S = 2\pi R(R + h)$$

$$V = \pi R^2 h$$

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

```
h= float (input ("Altura: "))
```

```
S= 2*m.pi*R*(R+H)
```

```
V= m.pi*m.pow(R,2)*h
```

```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```



Codificação

$$S = 2\pi R(R + h)$$

$$V = \pi R^2 h$$

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

```
h= float (input ("Altura: "))
```

```
S= 2*m.pi*R*(R+H)
```

```
V= m.pi*m.pow(R,2)*h
```

```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```

Na Entrada de Dados,
qualquer valor poderá
ser digitado!!
Vamos verificar...



Codificação

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

```
h= float (input ("Altura: "))
```



Codificação

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

```
h= float (input ("Altura: "))
```



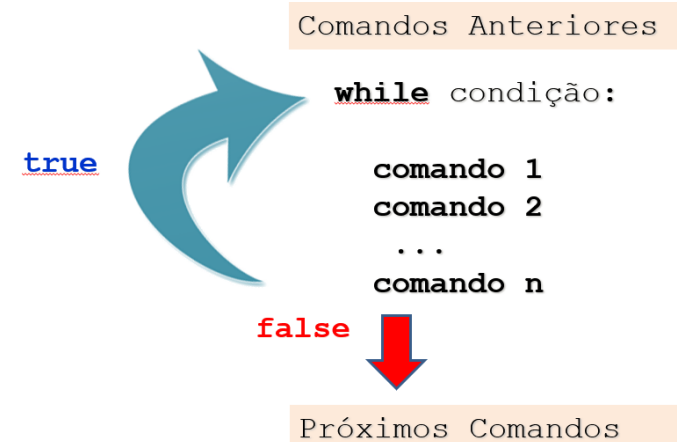
Codificação

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

Sintaxe do comando WHILE





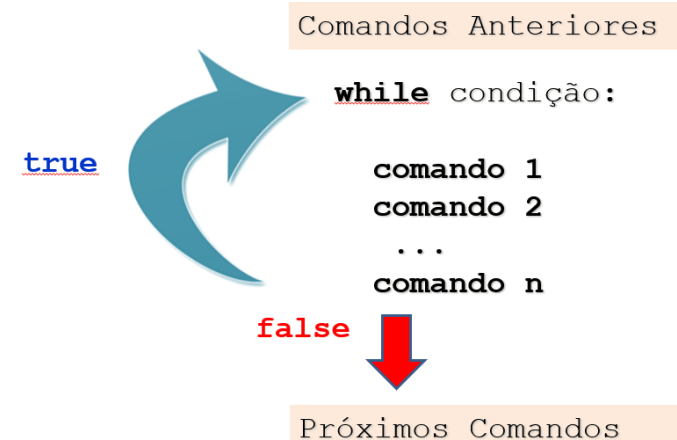
Codificação

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor errado!

R= float (input ("Raio: "))

Sintaxe do comando WHILE





Codificação

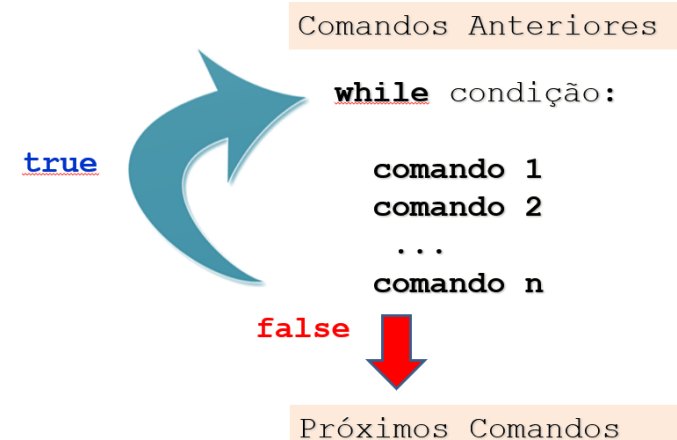
Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while ????

R= float (input ("Raio: "))

Sintaxe do comando WHILE





Codificação

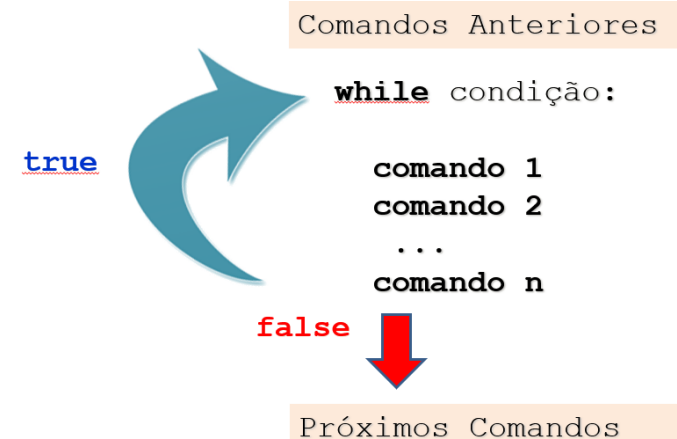
Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while **R<=0:**

R= float (input ("Raio: "))

Sintaxe do comando WHILE





Codificação

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while **R<=0:**

R= float (input ("Raio: "))





Codificação

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while R<=0:

R= float (input ("Raio: "))





Codificação

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while R<=0:

R= float (input ("Raio: "))

h= float (input ("Altura: "))





Codificação

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while R<=0:

R= float (input ("Raio: "))

h= float (input ("Altura: "))





Codificação do looping

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while R<=0:

R= float (input ("Raio: "))



h=0

while h<=0:

h = float (input ("Altura: "))





Codificação quase completa....

Área e Volume do Cilindro Reto

```
import math as m
```

```
R= float (input ("Raio: "))
```

```
h= float (input ("Altura: "))
```

```
S= 2*m.pi*R*(R+h)
```

```
V= m.pi*m.pow(R,2)*h
```

```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```

$$S = 2\pi R(R + h)$$

$$V = \pi R^2 h$$



Codificação quase completa....

```
import math as m
```

```
R=0
```

```
while R<=0:
```

```
    R= float (input ("Raio: "))
```



```
h=0
```

```
while h<=0:
```

```
    h= float (input ("Altura: "))
```



```
S= 2*m.pi*R*(R+h)
```

```
V= m.pi*m.pow(R,2)*h
```

```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```

$$S = 2\pi R(R + h)$$

$$V = \pi R^2 h$$

▶ Run

main.py × +

```
1 # Área e Volume do Cilindro Reto
2 import math as m
3 R=0
4 ▼ while R<=0:
5     R= float (input ("Raio: "))
6     h=0
7 ▼ while h<=0:
8     h= float (input ("Altura: "))
9     S= 2*m.pi*R*(R+h)
10    V= m.pi*m.pow(R,2)*h
11    print(" Área Total= ", S)
12    print(" Volume= ", V)
13
```

>_ Console ×



Shell × +

```
Raio: 0
Raio: -5
Raio: 10
Altura: 0
Altura: -20
Altura: 15
Área Total= 1570.7963267948965
Volume= 4712.38898038469
> █
```



Codificação quase completa....

```
import math as m
```

```
R=0
```

```
while R<=0:
```

```
    R= float (input ("Raio: "))
```



```
h=0
```

```
while h<=0:
```

```
    h= float (input ("Altura: "))
```

```
S= 2*m.pi*R*(R+h)
```

```
V= m.pi*m.pow(R,2)*h
```

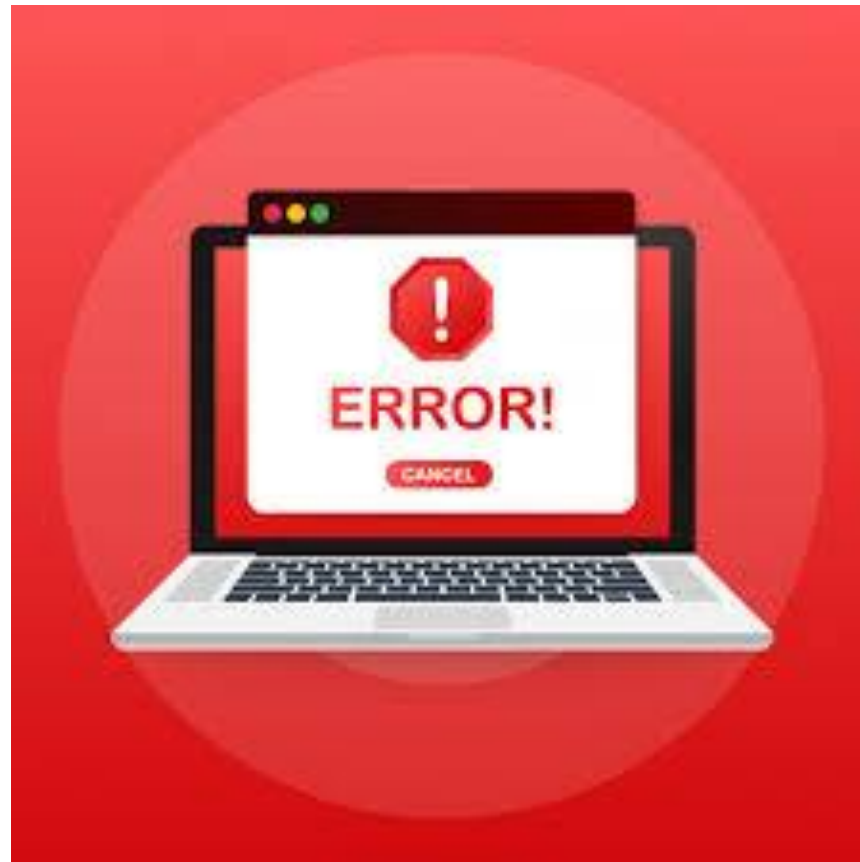
```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```



Lembre-se:
A condição será
sempre **o erro !!**

Validação/Consistência com mensagem de erro





Codificação do looping

Área/Volume do Cilindro com validação

R= 0 # iniciar o raio com um valor **errado!**

while R<=0:

R= float (input ("Raio: "))





Codificação do looping

Área/Volume do Cilindro com validação

R= 0

while R<=0:

R= float (input ("Raio: "))

if R<=0:

print(" Raio inválido!")





Run

main.py × +

```
1 # Área e Volume do Cilindro Reto
2 import math as m
3 R=0
4 while R<=0:
5     R= float (input ("Raio: "))
6     if R<=0:
7         print(" Raio inválido!")
8
9     h=0
10 while h<=0:
11     h= float (input ("Altura: "))
12     if h<=0:
13         print(" Altura inválida!")
14 S= 2*m.pi*R*(R+h)
15 V= m.pi*m.pow(R,2)*h
16 print(" Área Total= ", S)
17 print(" Volume= ", V)
```

>_ Console × Shell × +

```
Raio: 0
Raio inválido!
Raio: -5
Raio inválido!
Raio: 10
Altura: 0
Altura inválida!
Altura: -80
Altura inválida!
Altura: 15
Área Total= 1570.7963267948965
Volume= 4712.38898038469
> |
```


Área e Volume do Cilindro Reto

```
import math as m
```

```
R=0
```

```
while R<=0:
```

```
    R= float (input ("Raio: "))
```

```
    if R<=0:
```

```
        print(" Raio inválido!")
```

```
h=0
```

```
while h<=0:
```

```
    h= float (input ("Altura: "))
```

```
    if h<=0:
```

```
        print(" Altura inválida!")
```

```
S= 2*m.pi*R*(R+h)
```

```
V= m.pi*m.pow(R,2)*h
```

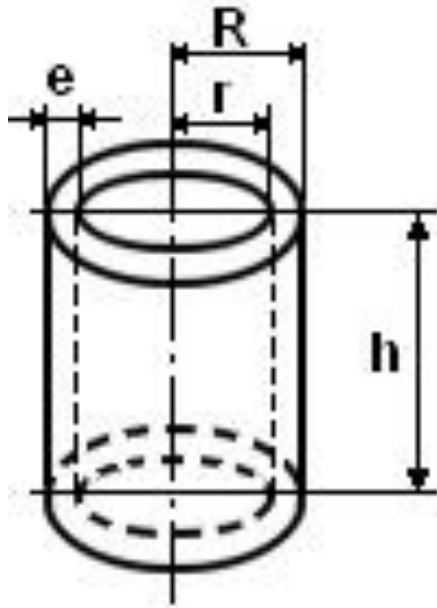
```
print(" Área Total= ", S)
```

```
print(" Volume= ", V)
```



Exercício

- 1) Elaborar um programa para calcular a área lateral e o volume de um cilindro oco. Fazer consistência (Validação) com mensagem de erro. Testar em casa.



$$V = \pi h (R^2 - r^2)$$

$$S_L = 2\pi h (R + r)$$



Boa semana a todos!