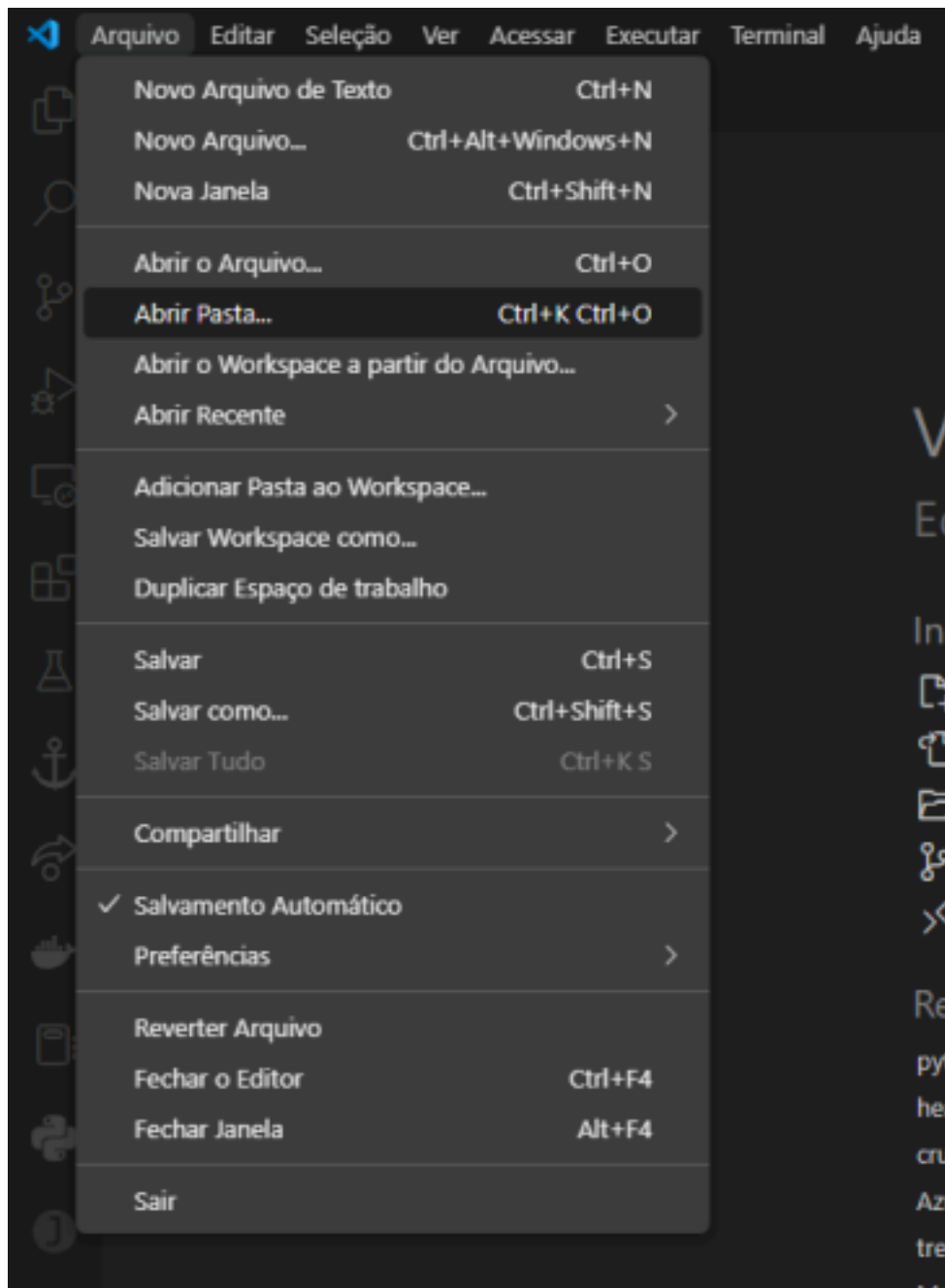


# Aula 04 - Rodando o primeiro Projeto Python

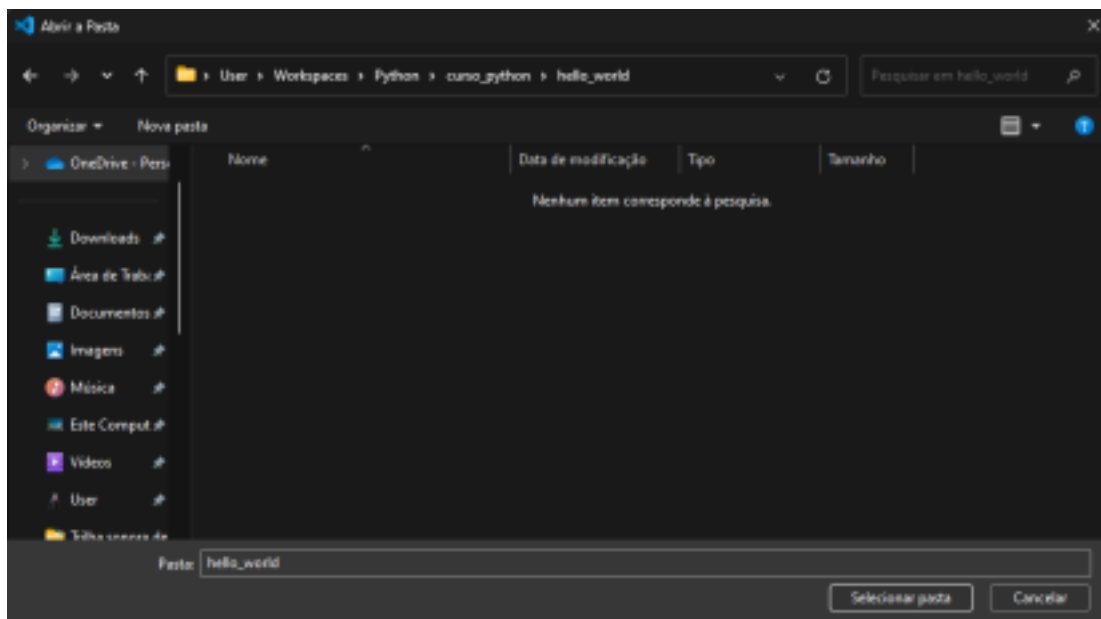
Chegou a hora!!!

1. Após personalizar o VSCode à gosto, abra-o, e vá no menu **Arquivo -> Abrir pasta**. Ou ainda usar a tecla de atalho **Ctrl+K Ctrl+O**.

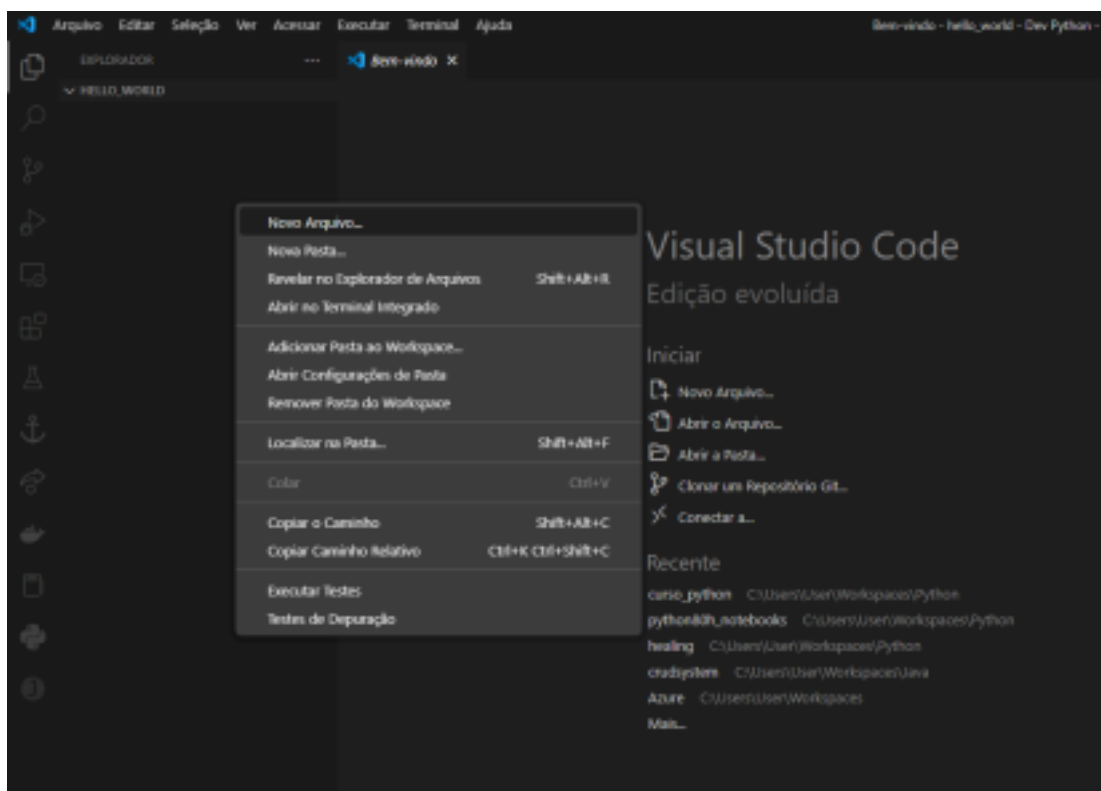


2. Crie uma pasta para guardar todos os seus projetos em Python. Exemplo: **Curso\_Python**. Escolha um lugar fácil para guardar essa pasta, como a Área de Trabalho, por exemplo. Abra a pasta, e crie outra pasta com o nome do seu projeto,

por exemplo: **hello\_world**. Após criar essa pasta, abra e clique no botão **Selecionar pasta**.



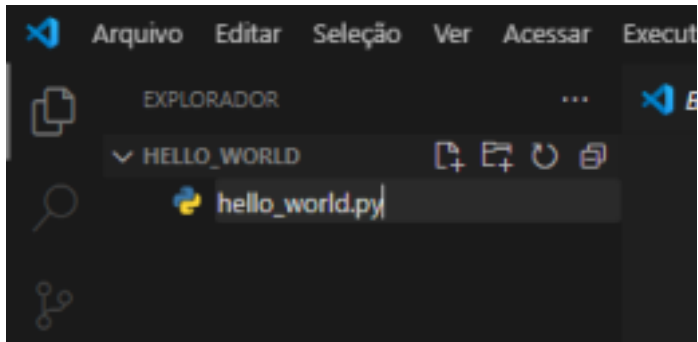
3. Essa pasta será definida como a pasta do seu projeto. À esquerda, irá abrir um painel, que funcionará como um explorador de pastas e arquivos dentro do seu VSCode. Clique com o botão direito dentro da área vazia desse painel e escolha a opção **Novo Arquivo**.



4. Isso fará abrir uma caixa de texto dentro do explorador do VSCode. Sua tarefa aí é digitar um nome de um arquivo, que nesse caso deve ser um arquivo de código fonte Python. Os arquivos Python são aqueles que terminam com a extensão **.py**.

Portanto, salve o arquivo com **.py** no final, como por exemplo: **hello\_world.py**,

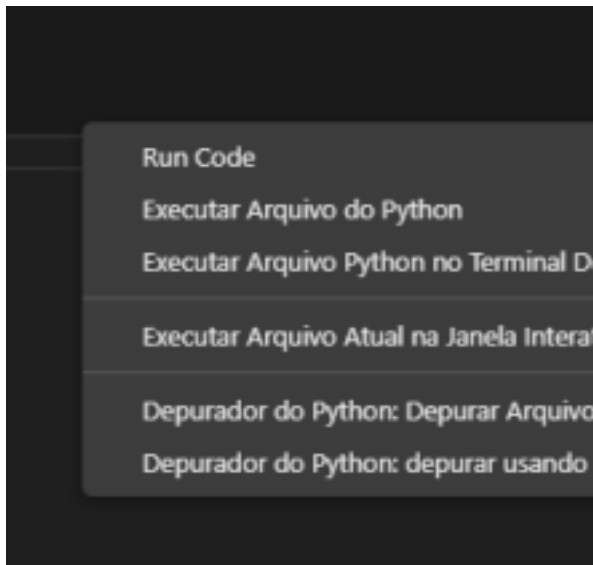
e finalize com **Enter**.



5. Ele irá abrir uma guia no VSCode com o arquivo recém-criado. Vamos usar ele para testar a nossa configuração Python. E a melhor forma de fazer isso é rodando o programa conhecido como **Hello World**, que tem como objetivo testar as configurações da linguagem, e também experimentar a linguagem, para ver como funciona. Para isso, precisamos executar o comando de **saída de dados** da linguagem, que irá exibir na tela a famosa frase **Alô, mundo!**. No arquivo que está aberto, escreva o código abaixo:

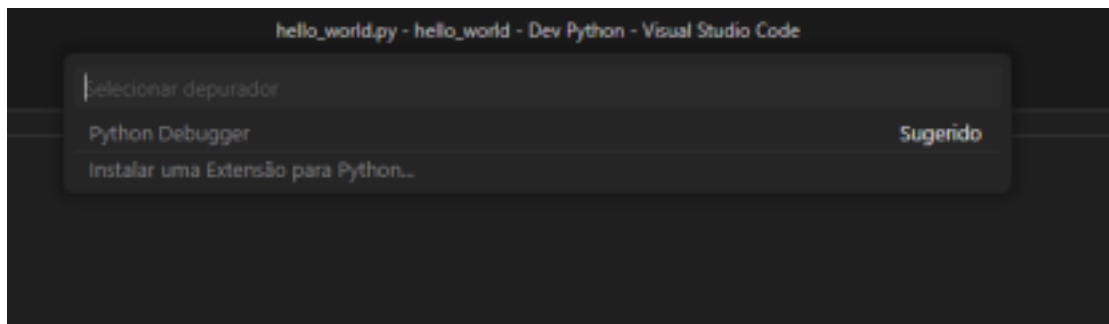
```
In [ ]:  
print('Alô, mundo')
```

6. Para executar o programa, podemos clicar no botão de **Play** no canto superior direito da tela, conforme imagem abaixo:

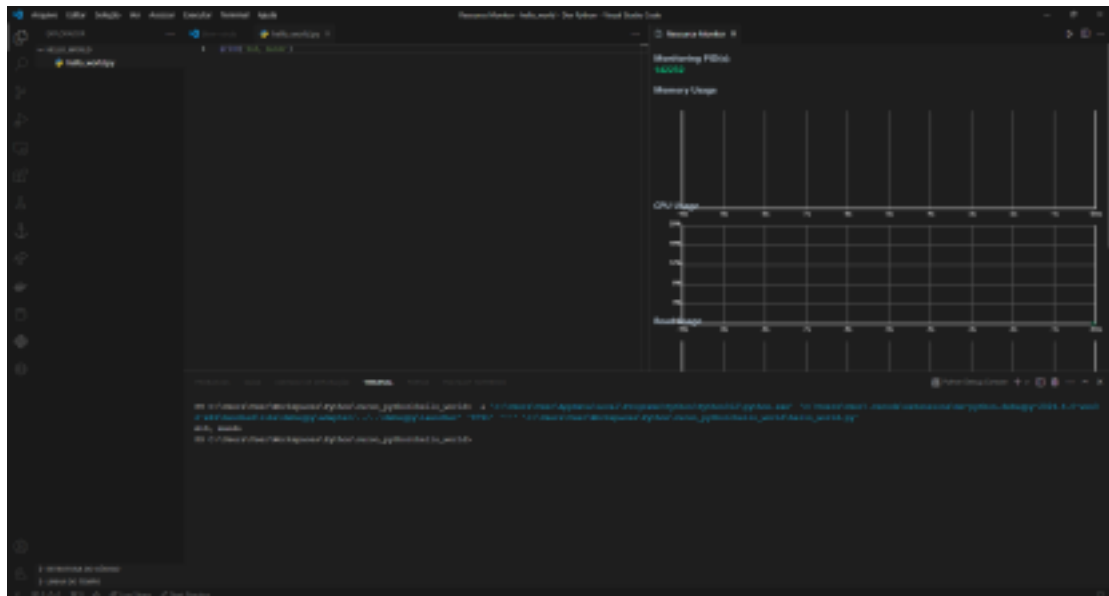


7. Você pode tanto clicar no símbolo do play quanto clicar na setinha e escolher a opção **Executar Arquivo do Python**, e ainda tem também a opção **Executar Arquivo Python no Terminal Dedicado**, que costuma executar o programa mais rápido. Há ainda uma outra opção: executar a tecla de atalho **Ctrl+F5**. Essa última opção tem uma vantagem: caso tenha instalado a extensão **Python Resource**

**Monitor**, ele irá mostrar o consumo de CPU e de RAM da sua aplicação. Basta escolher a opção **Python Debugger** quando executar.



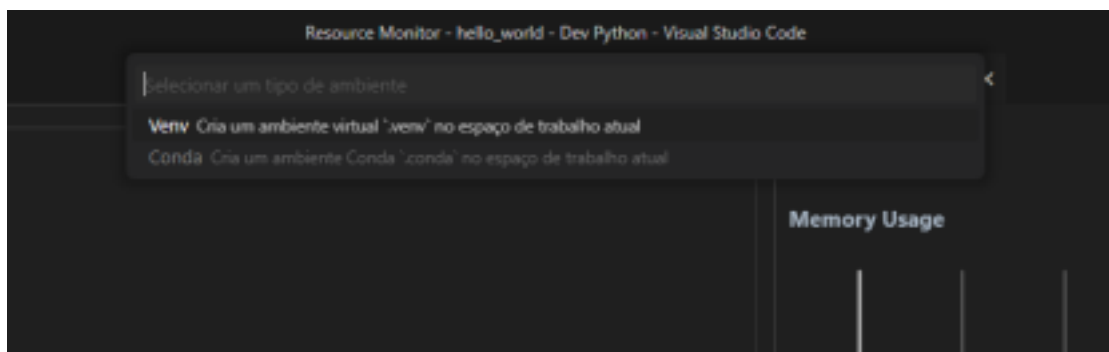
8. Veja o programa sendo executado no console abaixo, enquanto na tela ao lado é exibido o consumo do algoritmo.



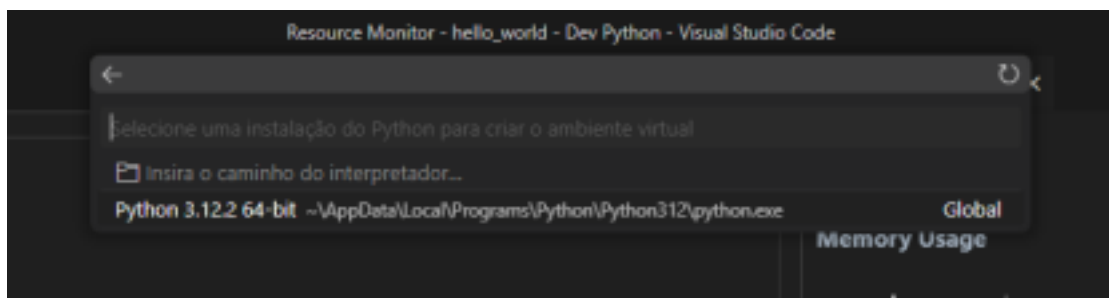
9. O programa irá rodar normalmente, já que ele conseguiu identificar o Python instalado na sua máquina através da extensão Python instalada na aula anterior. Entretanto, isso não será suficiente para alguns programas que iremos criar em aulas futuras. Isso porque em alguns deles serão necessárias a instalação de pacotes externos, e para que isso seja possível, é obrigatório a configuração do **Ambiente virtual Python**. Portanto, a partir de agora, faremos essa configuração para cada novo projeto Python.
10. Na tela do VSCode, aperte a tecla **F1**. A caixa de diálogo que irá abrir na parte de cima da tela do VSCode serve para a execução de alguns comandos de configuração do próprio VSCode. Digite **Python**, e procure pela opção **Criar Ambiente**, ou **Create Environment** no inglês, conforme imagem abaixo, e escolha essa opção:



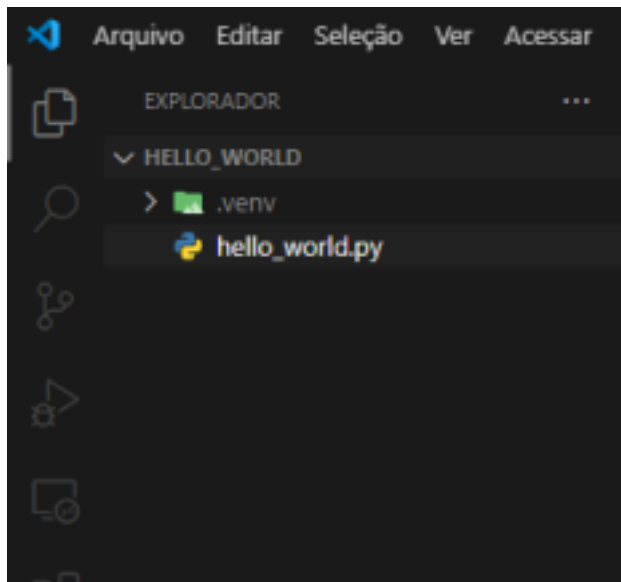
11. Ele irá solicitar para que escolha como deseja criar o ambiente Python: através do **Venv** ou do **Conda** (essa última opção só irá aparecer caso tenha instalado o Anaconda na sua máquina). Iremos escolher o **Venv**, mas sinta-se livre para escolher o que desejar.



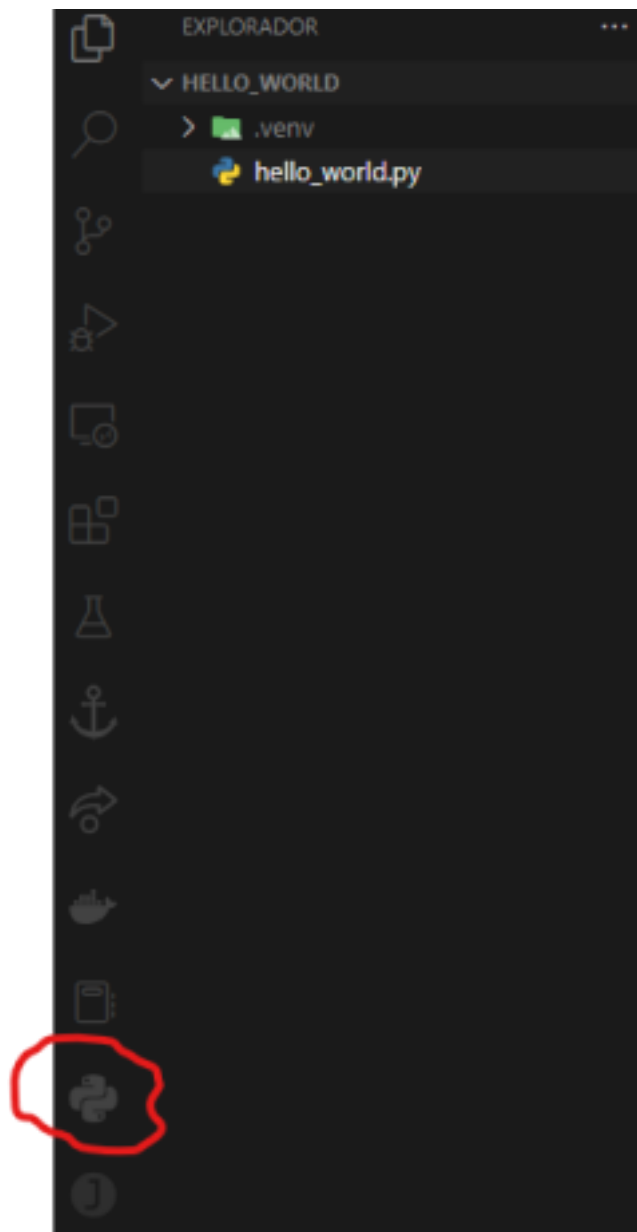
12. Você será solicitado a escolher a versão do Python que desejar. Caso tenha mais de uma versão instalada na máquina, você poderá escolher à vontade. Caso não tenha, ele irá mostrar apenas a versão instalada, que você deverá selecionar.



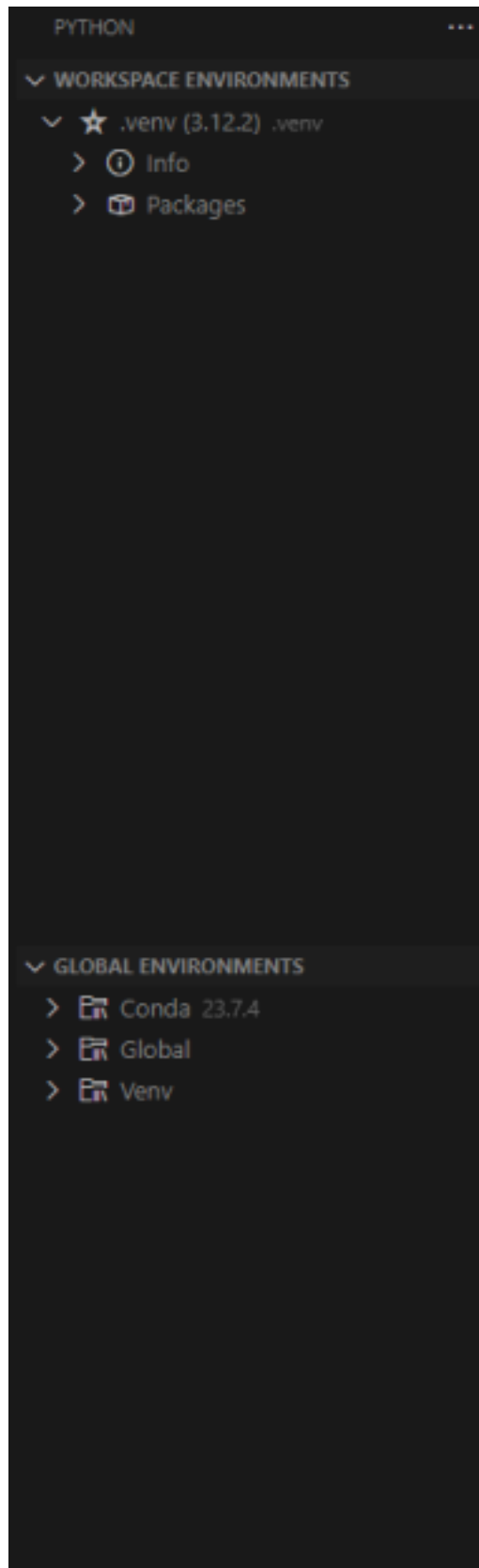
13. Se der certo, irá ser criada uma pasta chamada **.venv** na raiz do seu projeto. Este é o ambiente virtual Python configurado no seu projeto.



14. Caso tenha instalado a extensão **Python Environment Manager**, há ainda outra forma de criar o mesmo ambiente virtual. Clique no botão com o símbolo do Python na barra de ferramentas à esquerda.

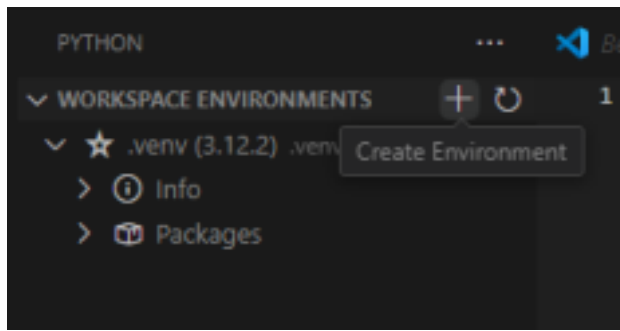


15. Irá abrir um novo Painel.



16. Nesse novo painel é possível criar um novo ambiente virtual, clicando no sinal de + sinalizado na imagem abaixo.





17. Há ainda uma outra forma, via CLI. Você pode abrir um terminal dentro do VSCode, clicando no menu **Terminal -> Novo Terminal**, ou clicando com o botão direito e escolhendo a opção **Abrir no Terminal Integrado**, ou usando a tecla de atalho **Ctrl+Shift+`**.
18. Depois, é só digitar o comando `python -m venv /path/to/new/virtual/environment`.
19. Uma outra opção com CLI: você pode instalar um pacote para criação de ambientes virtuais usando o gerenciador de pacotes pip, através do comando `pip install virtualenv`.
20. Em seguida, é só usar o comando `virtualenv .venv` para criar o ambiente virtual.
21. Em qualquer um dos casos usando CLI, será necessário a ativação do ambiente virtual, também via CLI. Para isso, é só inserir o comando `.venv/Scripts/Activate`.

**Meus parabéns!!! Você executou o seu primeiro código em Python!!!  
Na próxima aula iremos aprender mais sobre a sintaxe da linguagem.**