Como configurar o VS Code com Anaconda e Jupyter Notebooks no Windows 10

[](https://medium.com/@joaolggross?source=post_page-----b05258bf65c1----------------------)

[João Gross](https://medium.com/@joaolggross?source=post_page-----b05258bf65c1----------------------)

[Aug 21](https://medium.com/@joaolggross/como-configurar-o-vs-code-com-anaconda-e-jupyter-notebooks-b05258bf65c1?source=post_page-----b05258bf65c1----------------------) · 5 min read

Deixe sua IDE pronta para projetos de Data Science

Image for post



Fontes: Anaconda, Jupyter e VS Code

O **Jupyter Notebook** é uma aplicação de código aberto que permite a criação e compartilhamento de projetos com código de programação e texto. Em um Notebook é possível inserir trechos de códigos de diferentes linguagens de programação, tais como Python, R, Scala, e outras dezenas de linguagens, executar os códigos e exibir os resultados. Além disso, há a opção de colocar anotações em texto, criar tabelas e inserir imagens, o que permite documentar o projeto à medida que o código é inserido. A combinação de código executável com a documentação em texto faz dos Notebooks umas excelente ferramenta de experimentação e prototipação, perfeita para projetos de Data Science.

No âmbito de Data Science, é muito utilizada a linguagem de programação Python. Para trabalhar com Python nos Notebooks, bem como com as bibliotecas mais popularmente utilizadas, como Pandas, Matplotlib, Seaborn e Numpy, uma opção bastante interessante é a distribuição **Anaconda**. Ela provê um ambiente de software completo com todas as bibliotecas necessárias e versões atualizadas do Python e do R para o desenvolvimento de projetos científicos.

E para desenvolver o projeto, um ambiente de desenvolvimento geralmente facilita a vida do programador, pois possui ferramentas de depuração, refatoração, ferramentas para autocompletar funções e consultar documentação, dentre outras. O **Microsoft Visual Studio Code**, ou VS Code, é uma alternativa interessante. Nele é possível configurar o ambiente com a distribuição de Python do Anaconda e criar projetos com Notebooks para experimentação.

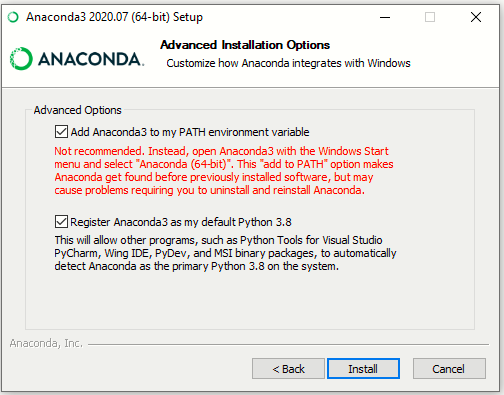
Neste artigo você vai aprender a:

1. Instalar a distribuição Anaconda.
2. Instalar a IDE VS Code e configurar o ambiente de desenvolvimento para Python.
3. Criar um projeto no VS Code com um Notebook e executar comandos simples.
4. Instalar pacotes com o comando conda no terminal do VS Code.

**Instalando a distribuição Anaconda**

Acesse a [página oficial](https://www.anaconda.com/products/individual) da distribuição Anaconda e faça o download do executável. Inicie a instalação e siga os passos. Quando chegar nas opções avançadas de instalação, certifique-se de marcar a opção “Add Anaconda3 to my PATH environment variable”.

Image for post



Marcando opção “Add Anaconda3 to my PATH environment variable”.

Marcar essa opção é importante para executar comandos via terminal dentro do VS Code.

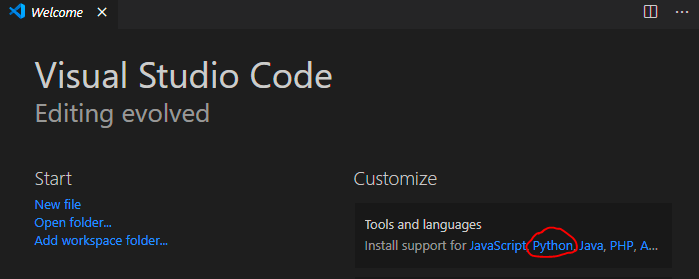
**Instalando o VS Code**

O VS Code possui uma versão gratuita e o download pode ser realizado na [página oficial](https://code.visualstudio.com/). A instalação é simples e direta, basta iniciar o executável e seguir os passos.

**Configurando o VS Code para Python**

Ao entrar no VS Code, na página inicial, você vai encontrar uma opção chamada “Tools and languages”. Escolha o Python e realize a instalação da extensão.

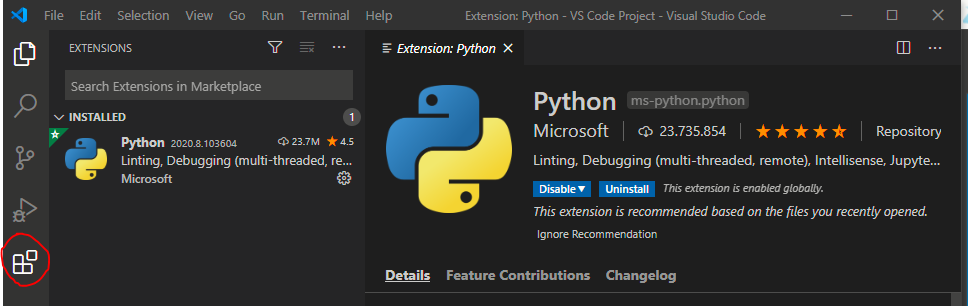
Image for post



VS Code. Opção “Tools and languages”.

Assim que o download for concluído, basta fechar e reabrir o VS Code para que a extensão já esteja ativa. Ao clicar no ícone do marketplace você pode ver que a extensão foi instalada com sucesso.

Image for post

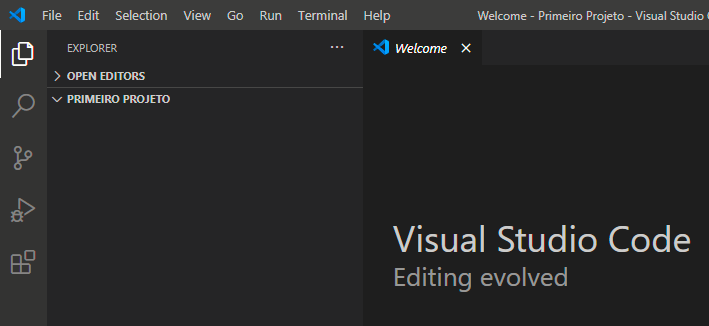


VS Code. Acessando o marketplace.

**Criando um projeto no VS Code**

Escolha um local para armazenar seu projeto e crie uma pasta para ele, por exemplo Área de Trabalho/Primeiro Projeto. Em seguida reabra o VS Code e no campo Start clique em “Open folder” e selecione a pasta. A pasta do projeto é carregada na coluna lateral esquerda da IDE.

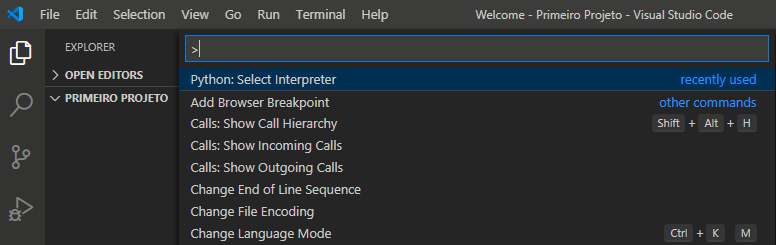
Image for post



VS Code. Pasta do projeto carregada.

Em seguida clique **Ctrl+Shift+P**e selecione o interpretador Python para o projeto.

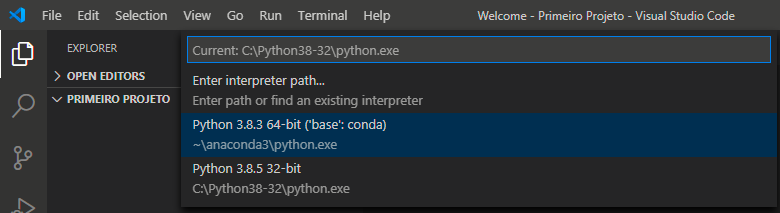
Image for post



VS Code. Escolha do Interpretador Python.

Também é preciso escolher a versão do Interpretador Python. No nosso caso escolhemos a versão fornecida pela distribuição Anaconda, que é a versão **Python 3.8.3 64-bit (‘base’: conda)**.

Image for post

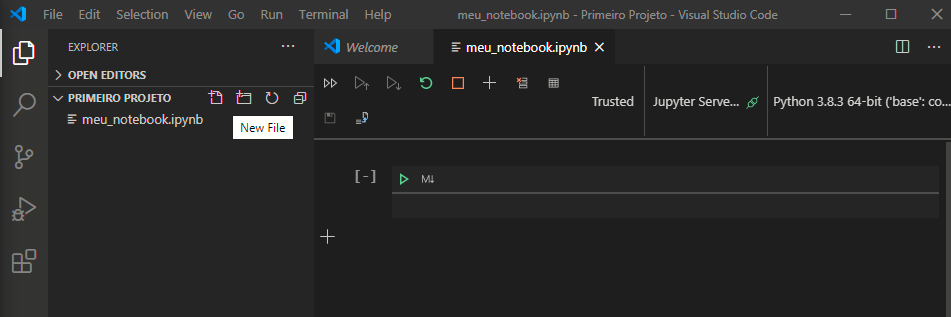


VS Code. Escolhendo a versão do Interpretador Python.

**Criando seu primeiro Notebook**

Na aba do projeto, clique no botão “New File” e crie um arquivo Notebook com a extensão .ipynb.

Image for post

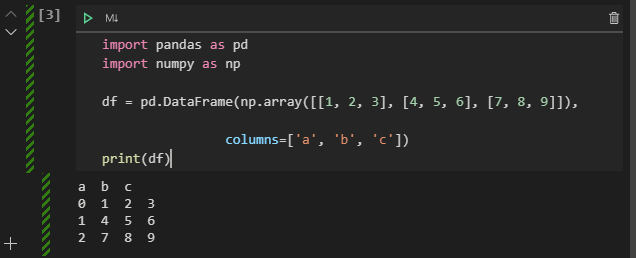


VS Code. Criando o primeiro Notebook.

Para testar o Notebook, vamos inserir na primeira caixa de execução o trecho de código abaixo:

A execução da célula do Notebook é feita com o atalho **Ctrl + Enter**.O resultado obtido é o seguinte:

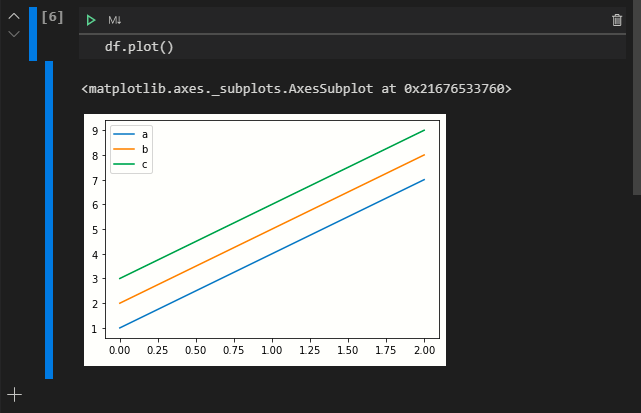
Image for post



VS Code. Execução de célula do Notebook.

Testando outro trecho de código:

Image for post

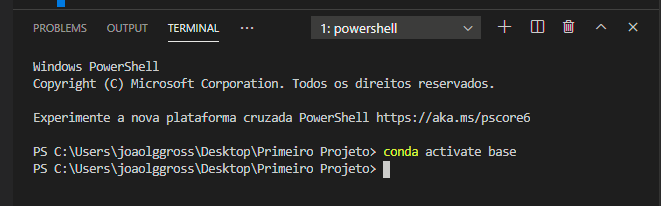


VS Code. Execução de célula do Notebook.

**Instalando novos pacotes com conda**

Dentro do VS Code você tem acesso a um terminal pelo atalho **Ctrl+Shift+`**.Assim que um terminal é aberto a distribuição Anaconda já é ativa automaticamente:

Image for post

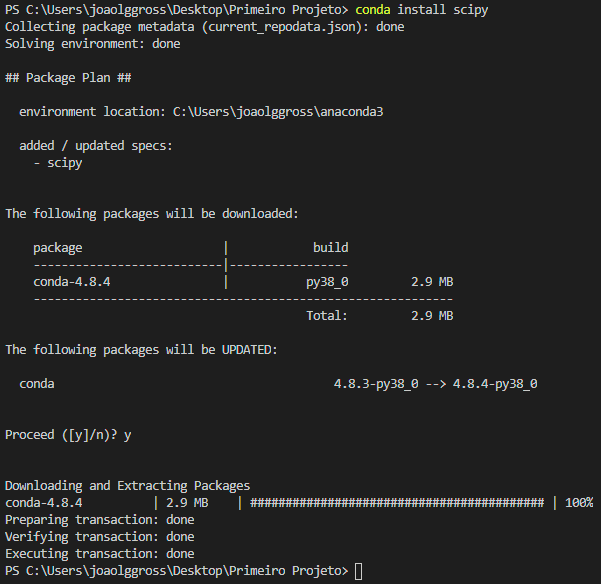


VS Code. Terminal com conda ativo.

O comando conda é uma interface para o gerenciamento de instalações na distribuição Anaconda, e permite buscar e instalar novos pacotes do repositório Anaconda, criar ambientes conda isolados e instalar e atualizar pacotes dentro desses ambientes.

Por exemplo, para instalar o ecosistema [SciPy](https://www.scipy.org/), que possui uma coleção de softwares open-source para computação científica em Python, basta executar o comando conda install scipy.

Image for post



VS Code. Instalando scipy com conda.

**Considerações Finais**

Trabalhar com Notebooks permite realizar experimentação de modo bastante ágil e prático. A distribuição Anaconda fornece um conjunto de pacotes de software para o desenvolvimento desses projetos em Python. E tudo isso integrado em uma IDE robusta como o VS Code facilita o desenvolvimento, depuração e organização do projeto. Criar um projeto em Python e inserir Notebooks para experimentação nunca foi tão fácil!

Para mais artigos detalhados como este, consulte meu repositório no GitHub, no qual mantenho um registro resumido de cada um:

**[jlggross/articles](https://github.com/jlggross/articles" \t "_blank)**

[This repository has a collection of all articles I've written for different websites such as LinkedIn and Medium (so…](https://github.com/jlggross/articles" \t "_blank)

[github.com](https://github.com/jlggross/articles" \t "_blank)