

{reprograma}

PYTHON PARA ANÁLISE DE DADOS

semana 3



Boas vindas, mulheres incríveis!

Só de estarem aqui, num sábado de
manhã, vocês já são vitoriosas

Quem sou eu?

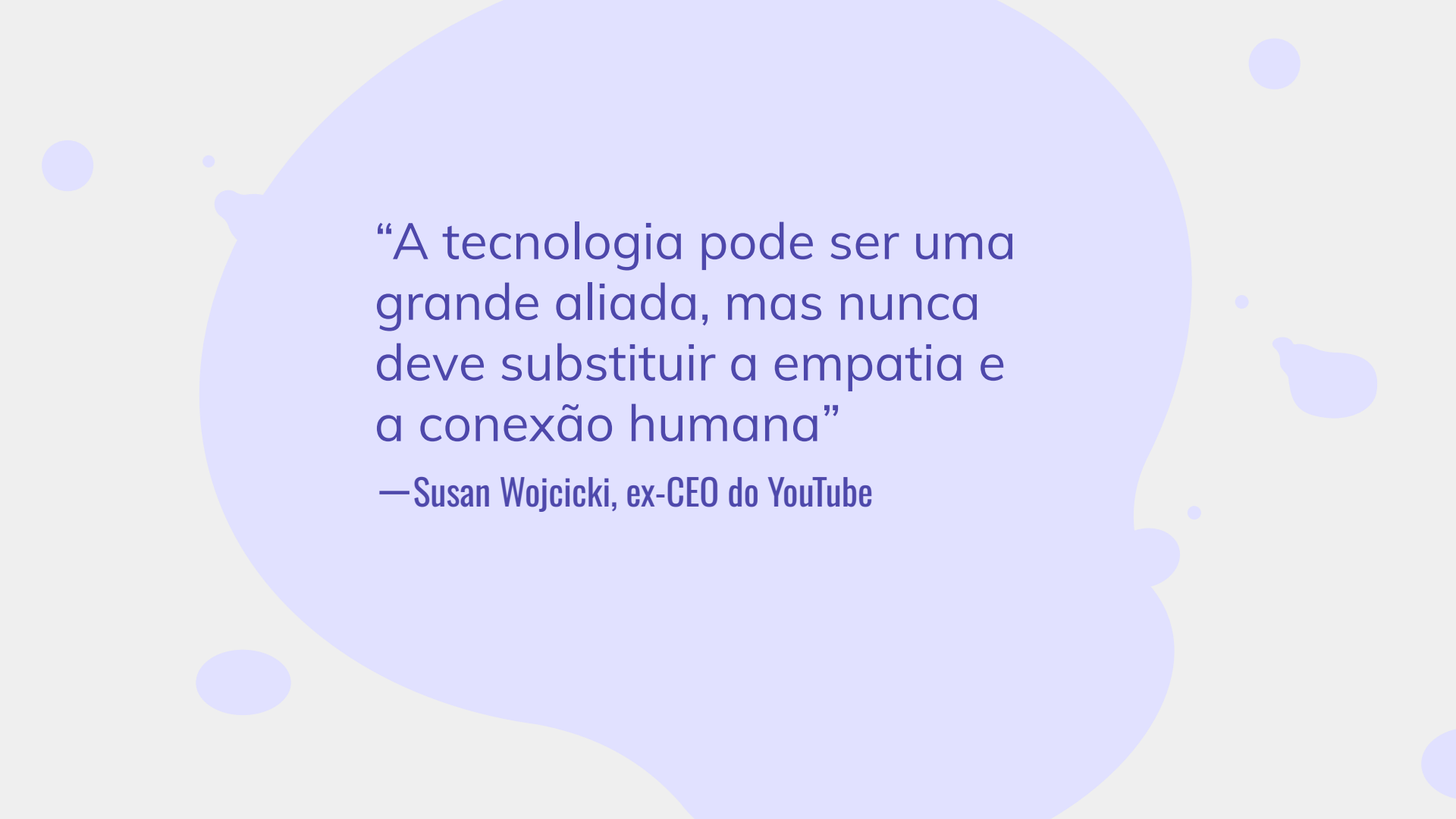
Erika Campos

- A humana do Fred
- Designer gráfica + jornalista + curiosa = em 2015 me envolvi mais com tecnologia
- Co-fundadora do PyLadies São Paulo
- Graduanda em Análise de Sistemas (ufa! só falta o TCC)
- Hoje trabalho ajudando pessoas iniciantes e entrar no mundo do desenvolvimento de software



Como estamos hoje?





“A tecnologia pode ser uma grande aliada, mas nunca deve substituir a empatia e a conexão humana”

—Susan Wojcicki, ex-CEO do YouTube

- Não fiquem com dúvida! Perguntem!
- Sempre que possível, ajude a colega
- Ver o rosto de vocês me ajuda a saber se tá tudo bem aí
- Quer falar? Levanta a mão ou manda no chat
- Microfones desligados enquanto não estiverem falando

**Nossos
combinados
do dia!**

Semana 2

- Valores, variáveis e tipos de valores
- Inputs
- Operações
- Formatação de strings
- Indentação
- Funções



Semana 3

- Condicionais e Operadores Relacionais e Lógicos
- Erros e seus Tipos
- Usando o Debug
- Try e Except



Condicionais e Operadores Relacionais e Lógicos



O que é uma condição?

MICHAELIS

Português

Inglês

Espanhol

Alemão

Italiano

Dicionário Brasileiro da Língua Portuguesa

[Sobre o dicionário](#) [Como consultar](#) [Noções gramaticais](#) [Créditos](#)



Português Brasileiro ▾

Digite o termo desejado

condição

con·di·ção

sf

5 Circunstância necessária para que ocorra determinado fato ou situação.

5 Circunstância necessária para que ocorra determinado fato ou situação.

6 Situação de uma pessoa perante a vida ou o grupo social.

7 Estado de saúde (geralmente no plural).

O que isso
quer dizer?

SE der
faça isso aqui

SENÃO
faça esse outro

Características da condicional

SE

faça isso aqui

SENÃO

faça esse outro



Estou com vontade de fazer brigadeiro

Tenho todos os
ingredientes?

Se tenho os ingredientes:
faço brigadeiro

Senão:
fico com vontade



Antes de
avançarmos,
lembre-se:

Verdadeiro \rightarrow True

Falso \rightarrow False



São apenas duas respostas possíveis

True

Se a relação entre os dois valores for verdadeira.

Por exemplo:

$1 < 3$

$1 \neq 3$

$1 == 1$

False

Se a relação entre os dois valores for falsa.

Por exemplo:

$1 > 3$

$1 == 3$

$5 < 1$

Estou com vontade de fazer brigadeiro

Tenho todos os ingredientes?

Se tenho os ingredientes:
faço brigadeiro

Senão:
fico com vontade

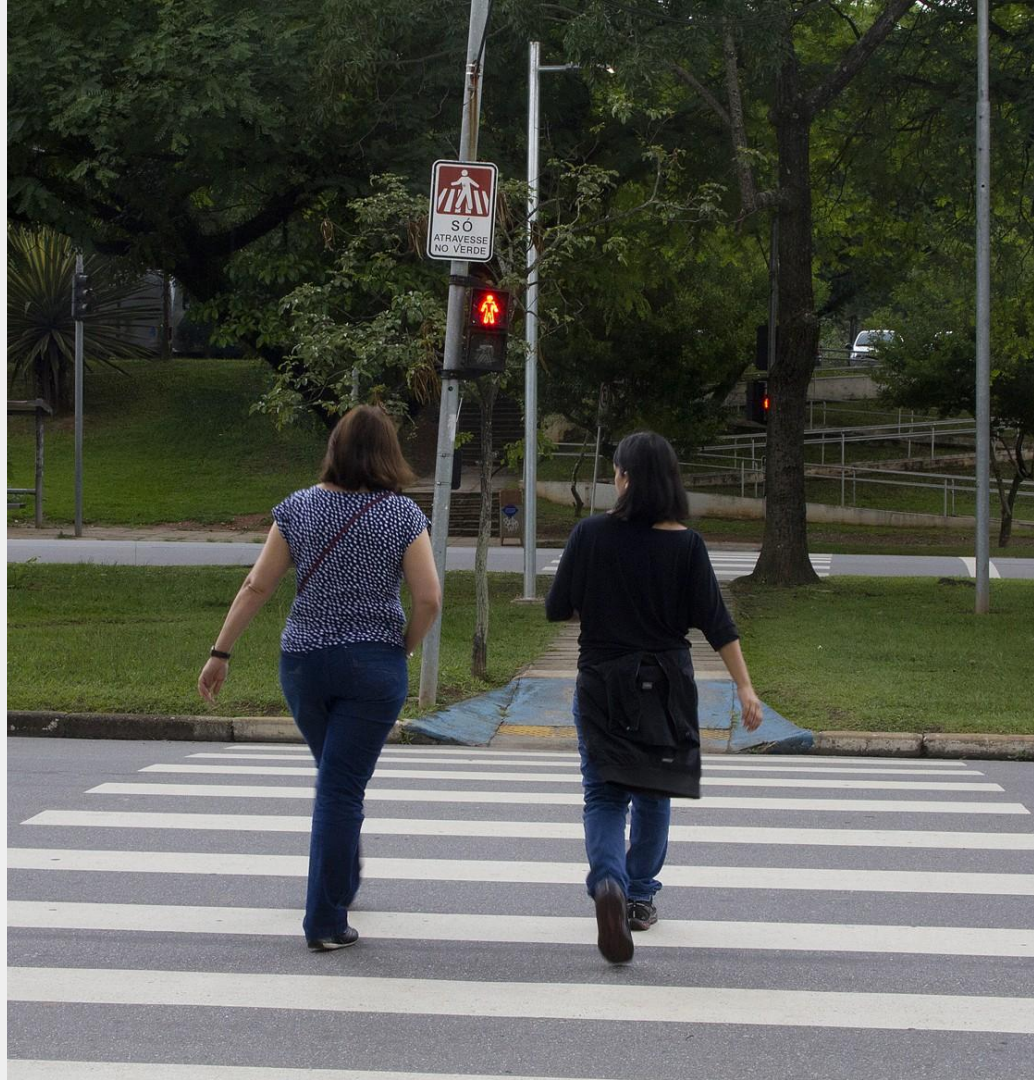


Como seria para atravessar a rua na faixa?

O semáforo está aberto
para pedestre?

Se o semáforo está aberto:
atravesso

Senão:
guardo



Python é uma linguagem escrita em inglês

se, em inglês, é **if**
senão, em inglês é **else**



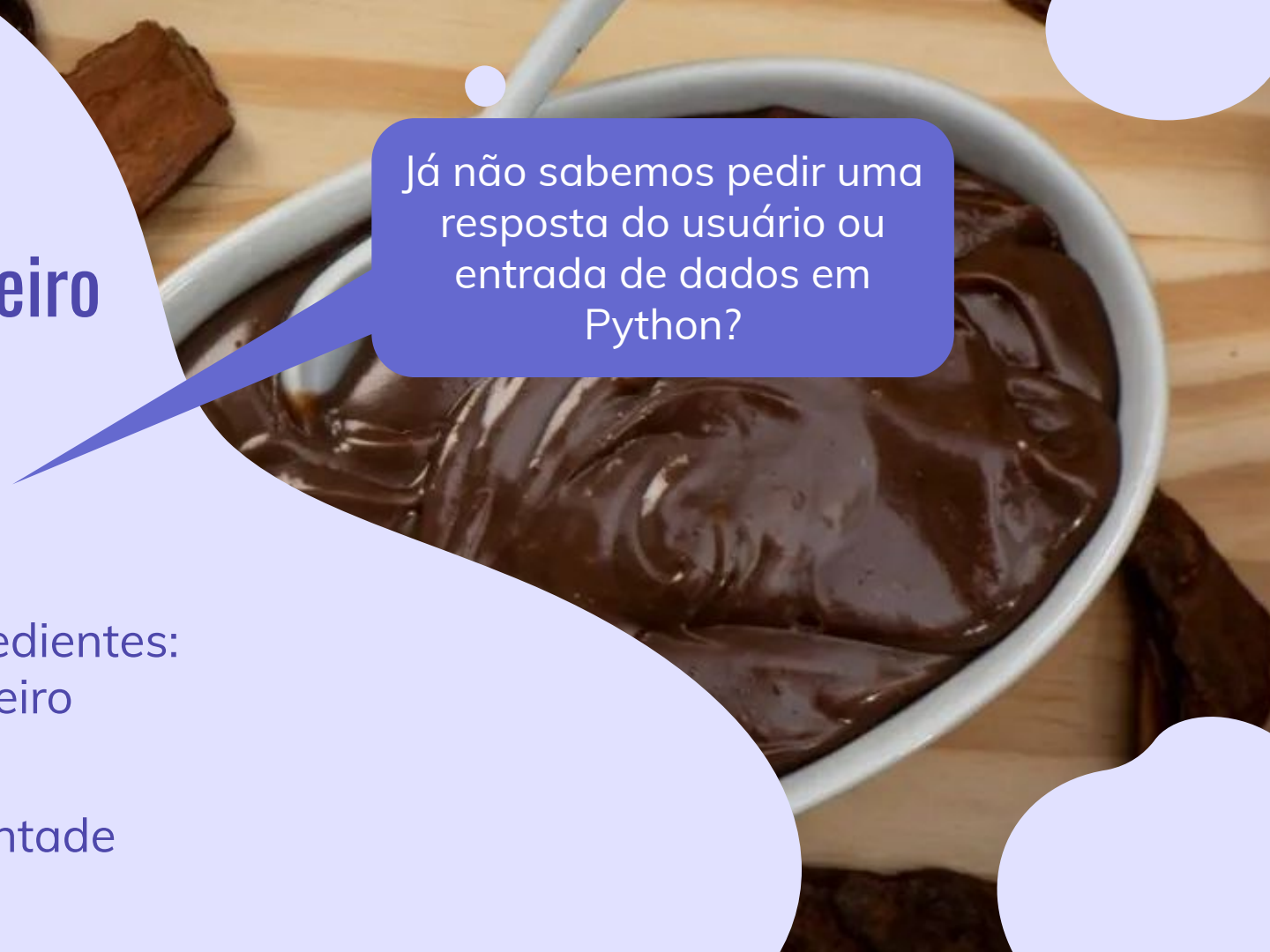
Estou com vontade de fazer brigadeiro

Tenho todos os ingredientes?

if tenho os ingredientes:
 faço brigadeiro

else:
 fico com vontade

Já não sabemos pedir uma resposta do usuário ou entrada de dados em Python?



Estou com vontade de fazer brigadeiro

resposta = input("Tenho todos os ingredientes?
sim ou não: ")

if tenho os ingredientes:
 faço brigadeiro

else:
 fico com vontade



Isso é o código?
Como isso ficaria em Python?



Estou com vontade de fazer brigadeiro

```
resposta = input("Tenho todos os ingredientes? sim ou não: ")
```

```
if resposta == "sim":  
    print("faço brigadeiro")  
else:  
    print("fico com vontade")
```

== ou = ????



Operadores relacionais

Demonstram a relação de um valor com o outro. A resposta que recebemos deve ser verdadeira (True) ou falsa (False)

==	igual	1 == 1	True
		1 == 3	False
!=	diferente	1 != 3	True
		1 != 1	False
>	maior que	3 > 1	True
		1 > 3	False
<	menor que	1 < 3	True
		3 < 1	False

Tá, e como eu
vou usar isso?



Sites de bebidas alcoólicas

Você só pode acessar se tiver mais de 18 anos

Vamos fazer um programa pra saber se o usuário é maior de idade?



Operadores relacionais

Demonstram a relação de um valor com o outro. A resposta que recebemos deve ser verdadeira (True) ou falsa (False)

==	igual	1 == 1	True
		1 == 3	False
!=	diferente	1 != 3	True
		1 != 1	False
>	maior que	3 > 1	True
		1 > 3	False
<	menor que	1 < 3	True
		3 < 1	False
>=	maior ou igual	3 >= 1	True
		3 >= 3	True
<=	menor ou igual	1 <= 3	True
		3 <= 3	True



Vamos decidir se as alunas foram aprovadas ou reprovadas

APROVAÇÃO POR NOTA

Peça para a pessoa
entrar com a nota de
0 a 10 e imprima se a
aluna foi aprovada ou
reprovada

A nota mínima de
aprovação é 7.



E se tivermos:
aprovação, reprovação
e **recuperação?**



else + if = elif

Parecido com:
senão, se

Uso do **elif**

Vamos fazer um exercício em que a gente peça a nota final.

Regras:

- A nota mínima para aprovação é 7.
- Se a aluna tirou ao menos 5, ela fica de recuperação.
- Abaixo de 5, ela está reprovada

Uso do **elif**

Se a aluna tirou pelo menos 7:
ela está aprovada

Senão, se ela tirou pelo menos 5:
ela está em recuperação

Senão:
ela está reprovada



Vamos decidir se as alunas foram aprovadas ou reprovadas

APROVAÇÃO POR NOTA

Peça para a pessoa entrar com a nota de 1 a 10 e imprima se a aluna foi aprovada ou reprovada
A nota mínima de aprovação é 7.

APROVAÇÃO POR NOTA E PRESENÇA

Aqui não tem recuperação, mas precisamos checar se a aluna teve presença em pelo menos 75% das aulas.

Operadores lógicos

Permitem combinar a comparação.

A resposta que recebemos também deve ser verdadeira (True) ou falsa (False)

and	e	O primeiro e o segundo precisam ser True	$1 \neq 3$ and $1 < 3$	True
			$1 \neq 3$ and $1 > 3$	False



Vamos decidir se as alunas foram aprovadas ou reprovadas

APROVAÇÃO POR NOTA

Peça para a pessoa entrar com a nota de 0 a 10 e imprima se a aluna foi aprovada ou reprovada

A nota mínima de aprovação é 7.

APROVAÇÃO POR NOTA E PRESENÇA

Parecido com o anterior, mas precisamos checar se a aluna também frequentou pelo menos 75% das aulas.

APROVAÇÃO POR NOTA OU PRESENÇA

Com base no anterior, mas a aluna pode ser aprovada por nota ou por presença, não precisa ser nos dois.

Operadores lógicos

Permitem combinar a comparação.

A resposta que recebemos também deve ser verdadeira (True) ou falsa (False)

and	e	O primeiro e o segundo precisam ser True	$1 \neq 3 \text{ and } 1 < 3$	True
			$1 \neq 3 \text{ and } 1 > 3$	False
or	ou	O primeiro ou o segundo precisam ser True	$1 == 3 \text{ or } 1 < 3$	True
			$1 == 3 \text{ or } 1 > 3$	False



Operadores lógicos

Permitem combinar a comparação.

A resposta que recebemos também deve ser verdadeira (True) ou falsa (False)

and	e	O primeiro e o segundo precisam ser True	1 != 3 and 1 < 3	True
			1 != 3 and 1 > 3	False
or	ou	O primeiro ou o segundo precisam ser True	1 == 3 or 1 < 3	True
			1 == 3 or 1 > 3	False
not	não	Inverte o booleano	not False	True
			not True	False



Condicionais aninhadas

Sim, aninhado, como
ninho! Piu piu

E quando uma resposta depende da outra?

Se a aluna ficou em recuperação, qual nota ela tirou
na recuperação?

Com essa nota, ela foi aprovada ou reprovada?



Tomei um **ERRO!**
E agora???





**A partir de agora, os erros devem
ser seus melhores amigos**

Eles vão aparecer, sempre!
Ouça/leia o que eles têm a dizer



Erro se sintaxe

Provavelmente o mais comum de todos

Numa conversa, tudo bem digitar errado, mas o computador entende tudo tin-tin por tin-tin

Se ele não interpretar certinho, o erro vai aparecer!

Então, vamos “forçar” um ‘SyntaxError’



Erro se sintaxe

Provavelmente o mais comum de todos

```
6_ex_SyntaxError.py > ...
1  nota = int(input("Qual a nota? "))
2
3  if nota >= 7:
4      print("Aluna aprovada")
5  else
6      print("Aluna reprovada")
7
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py
File "/root/reprograma/6_ex_SyntaxError.py", line 5
else
^
SyntaxError: expected ':'

Erro de sintaxe: esperava ':'



Erro se sintaxe

Provavelmente o mais comum de todos

```
6_ex_SyntaxError.py > ...
1  semaforo = input("O semáforo está aberto ou fechado? ")
2
3  if semaforo == "aberto":
4      print("Posso atravessar")
5  else:
6      print("Não posso atravessar")
7
```

PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
root@DESKTOP-G5D2ON2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py
File "/root/reprograma/6_ex_SyntaxError.py", line 1
    semaforo = input("O semáforo está aberto ou fechado? ")
                    ^
SyntaxError: '(' was never closed
```

Erro de sintaxe: '(' nunca foi fechado



Erro de sintaxe: sintaxe inválida. Talvez você quis dizer '==' ou ':' em vez de '='?

```
1 semaforo = input("O semáforo está aberto ou fechado? ")
2
3 if semaforo == "aberto":
4     print("Posso atravessar")
5 else:
6     print("Não posso atravessar")
7
```

1

DEBUG CONSOLE

TERMINAL

PORTS

```

root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py
File "/root/reprograma/6_ex_SyntaxError.py", line 3
    if semaforo = "aberto":
    ^^^^^^^^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax. Maybe you meant '==' or ':=' instead of '='?

```



Erro se sintaxe

Provavelmente o mais comum de todos

```
6_ex_SyntaxError.py > ...
1  nota = float(input("Qual a nota? "))
2  presenca = float(input("Qual a porcentagem de presença? "))
3
4  if nota >= 7 and presenca >= 75:
5      print("Aluna aprovada")
6  else:
7      print("Aluna reprovada")
8
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
root@DESKTOP-G5D2ON2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py
Qual a nota? 8
Qual a porcentagem de presença? 80
Traceback (most recent call last):
  File "/root/reprograma/6_ex_SyntaxError.py", line 4, in <module>
    if nota >= 7 and presenca >= 75:
NameError: name 'presença' is not defined. Did you mean: 'presenca'?
```

Erro de nome: nome 'presença' não está definido
Você quis dizer 'presenca' ?



Erro se sintaxe

Provavelmente o mais comum de todos

```
6_ex_SyntaxError.py > ...  
1  nota = float(input("Qual a nota? "))  
2  presenca = float(input("Qual a porcentagem de presença? "))  
3  
4  if nota >= 7 and presenca >= 75:  
5  |   print("Aluna aprovada")  
6  else:  
7  |   print("Aluna reprovada")  
8
```

PROBLEMS

2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py  
File "/root/reprograma/6_ex_SyntaxError.py", line 5  
    print("Aluna aprovada")  
      ^
```

SyntaxError: unterminated string literal (detected at line 5)

Erro de sintaxe: string literal não terminada (detectada na linha 5)



Erro de indentação

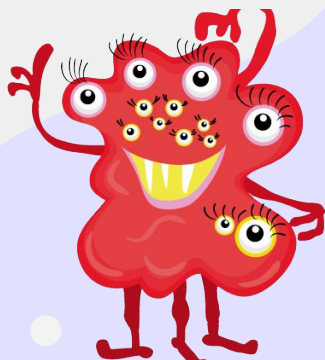
Um erro também de sintaxe, característico de Python

```
6_ex_SyntaxError.py > ...
1  nota = float(input("Qual a nota? "))
2  presenca = float(input("Qual a porcentagem de presença? "))
3
4  if nota >= 7 and presenca >= 75:
5      print("Aluna aprovada")
6  else:
7      print("Aluna reprovada")
8
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/6_ex_SyntaxError.py
File "/root/reprograma/6_ex_SyntaxError.py", line 7
    print("Aluna reprovada")
    ^
IndentationError: expected an indented block after 'else' statement on line 6
```

Erro de indentação: era esperada uma indentação depois do bloco 'else' declarado na linha 6



Erro de execução

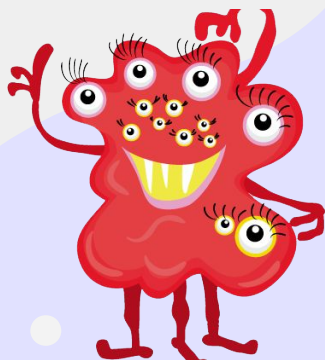
Tá tudo certo,
só que não...

```
7_mais_erros.py > ...  
1 print("Vamos rachar a conta")  
2  
3 valortotal = float(input("Qual o valor gasto? R$ "))  
4 amigos = int(input("Quantas pessoas? "))  
5  
6 print("Cada pessoa deve pagar R$", valortotal/amigos)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/7_mais_erros.py  
Vamos rachar a conta  
Qual o valor gasto? R$ 120  
Quantas pessoas? 0  
Traceback (most recent call last):  
  File "/root/reprograma/7_mais_erros.py", line 6, in <module>  
    print("Cada pessoa deve pagar R$", valortotal/amigos)  
ZeroDivisionError: float division by zero
```

Erro na Divisão por Zero: um tipo float está sendo dividido por zero



Erro de execução

Tá tudo certo,
só que não...

```
4b_nota_presenca.py X
4b_nota_presenca.py > ...
1  nota = float(input("Qual a nota? "))
2  presenca = float(input("Qual a porcentagem de presenca? "))
3
4  if nota >= 7 and presenca >= 75:
5      print("Aluna aprovada")
6  else:
7      print("Aluna reprovada")
8

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  1
root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprogr
rama/4b_nota_presenca.py
Qual a nota? 8
Qual a porcentagem de presenca? cem
Traceback (most recent call last):
  File "/root/reprograma/4b_nota_presenca.py", line 2, in <m
odule>
    presenca = float(input("Qual a porcentagem de presenca?
"))
ValueError: could not convert string to float: 'cem'
```

Erro de Valor: não é possível converter string para float: 'cem'



Erro semântico

Malvado! Te faz pensar: “o que tá acontecendo???”

```
7_mais_erros.py > ...
1  nota = float(input("Qual a nota? "))
2  presenca = float(input("Qual a porcentagem de presença? "))
3
4  if nota >= 7 and presenca >= 75:
5      print("Aluna aprovada")
6  elif nota >= 5:
7      if presenca == 100:
8          print("Aluna aprovada")
9      elif presenca >= 75:
10         print("Aluna em recuperação")
11 else:
12     print("Aluna reprovada")
13
```

PROBLEMS 2

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
root@DESKTOP-G5D2ON2:~/reprograma# /bin/python3 /root/reprograma/7_mais_erros.py
Qual a nota? 6.8
Qual a porcentagem de presença? 50
root@DESKTOP-G5D2ON2:~/reprograma#
```

Nada acontece...
Ahhhhhhhhhhhhhhhhhh

Tomou erro?
Não desista!



Continue a codar...
Continue a codar...

Debugger

Pra te ajudar a entender o
que está acontecendo

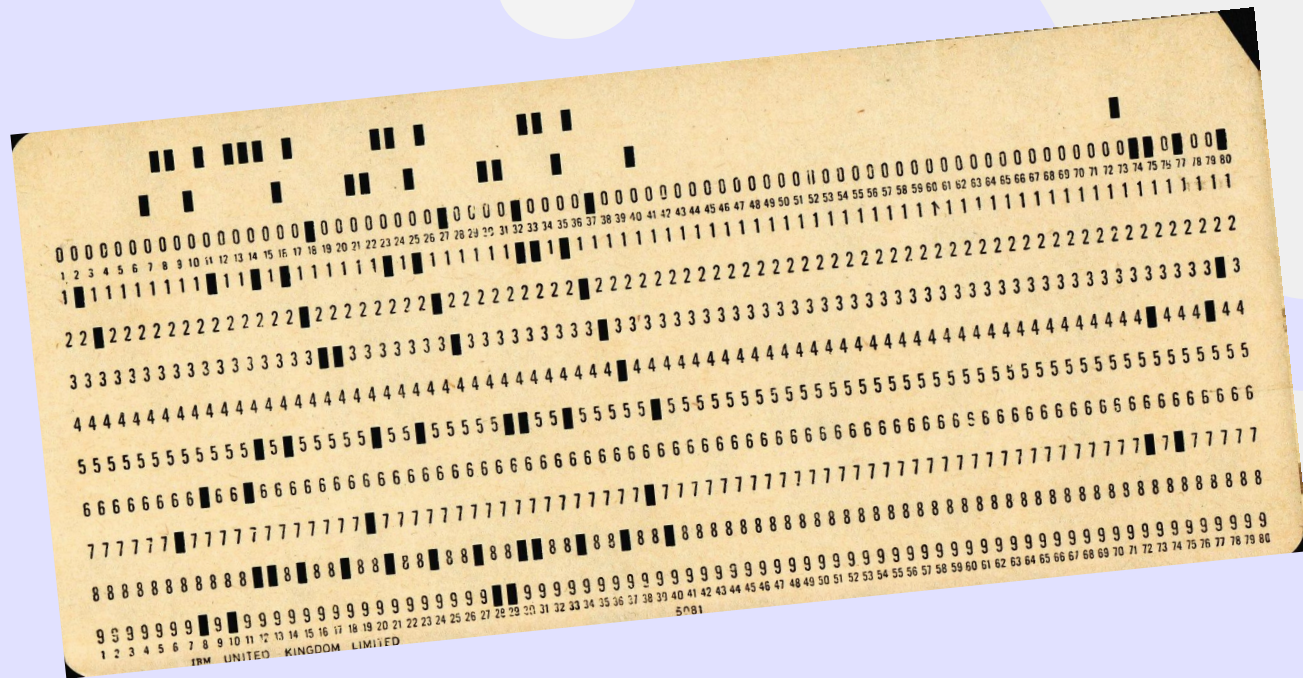


O que é
um *bug*?



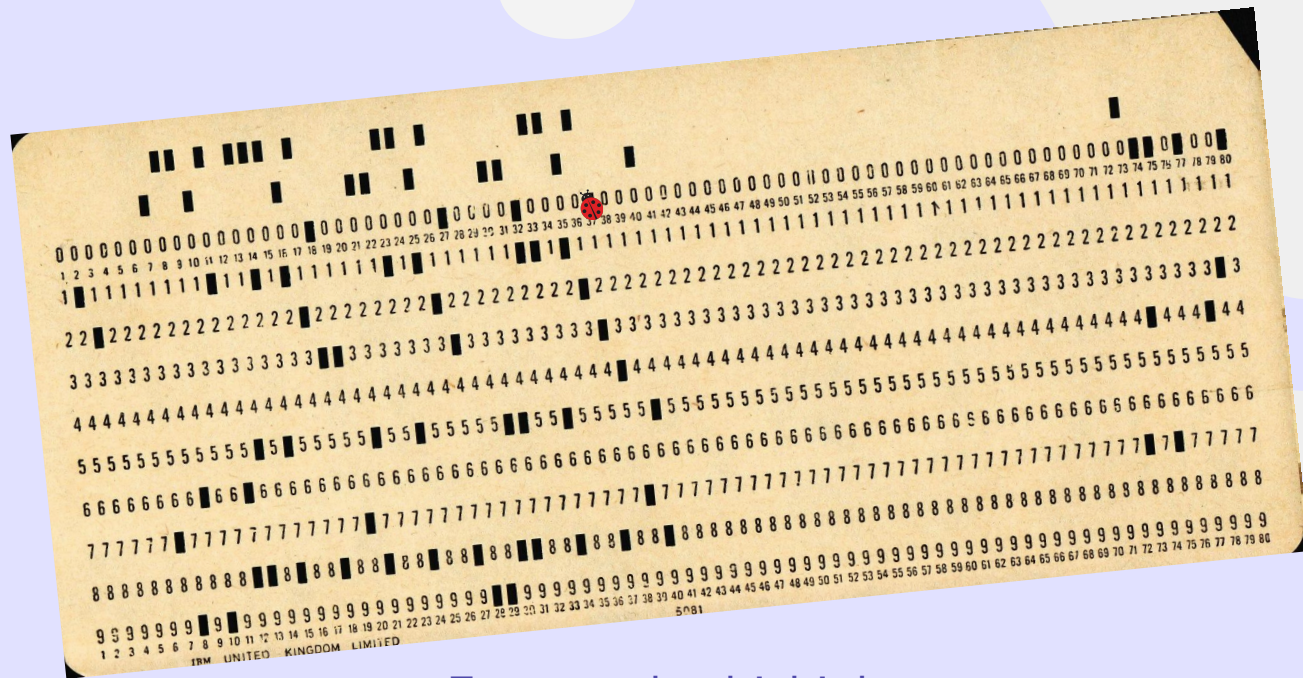
Senta que lá vem história

Final de
1800



Essa era a forma como
se gravava dados

Papel tem
traças...



E se um dos bichinhos comesse o
papel até furar?
E se o bichinho ficasse ali
tampando o buraco?



Olha o bug!



As caçadoras de bugs

Segue a receita
Vai ser sucesso!

No seu código,
clique à esquerda
da linha que quer
iniciar o debug



The screenshot shows a code editor window titled 'exemplo.py'. The code is as follows:

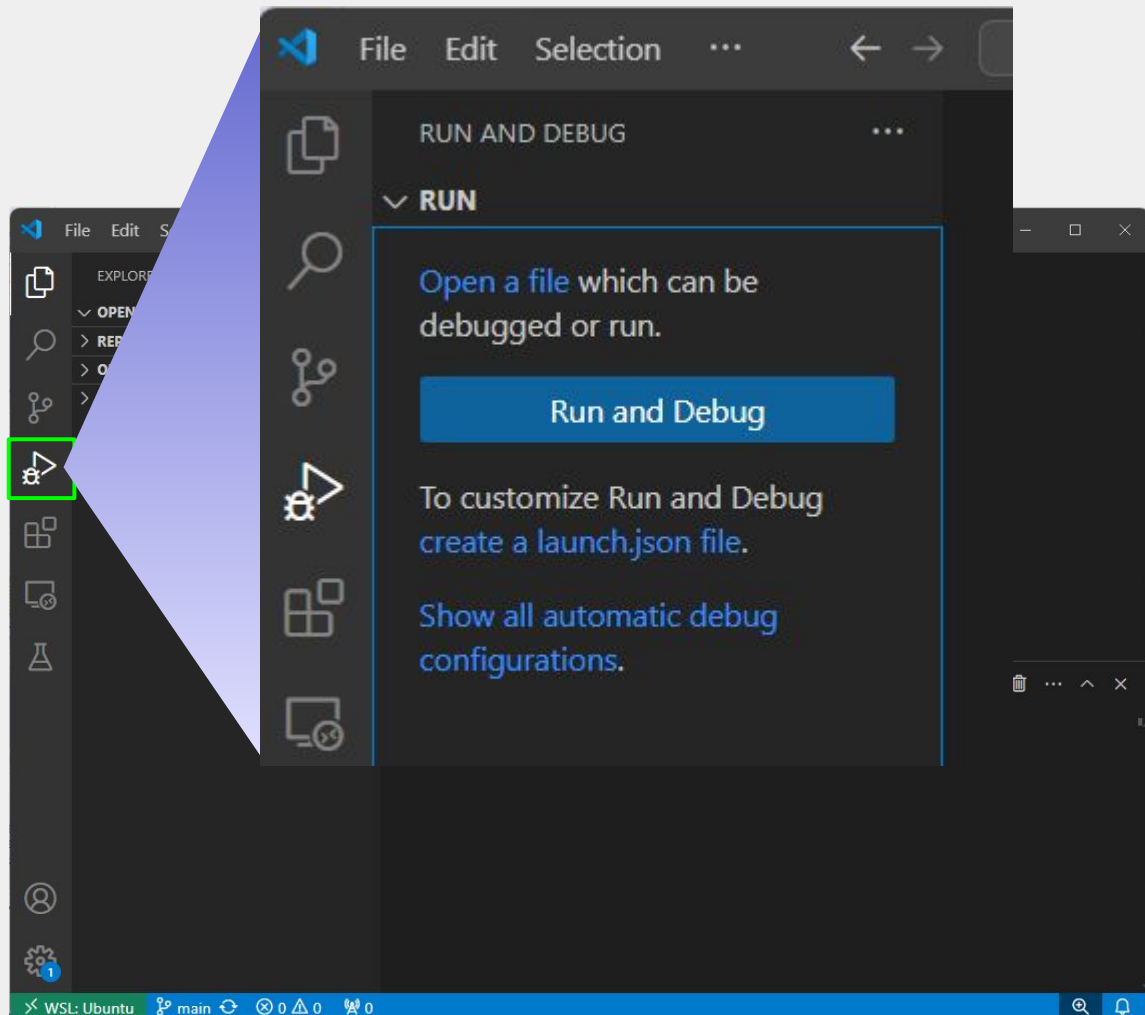
```
1  contador = 0
2
3  while contador < 3:
4      print("O contador é:", contador)
5      contador += 1
6
7  print("acabou!")
```

A red dot, representing a breakpoint, is located on the left margin of line 1. A green arrow points from the top left towards this red dot. The editor's interface includes a toolbar at the top right with icons for running, stepping through, and other debugging actions.

Esse pontinho vermelho é um
breakpoint ou ponto de parada

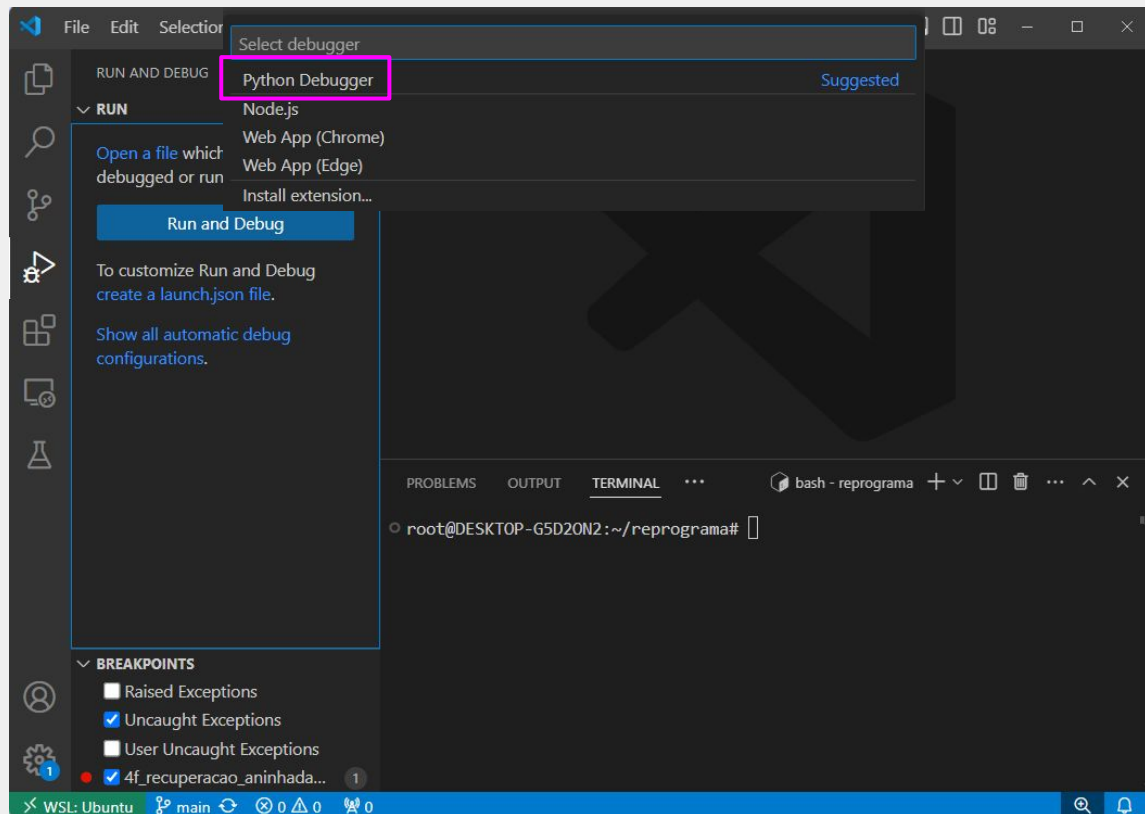
Olha o bugzinho ali do lado no VSCode

Clica nele pra aparecer a aba de **Run and Debug** (Executa e Debuga)



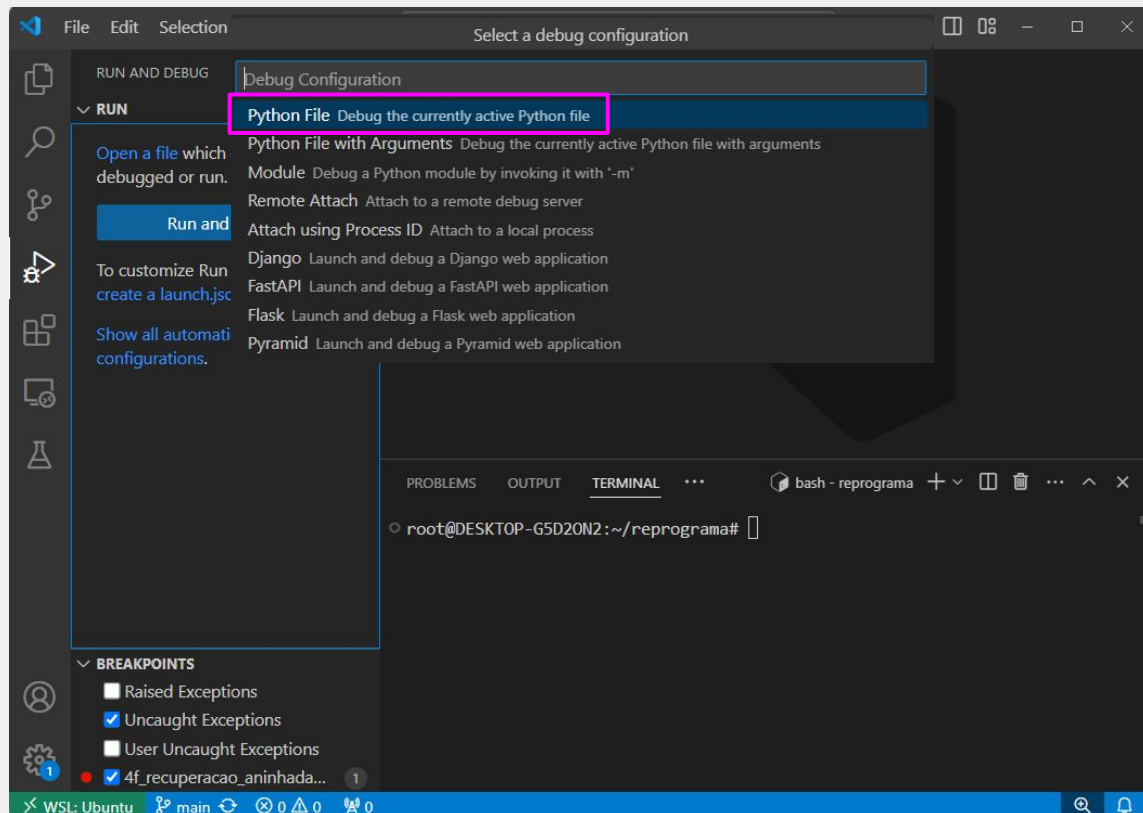
Clica no botão Run and Debug

Aparece uma
caixinha pra vc dizer
que quer usar o
Python Debugger



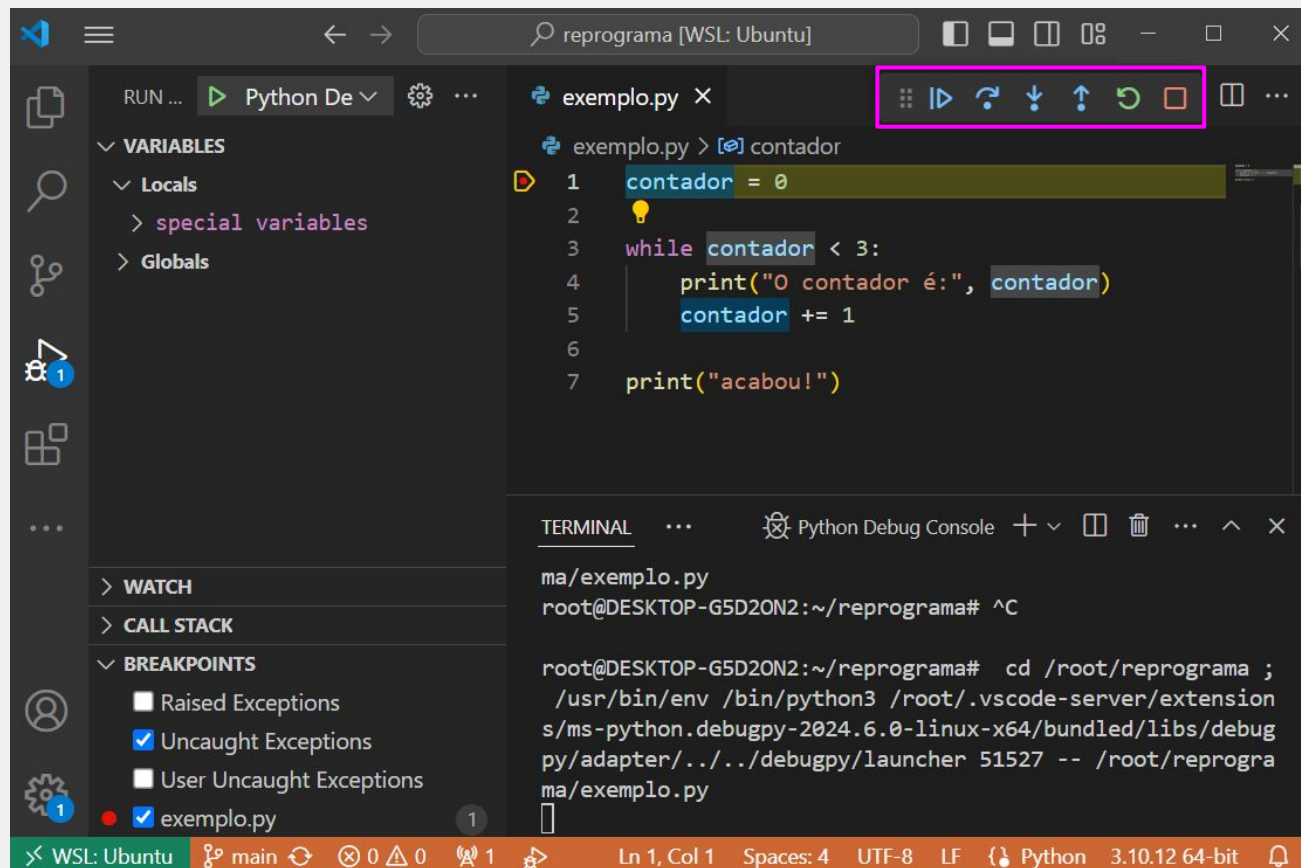
A janelinha muda e aparece outra lista

Clica no Python File
Debug the currently
active Python File
(Arquivo Python
Debuga o arquivo
Python ativo no
momento)



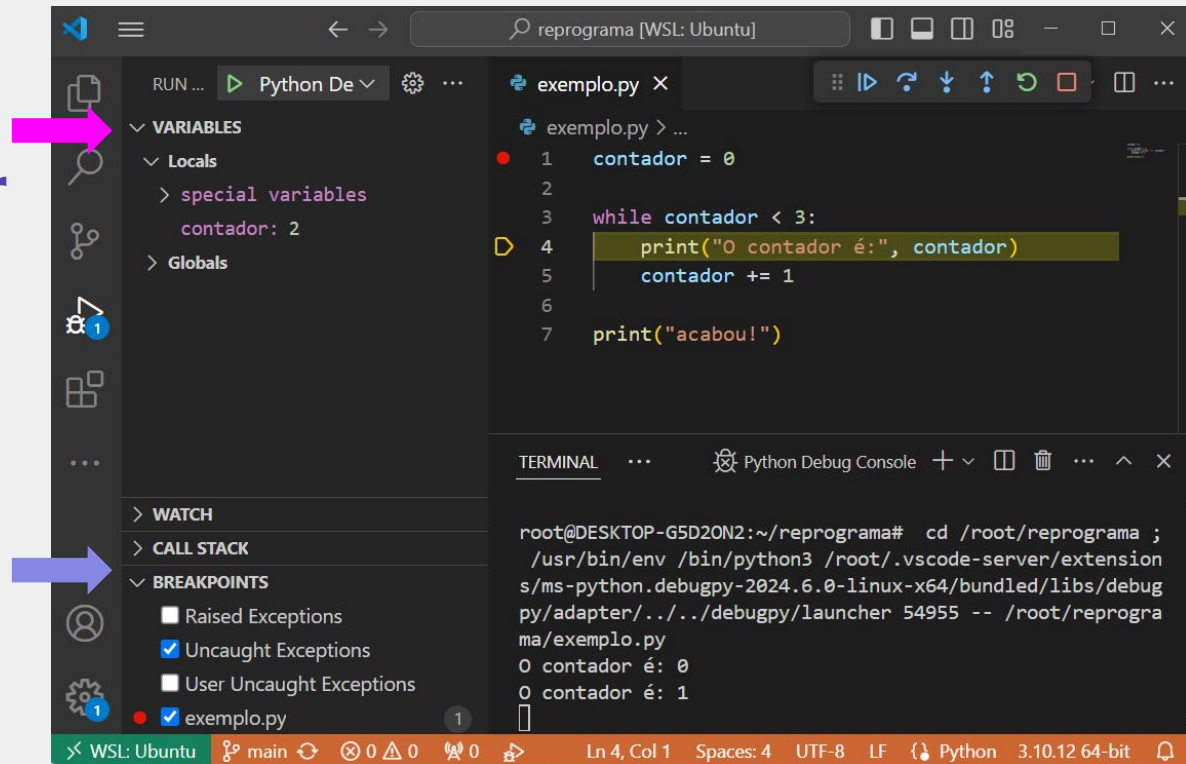
No painel

clique em
para seguir
o código
passo a passo



À esquerda, no alto, vão aparecer as **variáveis**

Embaixo, os **breakpoints** (pontos de parada)



Exceções

Às vezes precisamos desviar da
regra pra dar tudo certo

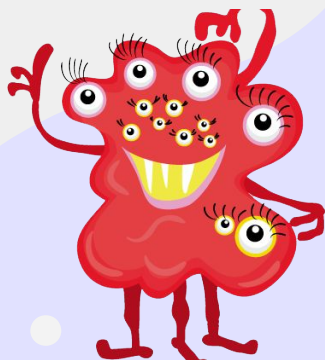


**Muitas vezes
somos capazes de
prever o que o
usuário vai errar**

Lembra dos nossos
erros agora há pouco?

Para evitar que o programa
“quebre” e pare totalmente,
podemos usar:

Try e except



Erro de execução

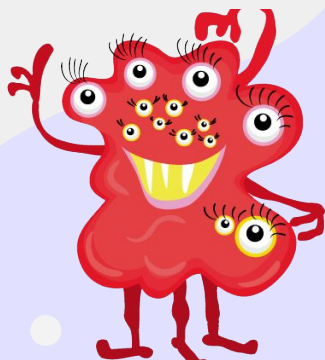
Tá tudo certo,
só que não...

```
7_mais_erros.py > ...  
1 print("Vamos rachar a conta")  
2  
3 valortotal = float(input("Qual o valor gasto? R$ "))  
4 amigos = int(input("Quantas pessoas? "))  
5  
6 print("Cada pessoa deve pagar R$", valortotal/amigos)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/7_mais_erros.py  
Vamos rachar a conta  
Qual o valor gasto? R$ 120  
Quantas pessoas? 0  
Traceback (most recent call last):  
  File "/root/reprograma/7_mais_erros.py", line 6, in <module>  
    print("Cada pessoa deve pagar R$", valortotal/amigos)  
ZeroDivisionError: float division by zero
```

Erro na Divisão por Zero: um tipo float está sendo dividido por zero



Erro de execução

Tá tudo certo,
só que não...

```
try_except.py X
try_except.py > ...
1  nota = float(input("Qual a nota? "))
2
3  if nota >= 7:
4      print("Aluna aprovada")
5  else:
6      print("Aluna reprovada")
7

PROBLEMS  OUTPUT  TERMINAL  ...
Python - reprograma + v [] [X] ... ^

root@DESKTOP-G5D20N2:~/reprograma# /bin/python3 /root/reprograma/try_except.py
Qual a nota? g
Traceback (most recent call last):
  File "/root/reprograma/try_except.py", line 1, in <module>
    nota = float(input("Qual a nota? "))
ValueError: could not convert string to float: 'g'
```

Erro de Valor: não é possível converter string para float: 'g'

Documentação

👉 Ela vai te dar pistas

<https://docs.python.org/3/library/exceptions.html>

Obrigada, mulheres maravilhosas!

Na segunda-feira vamos tirar
todas as dúvidas

