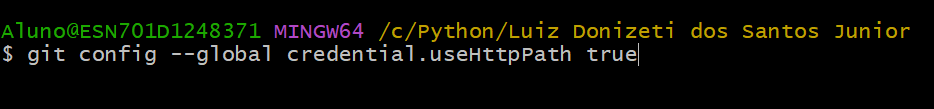
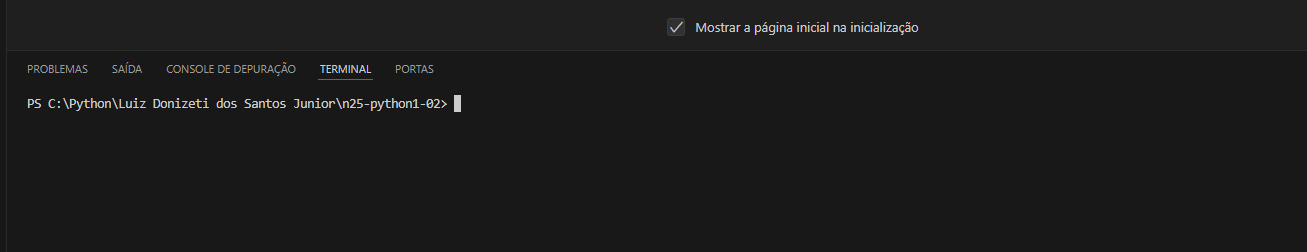


Setinha pra cima já copia tudo de novo



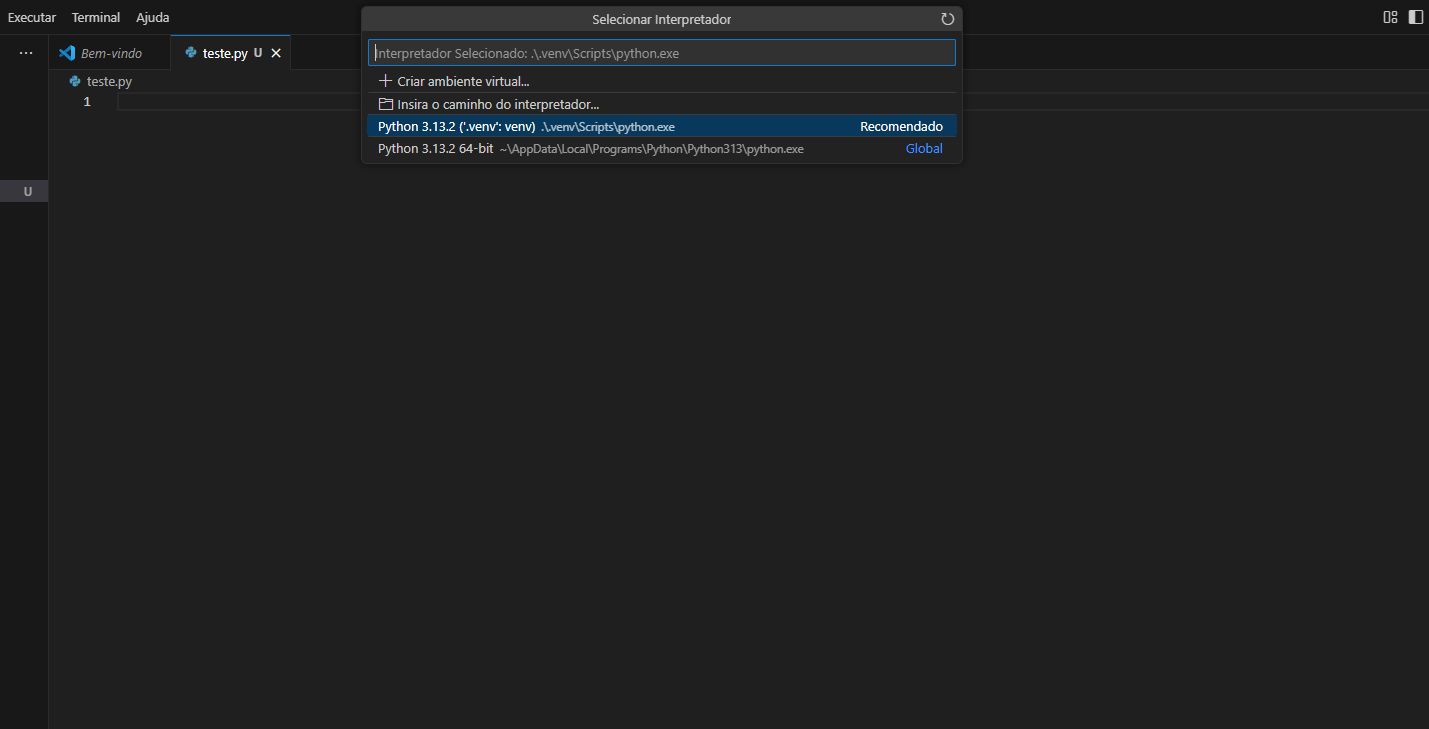
Para não haver duplicidades o git fez uma configuração para sempre fazer a autenticação de usuário.

VS Code CTRL + ‘

Precisa criar um ambiente virtual para o python dentro do VS CODE:  


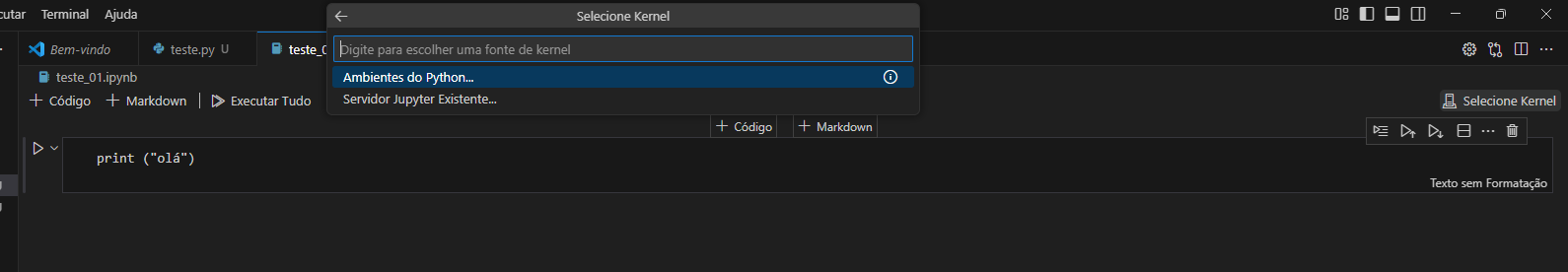


Python use o módulo “value in value”. Isso faz com que criemos uma pasta virtual para iniciar a programação.

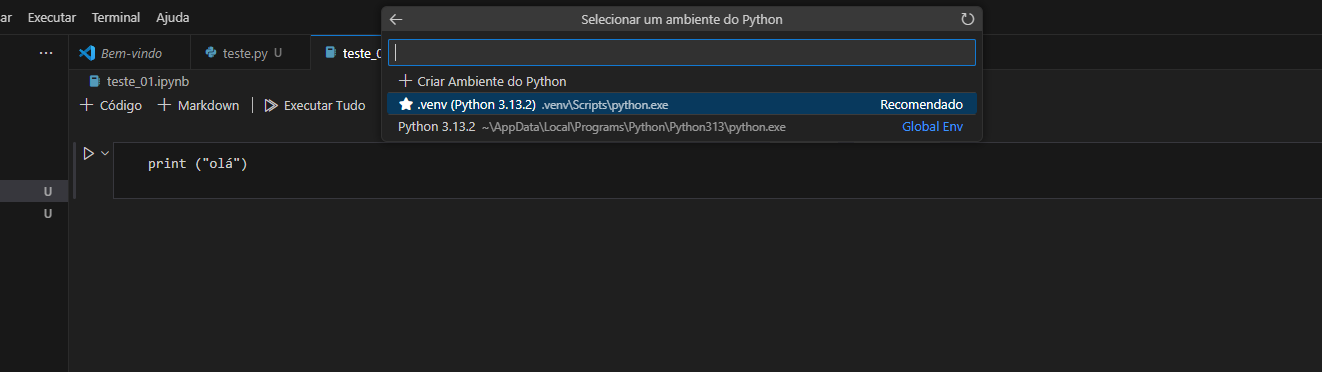


Clique na versão do python localizada abaixo do terminal do VS Code e coloque no venv para criar um novo ambiente virtual.

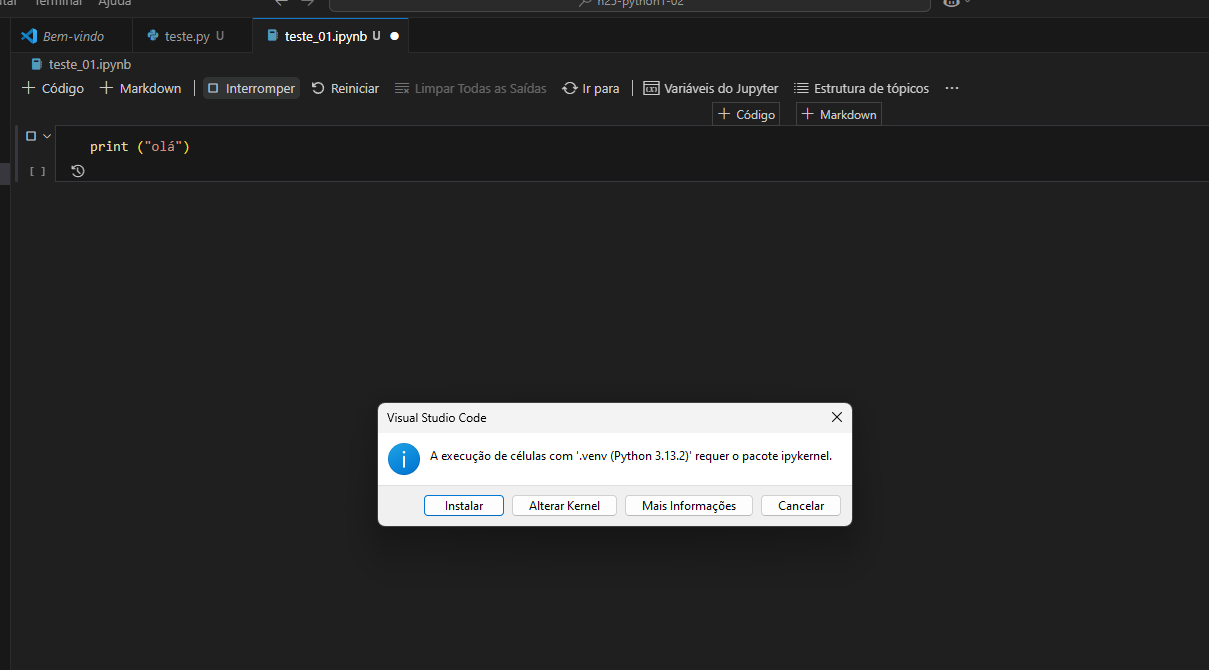
Exclua o terminal e aperte CTRL + ‘ para recarregar seu terminal.  
ipynb – interative python notebook



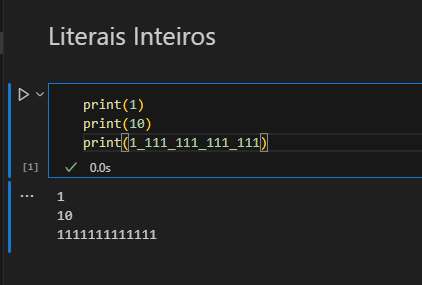
Selecione o Kernel



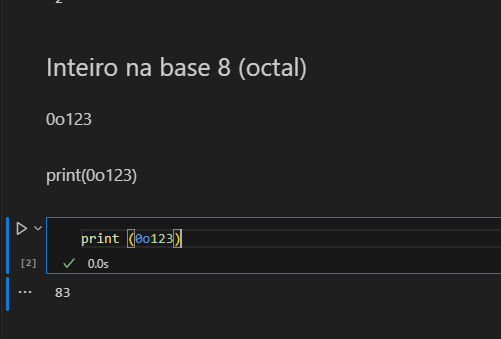
Escolhe o venv



Tudo isso para irmos realizando as instalações dos programas que permitem que façamos a programação. No caso ele está instalando o Python junto ao programa Jupiter para conseguirmos iniciar.



Use underlines para representar o número no código.

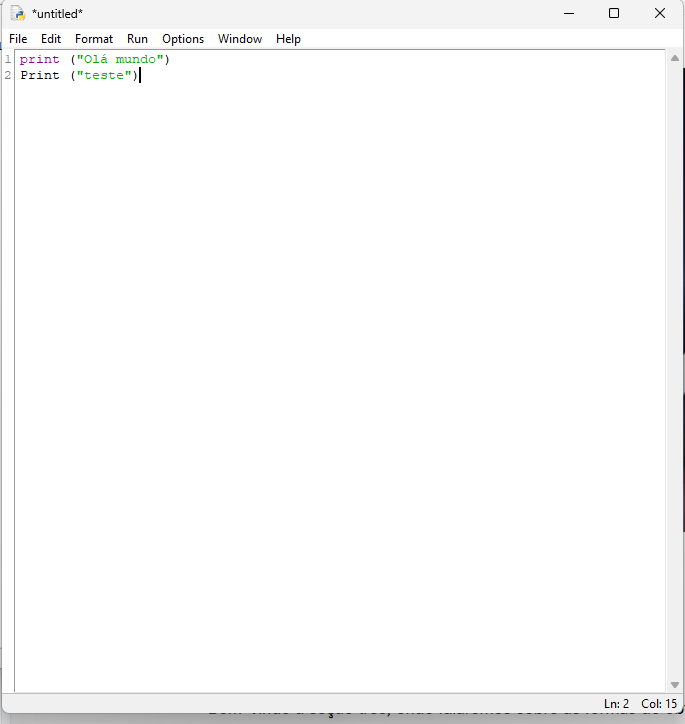


Número 123 representado na forma octal, base oito. Usado muito em redes de computadores. Um literal inteiro do python nunca vai começar sem o 0.

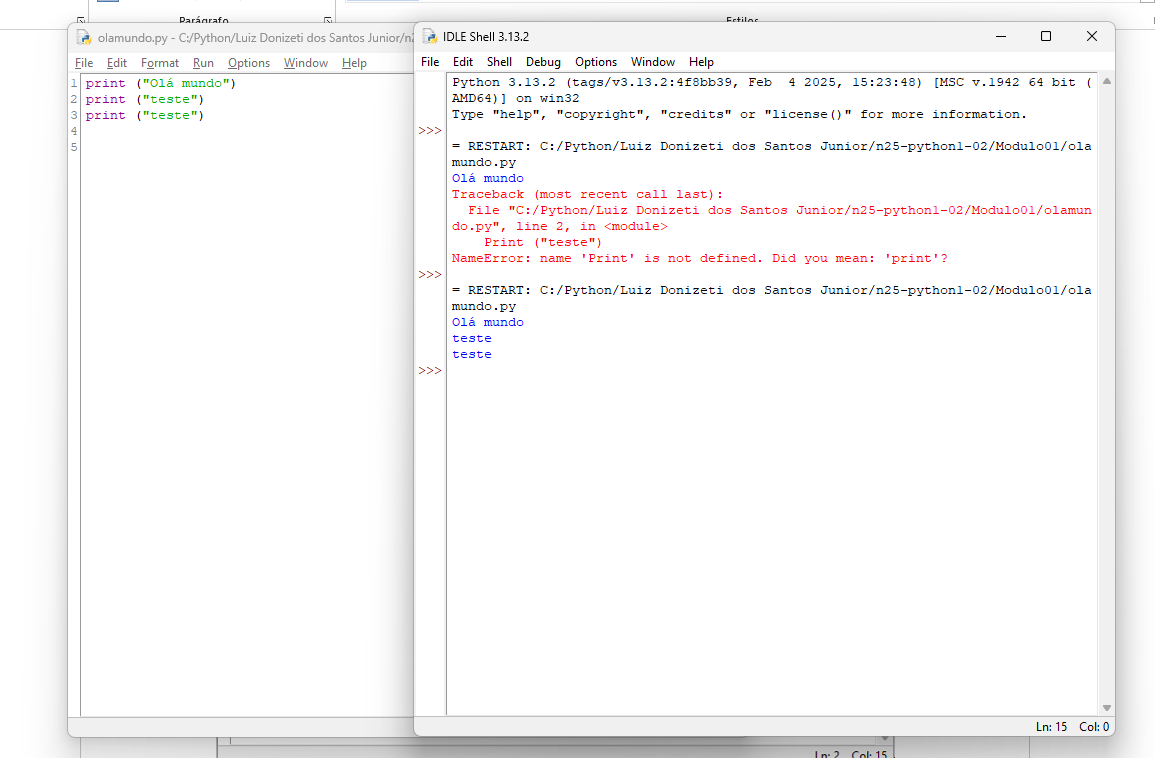
0o = Octal

0b = Binário

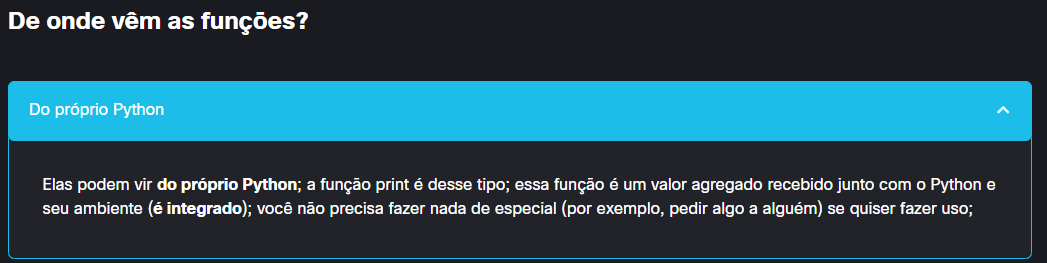
0x = Hexagonal

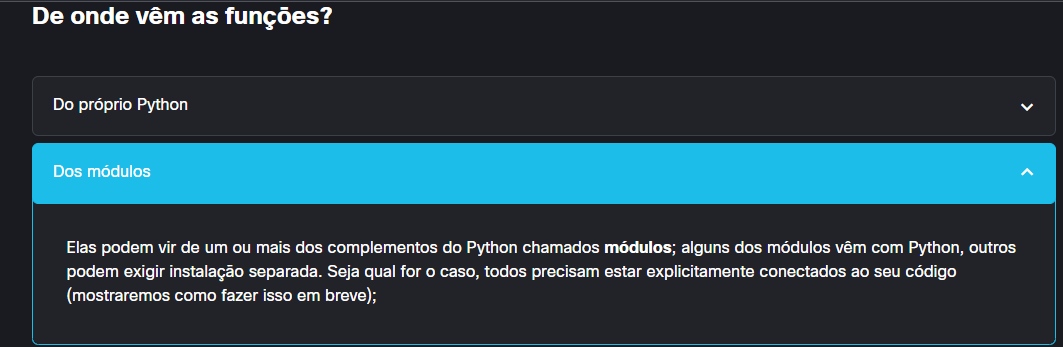


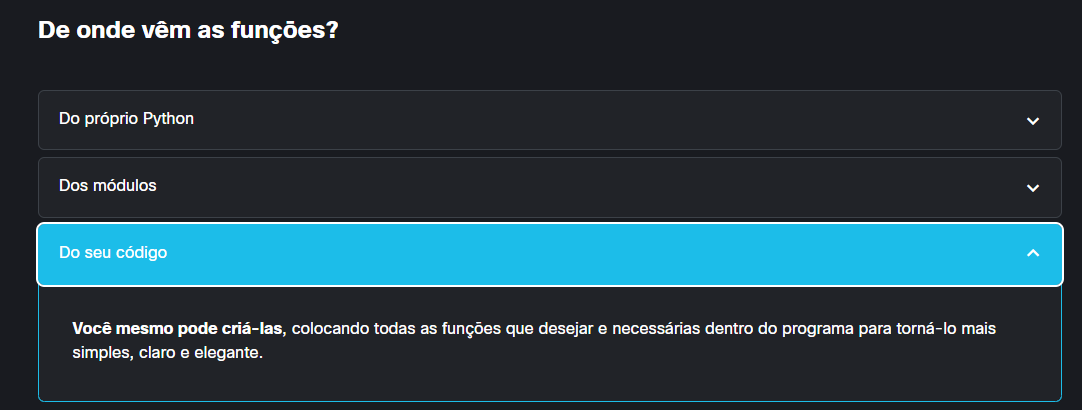
Repare a cor (no print com o P minúsculo funciona no P maiúsculo não)

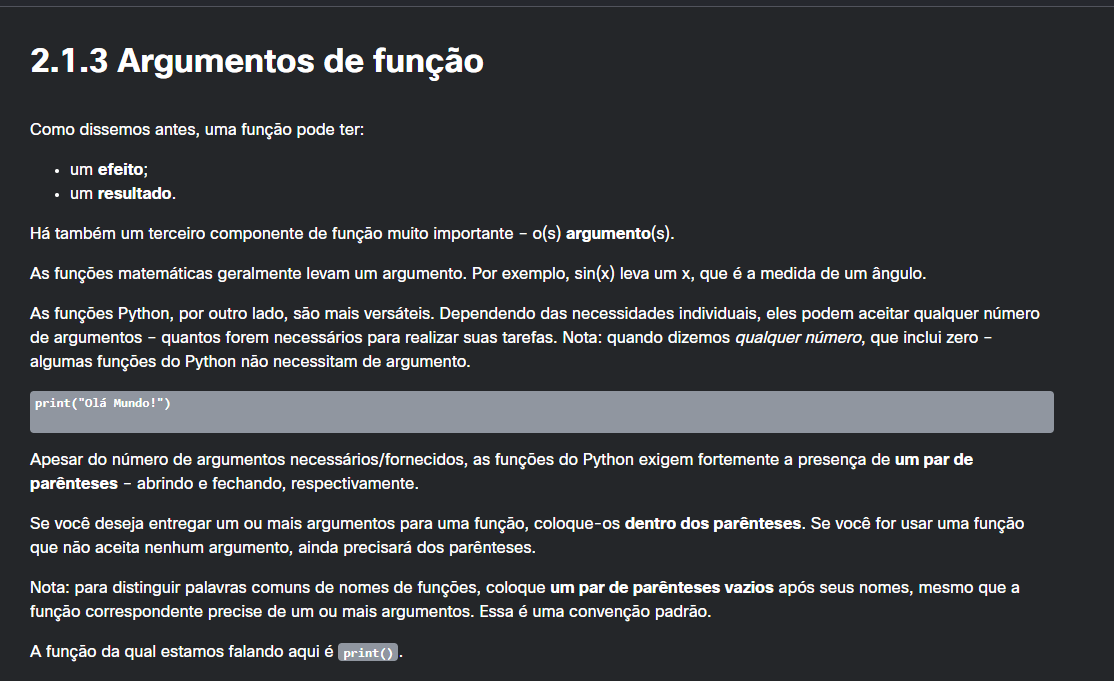


O python vai sempre ler linha a linha se você errar em uma linha ele vai depurar pra vc corrigir.



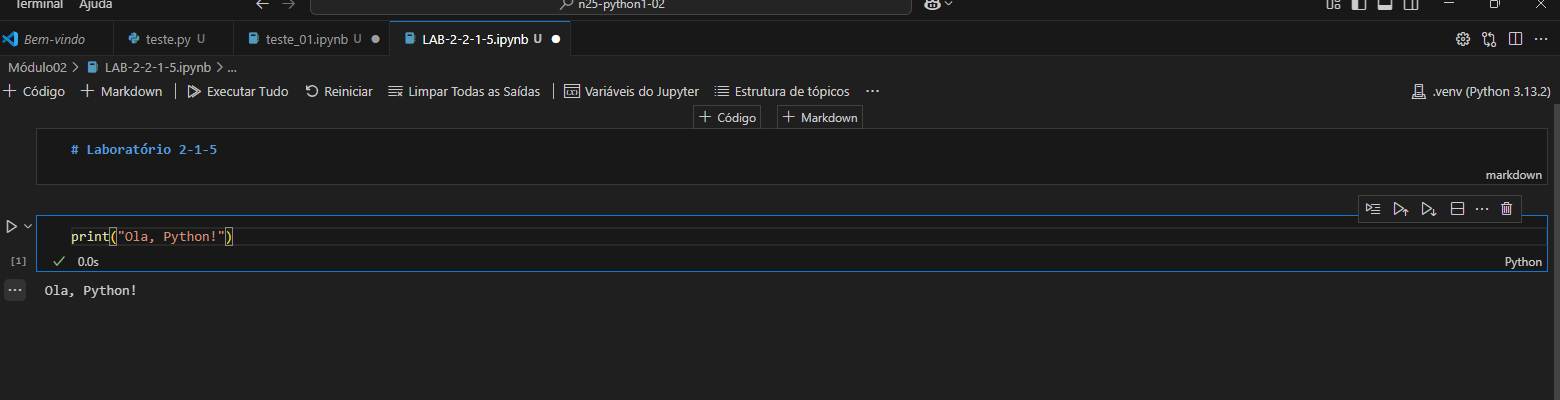


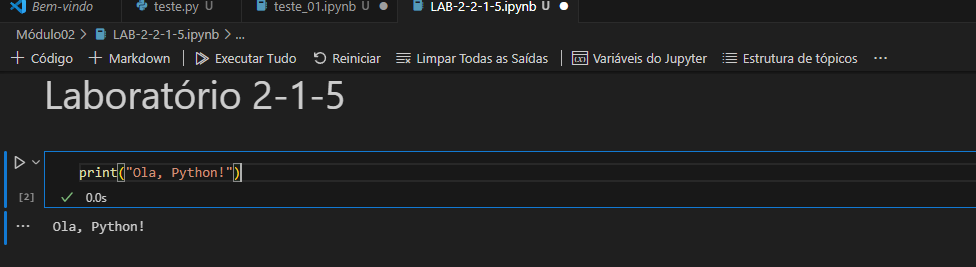






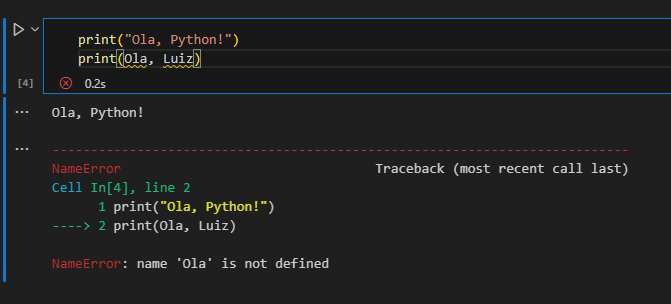


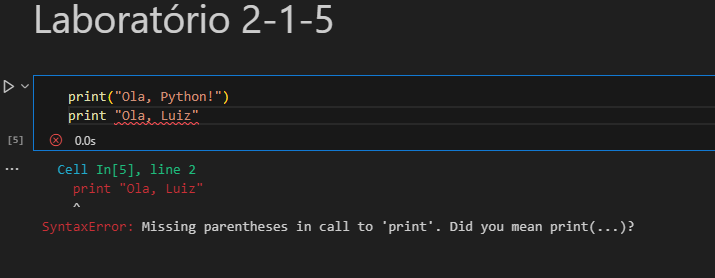




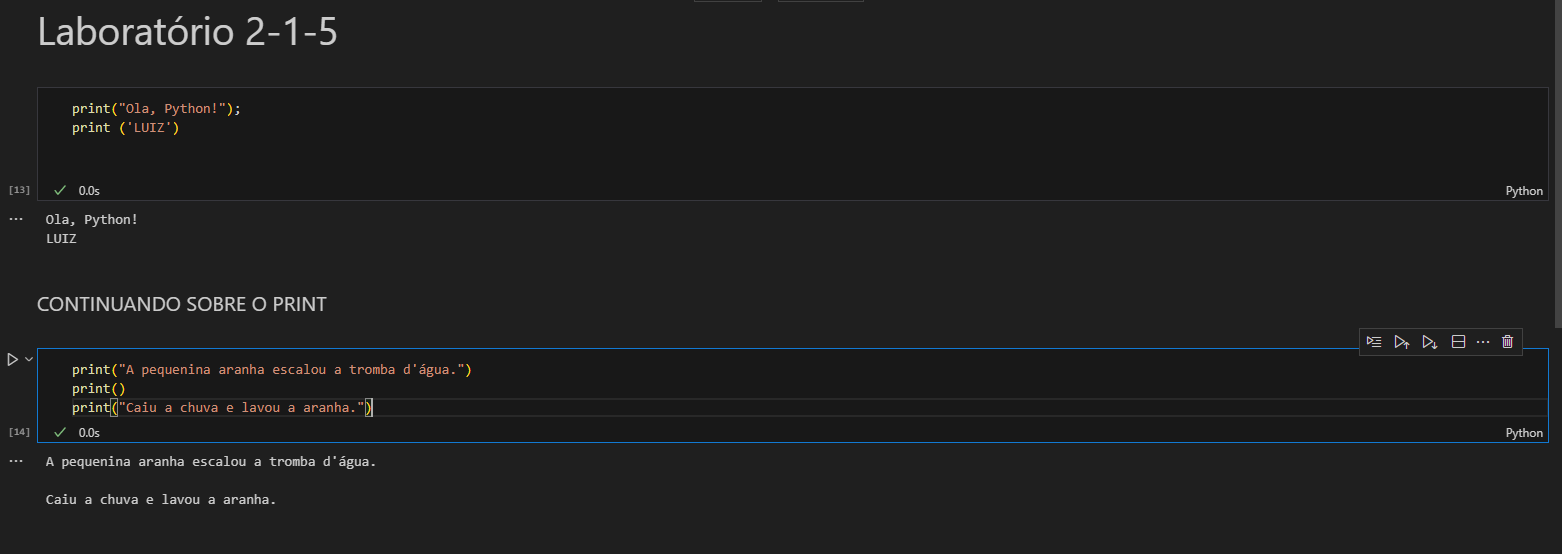
Os ### indicam o tamanho do título que você quer

Atalho para rodar CTRL + ALT + ENTER





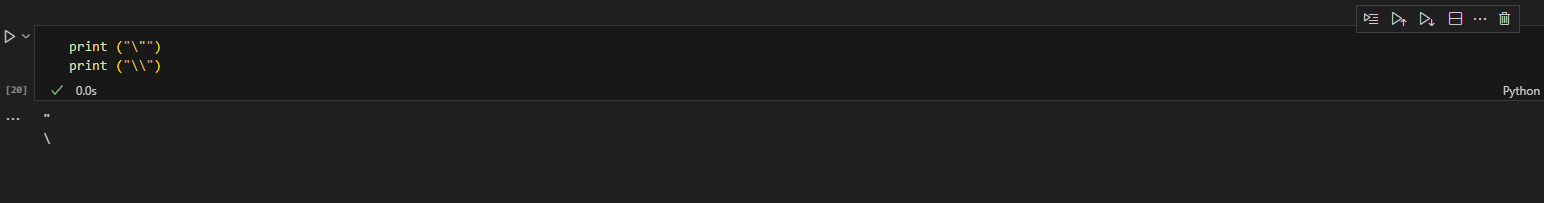
Usar o ; na linguagem VS code a virgula duplica o comando



Detalhe para o print () fechado para dar um salto de linha.

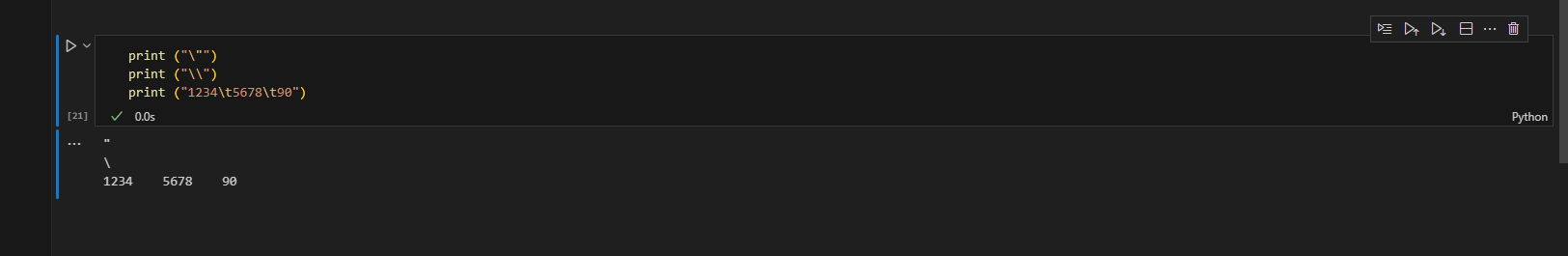


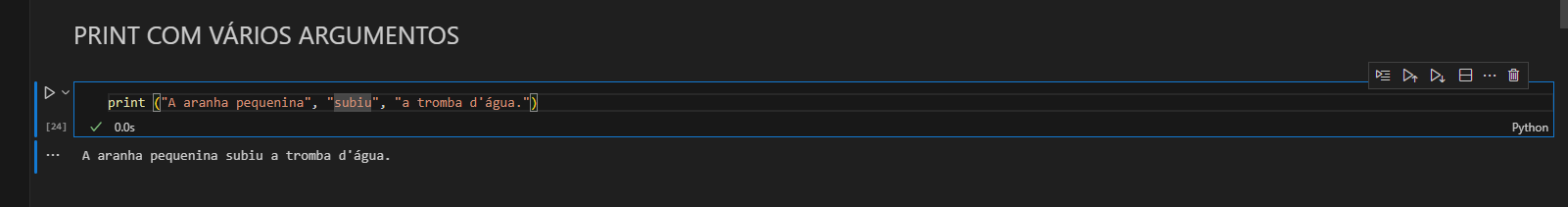
\n (new line) no código acima quebra uma linha antes do “subiu.”



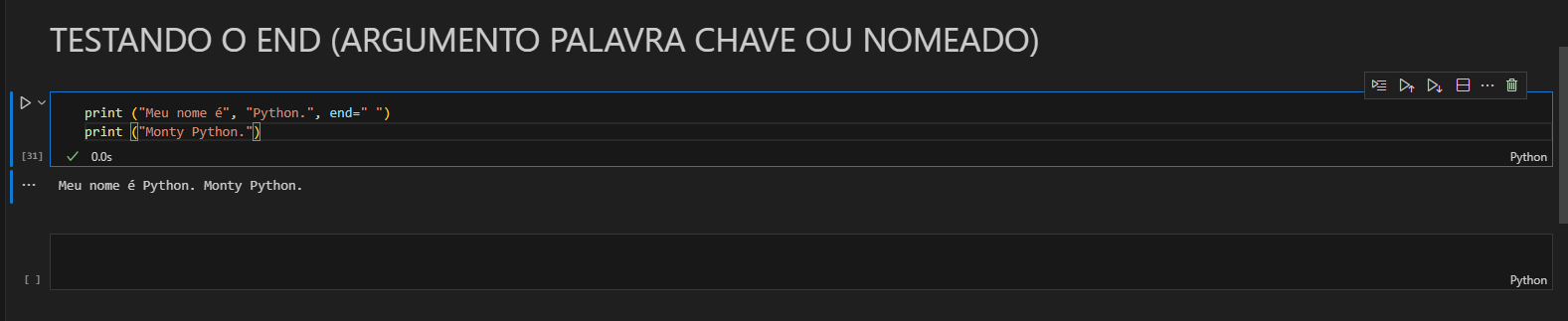
Com a barra invertida usar duas barras para o python entender o comando.

\t tabulação

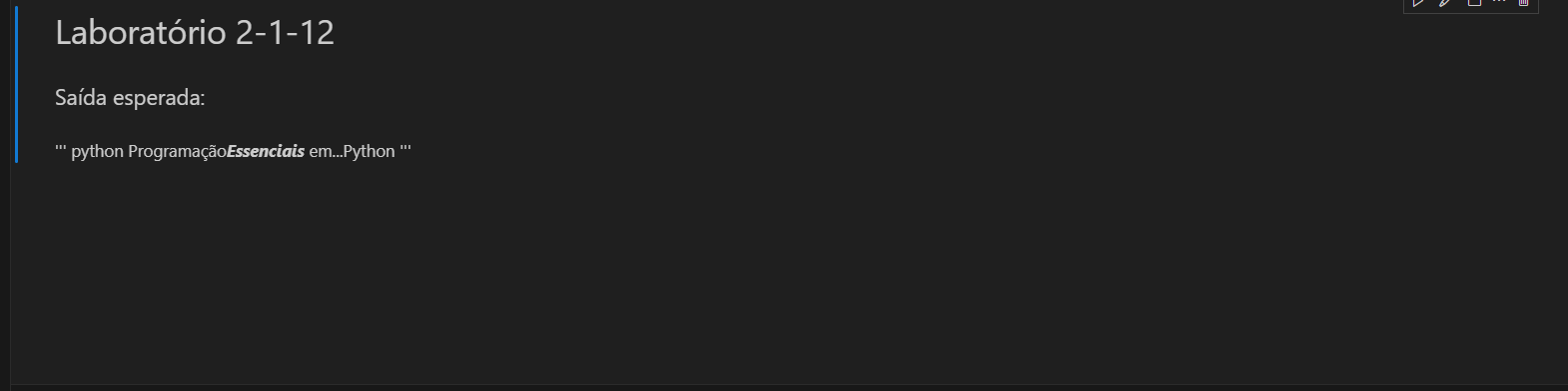




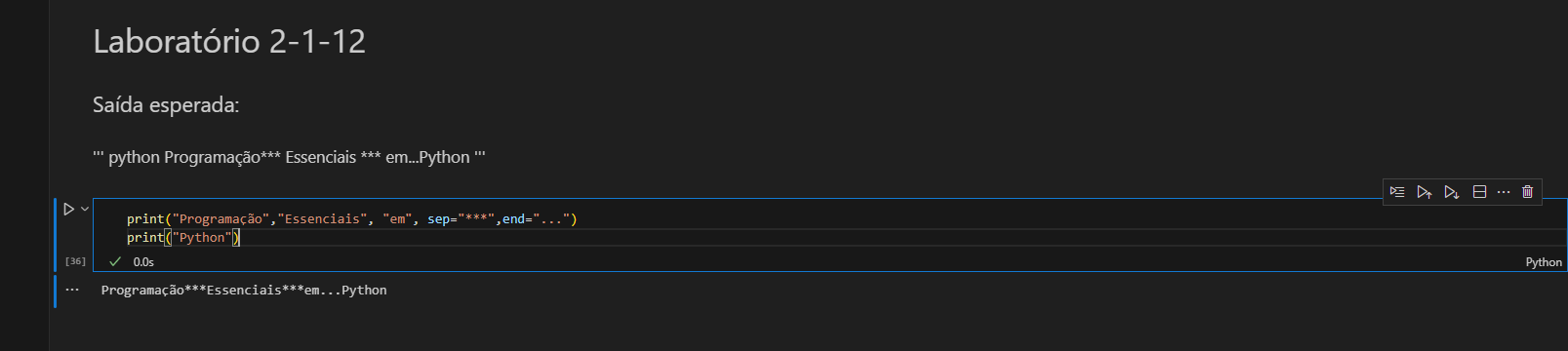
O print já separa os argumentos corretamente sem nenhuma função exigida, ou seja, os argumentos são posicionais, eles tem a posições definidas. Temos também o argumento nomeado que é o argumento de palavra chave (end)



Com o end você consegue mexer no padrão do Python (end=” “) ou seja, espaço.

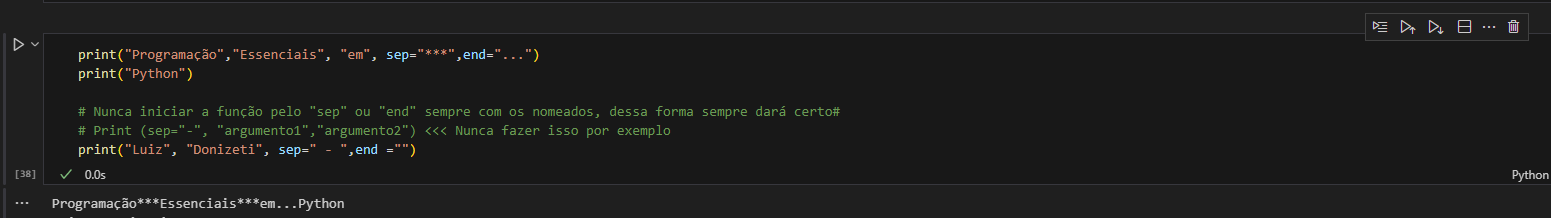


‘’’ (crase)

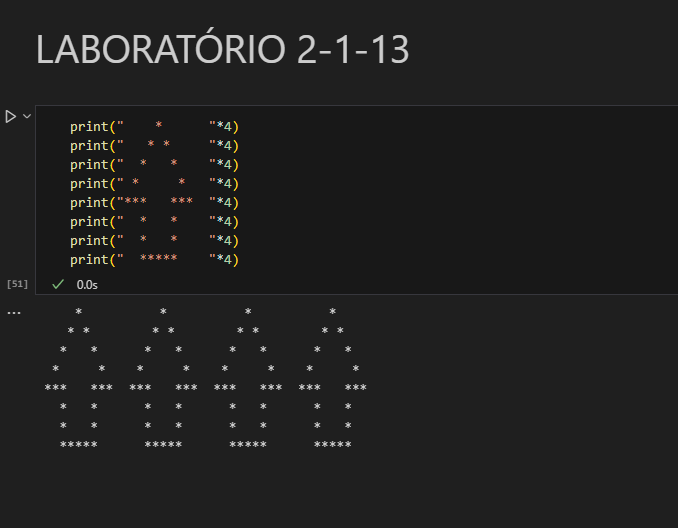


Veja acima, primeiro digita o texto, depois os separadores e end, ai significa que você escreveu com separadores (\*) e final (...).

Sempre posicional depois nomeado.







CTRL + D vai colocar vc em modo de múltipla edição. Para sair dele basta apertar ESC.

Para transformar um número em binário basta ir dividindo ele até o final. 0 e 1 e para multiplicar fazer o contrário potencializando.