



```
#1 - Em um país fictício, o direito ao voto começa aos 16 anos. Crie um programa que verifica se uma pessoa pode votar ou não.  
idade = int(input("Digite a sua idade: "))  
  
if idade >= 16:  
    print("Você JÁ PODE votar!")
```

```
Digite a sua idade: 16  
Você JÁ PODE votar!
```

```
#2 - Uma escola deseja automatizar a verificação do desempenho dos alunos.  
nota1 = float(input("Digite a primeira nota: "))  
nota2 = float(input("Digite a segunda nota: "))  
  
media = (nota1 + nota2) / 2  
  
if media >= 7:  
    print(f"Média: {media:.2f}. Aluno APROVADO.")  
else:  
    print(f"Média: {media:.2f}. Aluno REPROVADO.")
```

```
Digite a primeira nota: 36  
Digite a segunda nota: 34  
Média: 35.00. Aluno APROVADO.
```

```
#3 - Um sistema meteorológico quer classificar a sensação térmica do dia.  
temperatura = float(input("Informe a temperatura em °C: "))  
  
if temperatura < 15:  
    print("Sensação térmica: FRIA.")  
elif 15 <= temperatura < 25:  
    print("Sensação térmica: AGRADÁVEL.")  
elif 25 <= temperatura < 35:  
    print("Sensação térmica: QUENTE.")  
else:  
    # Se temperatura for 35 ou mais  
    print("Sensação térmica: MUITO QUENTE.")
```

```
Informe a temperatura em °C: 3  
Sensação térmica: FRIA.
```

```
#4 - Um painel informativo precisa identificar se um número é par ou ímpar.  
numero = int(input("Digite um número inteiro: "))  
  
if numero % 2 == 0:  
    print(f"O número {numero} é PAR.")  
else:
```

```
print(f"O número {numero} é ÍMPAR.")
```

```
Digite um número inteiro: 2
O número 2 é PAR.
```

```
#5 – Uma empresa deseja reajustar salários com base em faixas de rendimento

salario_atual = float(input("Informe o salário atual: R$ "))

if salario_atual <= 1500:
    percentual_aumento = 0.10 # 10%
elif salario_atual <= 3000:
    percentual_aumento = 0.07 # 7%
else:
    # Salário maior que R$ 3000
    percentual_aumento = 0.05 # 5%

aumento = salario_atual * percentual_aumento
novo_salario = salario_atual + aumento

print(f"Aumento aplicado: {percentual_aumento * 100:.0f}%")
print(f"Novo salário: R$ {novo_salario:.2f}")
```

```
Informe o salário atual: R$ 1500
Aumento aplicado: 10%
Novo salário: R$ 1650.00
```

```
#6 – Um aplicativo financeiro precisa identificar qual de três valores é o maior

n1 = float(input("Digite o primeiro número: "))
n2 = float(input("Digite o segundo número: "))
n3 = float(input("Digite o terceiro número: "))

maior = n1 # Inicializa o maior com o primeiro valor

if n2 > maior:
    maior = n2

if n3 > maior:
    maior = n3

print(f"O maior valor digitado é: {maior}")

# Alternativa mais concisa usando a função max()
# maior = max(n1, n2, n3)
# print(f"O maior valor digitado é: {maior}")
```

```
Digite o primeiro número: 1
Digite o segundo número: 2
Digite o terceiro número: 3
O maior valor digitado é: 3.0
```

#7 – Você está desenvolvendo um calendário digital. Crie um programa que

```
ano = int(input("Digite o ano para verificar: "))

if (ano % 4 == 0 and ano % 100 != 0) or (ano % 400 == 0):
    print(f"O ano {ano} é BISSEXTO.")
else:
    print(f"O ano {ano} NÃO é bissexto.")
```

```
Digite o ano para verificar: 2006
O ano 2006 NÃO é bissexto.
```

#8 – Um cinema utiliza preços diferentes para ingressos infantis e adulto

```
idade_ingresso = int(input("Informe a idade para o ingresso: "))

if idade_ingresso < 12:
    print("Vender ingresso INFANTIL.")
else:
    # 12 anos ou mais
    print("Vender ingresso ADULTO.")
```

```
Informe a idade para o ingresso: 16
Vender ingresso ADULTO.
```

#9 – Um supermercado usa códigos para identificar produtos. Receba um código

```
codigo = input("Digite o código do produto (1, 2 ou 3): ")

if codigo == '1':
    print("Produto: Arroz")
elif codigo == '2':
    print("Produto: Feijão")
elif codigo == '3':
    print("Produto: Óleo")
else:
    print("ERRO: Código de produto inválido.")
```

```
Digite o código do produto (1, 2 ou 3): 3
Produto: Óleo
```

#10 – Um aplicativo exibe mensagens conforme o período do dia. A partir da hora

```
hora = int(input("Informe a hora atual (0 a 23): "))

if 0 <= hora <= 5:
    print("Boa madrugada!")
elif 6 <= hora <= 11:
    print("Bom dia!")
elif 12 <= hora <= 17:
```

```
    print("Boa tarde!")
elif 18 <= hora <= 23:
    print("Boa noite!")
else:
    print("Hora inválida.")
```

```
Informe a hora atual (0 a 23): 6
Bom dia!
```

```
#11 - Uma calculadora avançada deve permitir que o usuário some quantos números quiser.
soma_total = 0
print("Digite números para somar. Digite 0 para encerrar.")

while True:
    try:
        numero = float(input("Número: "))
        if numero == 0:
            break # Encerra o laço quando 0 é digitado
        soma_total += numero
    except ValueError:
        print("Entrada inválida. Por favor, digite um número.")

print(f"A soma total dos números digitados é: {soma_total}")
64
```

```
Digite números para somar. Digite 0 para encerrar.
Número: 12
Número: 12
Número: 3
Número: 4
Número: 5
Número: 6
Número: 7
Número: 88
Número:
Entrada inválida. Por favor, digite um número.
Número: 3
Número: 22
Número: 0
A soma total dos números digitados é: 162.0
```

```
#12 - Um sistema de login precisa validar a senha de acesso. Continue pedindo a senha até que seja digitada corretamente.
senha_correta = "python123"

while True:
    senha_digitada = input("Digite a senha de acesso: ")

    if senha_digitada == senha_correta:
        print("Senha correta! Bem-vindo(a) ao sistema.")
        break # Sai do laço
```

```
        else:  
            print("Senha incorreta. Tente novamente.")
```

```
Digite a senha de acesso: 21  
Senha incorreta. Tente novamente.  
Digite a senha de acesso: 123  
Senha incorreta. Tente novamente.  
Digite a senha de acesso: PYTHON123  
Senha incorreta. Tente novamente.  
Digite a senha de acesso: python123  
Senha correta! Bem-vindo(a) ao sistema.
```

#13 – Uma máquina de simulação deve somar números sequenciais começando em

```
soma = 0  
numero_atual = 1  
  
while soma <= 100:  
    soma += numero_atual  
    print(f"Número utilizado: {numero_atual}, Soma atual: {soma}")  
    numero_atual += 1  
  
# O último print é opcional para mostrar o resultado final.  
print(f"A soma final ultrapassou 100, terminando em: {soma}")
```

```
Número utilizado: 1, Soma atual: 1  
Número utilizado: 2, Soma atual: 3  
Número utilizado: 3, Soma atual: 6  
Número utilizado: 4, Soma atual: 10  
Número utilizado: 5, Soma atual: 15  
Número utilizado: 6, Soma atual: 21  
Número utilizado: 7, Soma atual: 28  
Número utilizado: 8, Soma atual: 36  
Número utilizado: 9, Soma atual: 45  
Número utilizado: 10, Soma atual: 55  
Número utilizado: 11, Soma atual: 66  
Número utilizado: 12, Soma atual: 78  
Número utilizado: 13, Soma atual: 91  
Número utilizado: 14, Soma atual: 105  
A soma final ultrapassou 100, terminando em: 105
```

,#14 – Uma pesquisa está registrando idades. Continue lendo idades até que

```
soma_idades = 0  
contador_idades = 0  
  
print("Digite as idades. Digite um valor negativo para encerrar.")  
  
while True:  
    try:  
        idade = int(input("Idade: "))  
        if idade < 0:  
            break # Valor negativo encerra
```

```
soma_idades += idade
contador_idades += 1
except ValueError:
    print("Entrada inválida. Por favor, digite um número inteiro.")

print("-" * 20)

if contador_idades > 0:
    media = soma_idades / contador_idades
    print(f"Total de idades digitadas: {contador_idades}")
    print(f"Média das idades: {media:.2f}")
else:
    print("Nenhuma idade válida foi digitada.")
```

Digite as idades. Digite um valor negativo para encerrar.

```
Idade: 16
Idade: 12
Idade: 13
Idade: 22
Idade: 25
Idade: 3456
Idade: 124
Idade: 3
Idade: 5
Idade: 6
Idade: 7
Idade: 8
Idade: 9
Idade: 9
Idade: 000
Idade: 12
Idade: 20
Idade: 0
Idade: 23
Idade: 16
Idade: 59
Idade: 8*
Entrada inválida. Por favor, digite um número inteiro.
Idade: 59
Idade: 56
Idade: 55
Idade: 65
Idade: 8589
Idade: 6855555
Idade: 5252
Idade: 523
Idade: 523546323
Idade: 52
Idade:
Entrada inválida. Por favor, digite um número inteiro.
Idade:
Entrada inválida. Por favor, digite um número inteiro.
Idade: 3
```

KeyboardInterrupt

```
#15 - Um cronômetro digital deve realizar uma contagem regressiva personalizada

try:
    contador = int(input("Digite o número inicial da contagem regressiva:"))

    if contador < 0:
        print("O número inicial deve ser não negativo.")
    else:
        while contador >= 0:
            print(contador)
            contador -= 1
        print("Contagem regressiva finalizada!")
except ValueError:
    print("Entrada inválida. Por favor, digite um número inteiro.")
```

```
Digite o número inicial da contagem regressiva: 1
```

```
1
```

```
0
```

```
Contagem regressiva finalizada!
```

```
#16 - Um painel de testes precisa exibir todos os números de 1 a 50. Utilize o comando range(1, 51)
```

```
# O range(1, 51) inclui o 1 e vai ATÉ o 51 (ou seja, até 50)
print("Números de 1 a 50:")
for numero in range(1, 51):
    print(numero)
```

```
Números de 1 a 50:
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

```
10
```

```
11
```

```
12
```

```
13
```

```
14
```

```
15
```

```
16
```

```
17
```

```
18
```

```
19
```

```
20
```

```
21
```

```
22
```

```
23
```

```
24
```

```
25
```

```
26
```

```
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50
```

```
#17 – Um módulo educacional precisa gerar a tabuada de um número escolhido.  
  
try:  
    num_tabuada = int(input("Digite o número para gerar a tabuada (1 a 10):"))  
  
    print(f"\nTabuada de {num_tabuada}:")  
    for i in range(1, 11): # i vai de 1 a 10  
        resultado = num_tabuada * i  
        print(f"{num_tabuada} x {i:2} = {resultado}") # 0 :2 é para alinhar  
except ValueError:  
    print("Entrada inválida. Por favor, digite um número inteiro.")
```

Digite o número para gerar a tabuada (1 a 10): 2

Tabuada de 2:  
2 x 1 = 2  
2 x 2 = 4  
2 x 3 = 6  
2 x 4 = 8  
2 x 5 = 10  
2 x 6 = 12  
2 x 7 = 14  
2 x 8 = 16  
2 x 9 = 18  
2 x 10 = 20

```
#18 – Um programa de análise de texto deve mostrar cada letra de uma palavra.  
palavra = input("Digite uma palavra: ")
```

```
print("Caracteres da palavra:")
for letra in palavra:
    print(letra)
```

```
Digite uma palavra: uma
Caracteres da palavra:
u
m
a
```

```
#19 - Em um formulário digital, serão digitadas 5 idades. Use um laço for
```

```
contador_maiores = 0
NUMERO_IDADES = 5

print(f"Digite {NUMERO_IDADES} idades:")

for i in range(NUMERO_IDADES):
    try:
        idade = int(input(f"Idade #{i + 1}: "))
        if idade > 18:
            contador_maiores += 1
    except ValueError:
        print("Idade inválida. Esta entrada será ignorada.")

print(f"\nTotal de idades maiores que 18: {contador_maiores}")
```

```
Digite 5 idades:
Idade #1: 12
Idade #2: 16
Idade #3: 18
Idade #4: 26
Idade #5: 52
```

```
Total de idades maiores que 18: 2
```

```
#20 - Um sistema estatístico deve analisar números fornecidos pelo usuário
```

```
NUMERO_DE_VALORES = 7
maior_numero = None # Inicializa como vazio

print(f"Digite {NUMERO_DE_VALORES} números:")

for i in range(NUMERO_DE_VALORES):
    try:
        numero = float(input(f"Número #{i + 1}: "))

        # Na primeira iteração, define o primeiro número como o maior.
        if i == 0 or numero > maior_numero:
            maior_numero = numero

    except ValueError:
        print("Entrada inválida. Tente novamente com um número válido.")
```

```
if maior_numero is not None:  
    print(f"\nO maior número digitado foi: {maior_numero}")  
else:  
    print("Não foi possível determinar o maior número (nenhuma entrada válida)")
```

Digite 7 números:

Número #1: 15  
Número #2: 165  
Número #3: 16  
Número #4: 15  
Número #5: 56  
Número #6: 296  
Número #7: 26

O maior número digitado foi: 296.0

#21 – Um app de produtividade quer identificar qual de quatro tarefas possuem o maior comprimento

```
NUMERO_TAREFAS = 4  
tarefa_mais_longa = ""  
max_comprimento = -1  
  
print(f"Digite o nome de {NUMERO_TAREFAS} tarefas:")  
  
for i in range(NUMERO_TAREFAS):  
    nome_tarefa = input(f"Tarefa #{i + 1}: ")  
  
    comprimento_atual = len(nome_tarefa)  
  
    if comprimento_atual > max_comprimento:  
        max_comprimento = comprimento_atual  
        tarefa_mais_longa = nome_tarefa  
  
print(f"\nA tarefa com o nome mais longo é: **{tarefa_mais_longa}** (com {max_comprimento} caracteres)")
```

Digite o nome de 4 tarefas:

Tarefa #1: limpar a casa  
Tarefa #2: atividades escolares  
Tarefa #3: lavar o cabelo  
Tarefa #4: mudar a cama de lugar

A tarefa com o nome mais longo é: \*\*mudar a cama de lugar \*\* (com 22 caracteres)

#22 – Um programa de cálculos deve somar todos os números pares entre 2 e 100.

```
soma_pares = 0  
  
for par in range(2, 101, 2):  
    soma_pares += par  
  
print(f"A soma de todos os números pares entre 2 e 100 é: {soma_pares}")
```

A soma de todos os números pares entre 2 e 100 é: 2550

#23 – Um módulo de análise precisa classificar uma lista de valores. Leia

```
contador_pos = 0
contador_neg = 0
contador_zero = 0
NUMERO_VALORES = 10

print(f"Digite {NUMERO_VALORES} números:")

for i in range(NUMERO_VALORES):
    try:
        numero = float(input(f"Valor #{i + 1}: "))

        if numero > 0:
            contador_pos += 1
        elif numero < 0:
            contador_neg += 1
        else:
            contador_zero += 1
    except ValueError:
        print("Entrada inválida. Esta entrada será ignorada.")

print("-" * 20)
print(f"Resumo da Classificação:")
print(f"Positivos: {contador_pos}")
print(f"Negativos: {contador_neg}")
print(f"Iguais a zero: {contador_zero}")
```

Digite 10 números:

```
Valor #1: 1
Valor #2: 2
Valor #3: 3
Valor #4: 4
Valor #5: 5
Valor #6: 6
Valor #7: 32
Valor #8: 21
Valor #9: 56
Valor #10: 21
```

-----

Resumo da Classificação:

```
Positivos: 10
Negativos: 0
Iguais a zero: 0
```

#24 – Uma escola quer filtrar nomes que comecem com determinadas letras.

```
NUMERO_NOMES = 6
```

```
print(f"Digite {NUMERO_NOMES} nomes:")
print("\nNomes que começam com 'A' ou 'C':")

for i in range(NUMERO_NOMES):
    nome = input(f"Nome #{i + 1}: ").strip() # Remove espaços em branco

    # Verifica se a primeira letra, após ser convertida para maiúscula, é
    if nome.upper().startswith('A') or nome.upper().startswith('C'):
        print(f"-> {nome}")
```

Digite 6 nomes:

```
Nomes que começam com 'A' ou 'C':
Nome #1: ana
-> ana
Nome #2: calor
-> calor
Nome #3: andressa
-> andressa
Nome #4: carol]
-> carol]
Nome #5: eriqoe
Nome #6: iza
```

```
#25 - Uma caixa registradora digital permite que o usuário digite valores e os some.

soma_valores = 0

print("Digite valores para a caixa registradora. Digite 999 para encerrar.")

while True:
    try:
        valor = float(input("Valor do produto: R$ "))

        if valor == 999:
            break

        if valor < 0:
            print("Valor inválido (negativo) e ignorado.")
            continue

        soma_valores += valor
    except ValueError:
        print("Entrada inválida. Digite um número.")

print(f"\nCaixa fechada. Total a pagar: R$ {soma_valores:.2f}")
```

```
Digite valores para a caixa registradora. Digite 999 para encerrar.
Valor do produto: R$ 12
Valor do produto: R$ 23
Valor do produto: R$ 56
Valor do produto: R$ 45
Valor do produto: R$ 26
Valor do produto: R$ 158
```

```
Valor do produto: R$ 528  
Valor do produto: R$ 569  
Valor do produto: R$ 999
```

```
Caixa fechada. Total a pagar: R$ 1417.00
```

```
#26 - Um painel de processamento deve listar números de 1 a 20, mas ignorar
```

```
print("Números de 1 a 20, exceto múltiplos de 4:")
```

```
for numero in range(1, 21):
```

```
    # Se o número for divisível por 4 (múltiplo de 4)
```

```
    if numero % 4 == 0:
```

```
        continue # Pula a impressão e vai para a próxima iteração
```

```
    print(numero)
```

```
Números de 1 a 20, exceto múltiplos de 4:
```

```
1
```

```
2
```

```
3
```

```
5
```

```
6
```

```
7
```

```
9
```

```
10
```

```
11
```

```
13
```

```
14
```

```
15
```

```
17
```

```
18
```

```
19
```

```
#27 - Crie um sistema de login com limite de tentativas. O usuário terá n
```

```
senha_correta = "python123"
```

```
MAX_TENTATIVAS = 3
```

```
for tentativa in range(1, MAX_TENTATIVAS + 1):
```

```
    senha_digitada = input(f"Tentativa {tentativa}/{MAX_TENTATIVAS} - Dig
```

```
    if senha_digitada == senha_correta:
```

```
        print("\nAcesso liberado! Bem-vindo(a).")
```

```
        break # Senha correta, encerra o laço
```

```
    else:
```

```
        print("Senha incorreta.")
```

```
else:
```

```
    # Este 'else' do for é executado se o laço terminar SEM um 'break' (o  
    print("\nNúmero máximo de tentativas excedido. Acesso bloqueado.")
```

```
Tentativa 1/3 - Digite a senha: 123
```

```
Senha incorreta.
```

```
Tentativa 2/3 - Digite a senha: 456.
```

```
Senha incorreta.  
Tentativa 3/3 - Digite a senha: 3  
Senha incorreta.
```

```
Número máximo de tentativas excedido. Acesso bloqueado.
```

```
#28 – Desenvolva um programa que leia 10 números e some apenas os valores  
  
soma_positivos = 0  
NUMERO_LEITURAS = 10  
  
print(f"Digite {NUMERO_LEITURAS} números para somar apenas os positivos."  
  
for i in range(NUMERO_LEITURAS):  
    try:  
        numero = float(input(f"Número #{i + 1}: "))  
  
        if numero < 0:  
            print("Número negativo ignorado.")  
            continue # Ignora o número negativo e volta para o início do  
  
        soma_positivos += numero  
  
    except ValueError:  
        print("Entrada inválida. Esta entrada será ignorada."  
  
print(f"\nA soma de todos os números positivos digitados é: {soma_positivos}")
```

```
Digite 10 números para somar apenas os positivos.  
Número #1: 165  
Número #2: 265  
Número #3: 2  
Número #4: 285  
Número #5: 285  
Número #6: 285  
Número #7: 5485  
Número #8: 285  
Número #9: 2  
Número #10:  
Entrada inválida. Esta entrada será ignorada.
```

```
A soma de todos os números positivos digitados é: 7059.00
```

```
#29 – Simule um acumulador que percorre números até que a soma ultrapasse 2000.  
  
soma_acumulada = 0  
LIMITE = 2000  
numero_atual = 1  
  
print(f"Acumulando números até ultrapassar {LIMITE}:")  
  
while True:
```

```
# Verifica a condição de parada antes de somar.  
if soma_acumulada > LIMITE:  
    print(f"Soma ultrapassou {LIMITE}!")  
    break # Encerra o laço  
  
print(f"Somando {numero_atual}. Soma atual: {soma_acumulada}")  
soma_acumulada += numero_atual  
numero_atual += 1  
  
print(f"\nSoma final: {soma_acumulada}")
```

```
Acumulando números até ultrapassar 200:  
Somando 1. Soma atual: 0  
Somando 2. Soma atual: 1  
Somando 3. Soma atual: 3  
Somando 4. Soma atual: 6  
Somando 5. Soma atual: 10  
Somando 6. Soma atual: 15  
Somando 7. Soma atual: 21  
Somando 8. Soma atual: 28  
Somando 9. Soma atual: 36
```