

Aula 03 - Configurando o ambiente de desenvolvimento Python

Conhecendo os programas a serem instalados

Antes de começarmos a programar em Python, precisaremos configurar o nosso ambiente, o que na prática significa instalar alguns programas. São eles:

- Python 3.x
- Visual Studio Code (VSCode)
- Git

É importante também que crie uma conta na plataforma do GitHub: <https://github.com>

Além desses, caso deseje se aprofundar mais nos estudos, os programas abaixo são recomendados, mas não serão objetos de estudo para este curso, e portanto, fica a seu critério se deseja instalá-los ou não:

- GitHub Desktop
- Anaconda
- PyCharm

Verificando a versão do Python na máquina

Antes de tudo, precisaremos verificar se o Python está instalado na máquina.

Caso a sua máquina possua um **Linux** como Sistema Operacional, provavelmente o Python já está instalado, mas poderá ser necessário a atualização dele. No caso do **Windows**, pode ser necessário a instalação, mas antes iremos verificar se sua máquina já possui ou não o Python, e caso possua, qual versão está sendo utilizada.

1. Abra o **Terminal** do seu SO. Se for Windows, é o **Prompt de Comando**. Isso pode ser feito no Windows usando a tecla de atalho **Ctrl+R** e digitando `cmd` na caixa de diálogo.
2. No terminal ou prompt de comando, digite `python --version` (dois hífen) ou `python -V` (um hífen e "V" maiúsculo). Ambos os comandos funcionam.

```
Microsoft Windows [versão 10.0.22631.3527]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\User>python --version
Python 3.12.2

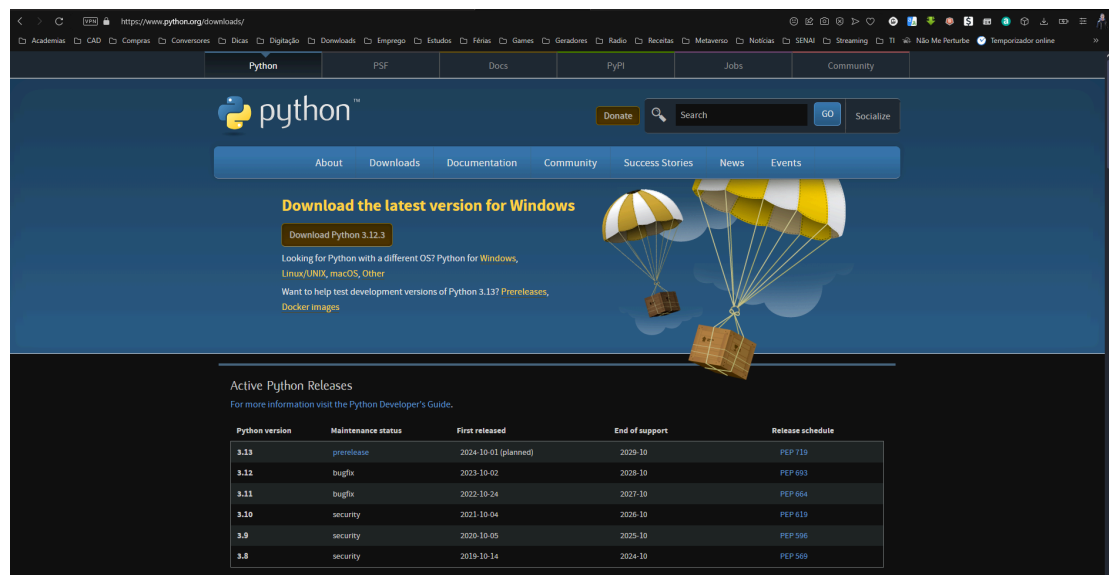
C:\Users\User>
```

3. Se der certo, irá aparecer a versão do Python reconhecida pelo seu SO. No nosso caso da imagem, a versão utilizada é a 3.12.2 (não é a versão mais recente, mas podemos trabalhar com ela). Pule o próximo passo e vá para a instalação do Git. Caso o comando não seja reconhecido, é porque o Python não está instalado. E aí será necessário a sua instalação. Siga para o próximo passo.

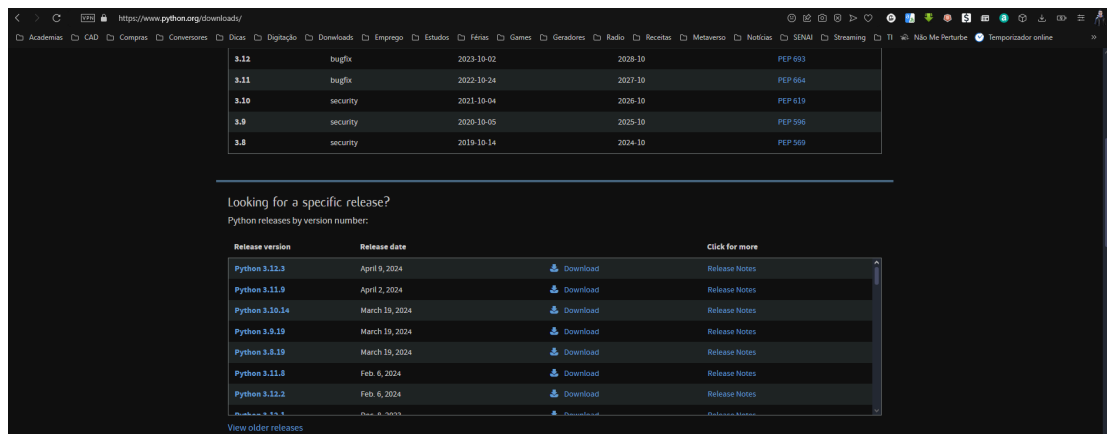
Instalando o Python (caso não tenha instalado)

A primeira coisa a ser feita é instalar o Python. O tutorial a seguir serve para a instalação do Python em ambiente Windows:

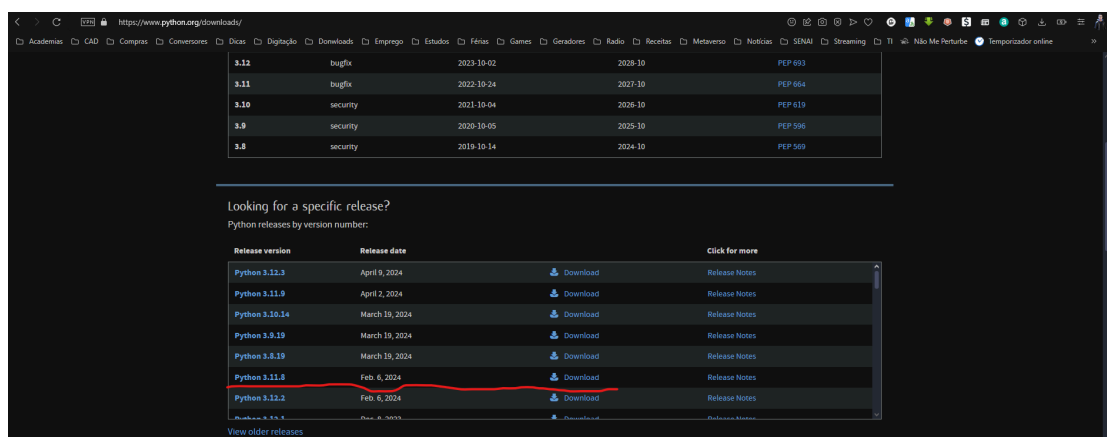
1. Acesse o site do Python: <https://www.python.org/downloads/>



2. Escolha a versão a ser instalada. **Obs:** a versão mais recente no momento da montagem desse material era a 3.12.3. Entretanto, nem sempre é mais vantajoso instalar a versão mais recente, como será explicado mais para frente no curso. Por esse motivo, vamos descer a barra de rolagem para procurar uma versão um pouco mais antiga. Segue a imagem abaixo:



3. A versão que iremos instalar é a versão 3.12.2. Baixe essa versão, seguindo a imagem abaixo:



4. Ao iniciar a instalação, certifique-se de marcar a caixinha que adiciona o caminho do Python ao PATH do Windows.
5. prossiga com a instalação normalmente até finalizar, e o Python estará instalado com sucesso.

Obs: a instalação do Python é importante para os próximos passos, e embora seja possível instalar o Python depois, é mais interessante começar as instalações dos programas por ele. Outro ponto importante a se ressaltar, é que no caso dos devs que usam Linux, o Python já vem instalado por padrão. Talvez a única coisa necessária a se fazer nesse caso é atualizar o Python. Siga as instruções do site para realizar a atualização.

Atualizando o Python (caso já tenha instalado mas precise atualizar)

É claro que dá para ir no site do Python, baixar uma versão mais recente e instalar, mas esse processo pode ser feito pelo terminal também.

E por que atualizariamos o Python?

Segurança

Manter o Python atualizado é essencial para garantir a segurança do seu sistema. As atualizações regulares incluem correções de segurança que protegem contra vulnerabilidades conhecidas. Ao atualizar o Python pelo terminal, você garante que está utilizando a versão mais recente e segura da linguagem.

Correção de bugs

Assim como em qualquer software, o Python também pode conter bugs. As atualizações fornecem correções para esses bugs, melhorando a estabilidade e o desempenho do Python. Ao atualizar pelo terminal, você pode corrigir problemas conhecidos e aproveitar as melhorias implementadas nas versões mais recentes.

Novos recursos

Uma das vantagens de atualizar o Python pelo terminal é ter acesso aos novos recursos e funcionalidades adicionados nas versões mais recentes. As atualizações trazem melhorias significativas para a linguagem, como suporte a bibliotecas atualizadas, melhor desempenho e novas sintaxes. Manter-se atualizado permite que você aproveite esses recursos e mantenha-se atualizado com as tendências do Python.

Como atualizar o Python pelo terminal?

1. Abra o terminal/prompt de comando.
2. No Windows, o comando de atualização do Python é `python -m pip install --upgrade python`.
3. Aguarde a instalação finalizar.
4. Verifique novamente a versão do Python instalada.

Como atualizar o Python no Linux

1. Siga o tutorial nesse site: <https://awari.com.br/como-atualizar-o-python-no-linux-guia-completo-e-pratico-2/>

Fonte

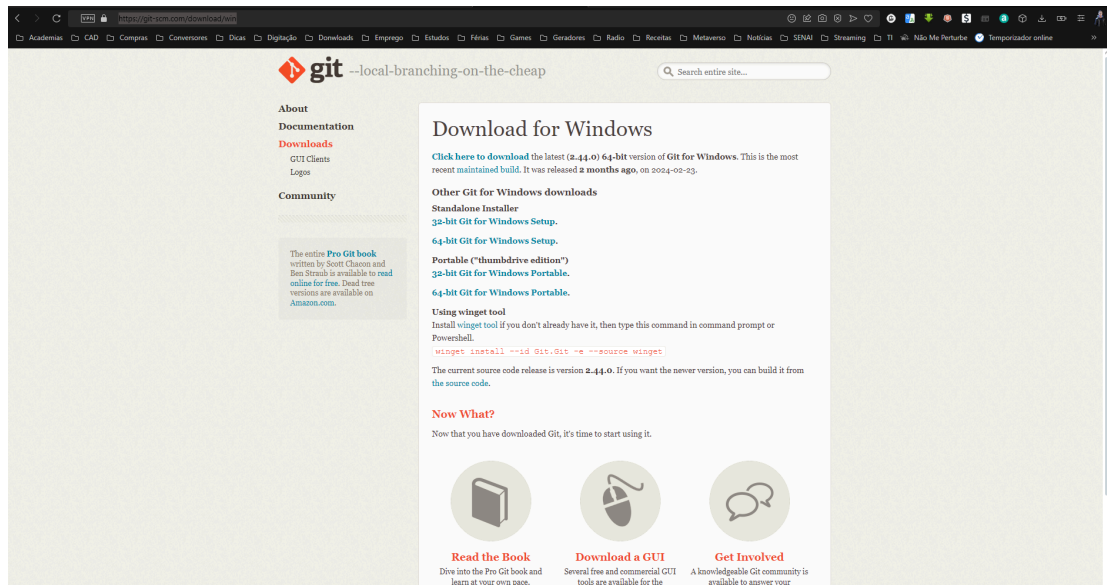
- <https://awari.com.br/python-como-atualizar-pelo-terminal/>

Git

O Git é um programa que trabalha com o chamado **versionamento** de código, importantíssimo para o controle das versões do seu programa e também para fins de backup, e para evitar perder trechos de códigos que funcionavam antes de uma determinada funcionalidade do seu sistema ser implementada.

Instalando o Git

1. Abra o navegador e vá até o site do git: <https://git-scm.com/download/win>



2. Mais uma vez, a versão a ser instalada aqui é para Windows. Se estiver usando outro SO, procure saber como fazer as instalações para o seu Sistema Operacional.
3. Escolha a versão do seu SO e faça o download.
4. Ao término do download, abra o executável e prossiga com a instalação.
5. Um ponto a ser mencionado aqui é a marcação da caixa que permite que o Git seja acessado a partir de outros prompts. Marque essa caixa, pois esse detalhe será importante.
6. Prossiga com a instalação normalmente, dando Next até concluir.

Visual Studio Code

O programa a ser utilizado nesse curso será o **Visual Studio Code**, mais conhecido como **VSCode**.



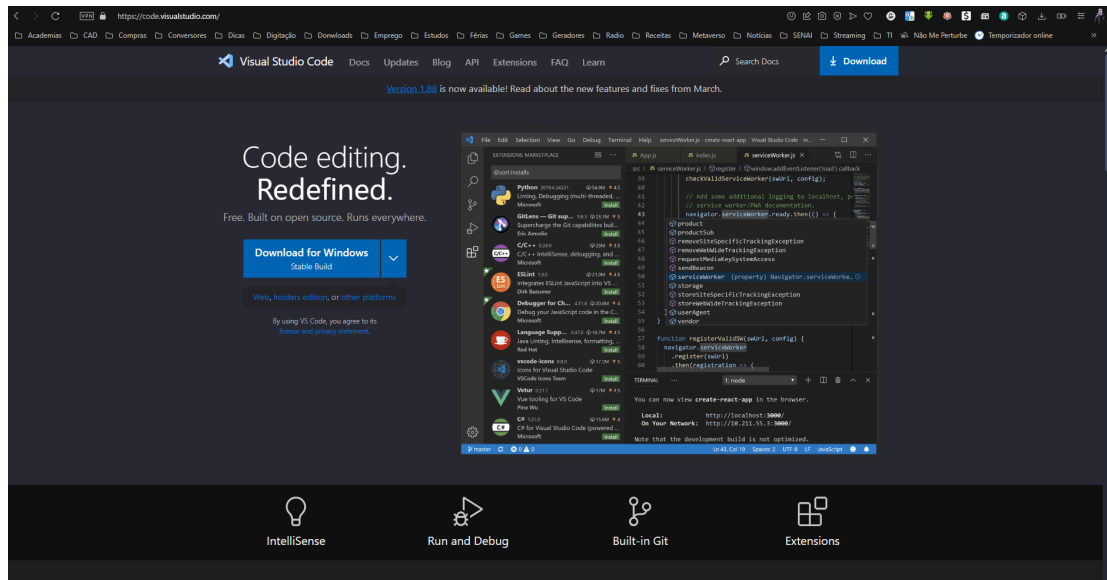
Apesar de ter menos recursos nativos que o PyCharm, principal IDE para Python no mercado, o VSCode permite uma grande liberdade na hora da configuração do ambiente. Isto, aliado à gigantesca quantidade de *plug-ins* disponíveis para o editor, permite um altíssimo nível de personalização do programa, fazendo com que o desenvolvedor deixe o software com a "sua cara". Além disso, podemos adicionar tantos recursos através dos plug-ins que no final o nosso VSCode terá tantos ou até mesmo mais recursos que a versão paga e mais completa do PyCharm. Isso tudo sem tirar um único centavo do bolso. Por esses motivos e também por outros que o VSCode é o **programa de desenvolvimento mais usado no mundo**. Tanto comparando com Editores de textos como IDEs.

Instalando o VSCode

A instalação do VSCode é ainda mais simples:

1. Vá até o site de download do VSCode: <https://code.visualstudio.com/>

Obs: Não confundir com o programa Visual Studio IDE, que possui um nome parecido, uma logo parecida também, e ambos são da Microsoft.



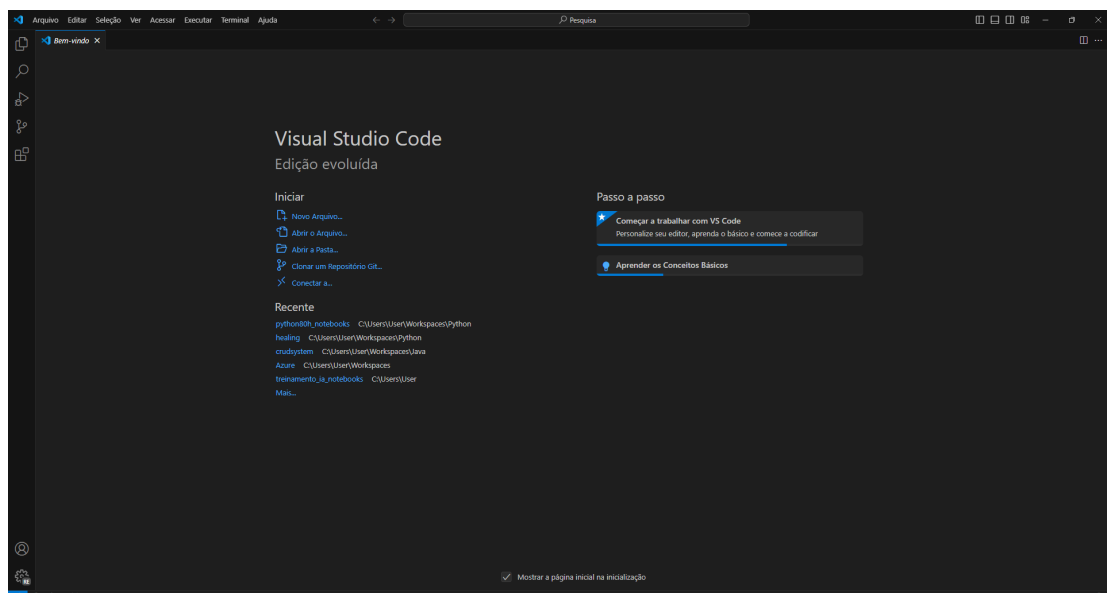
2. Faça o download normalmente para o seu Sistema Operacional.
3. Abra o executável e inicie a instalação normalmente.
4. Desta vez, não há nenhum procedimento especial durante a instalação. Apenas prossiga clicando em Next até concluir normalmente a instalação do programa.
5. Após finalizar a instalação de todos os programas, para garantir que tudo vai dar certo, reinicie o computador.

Configurando o VSCode

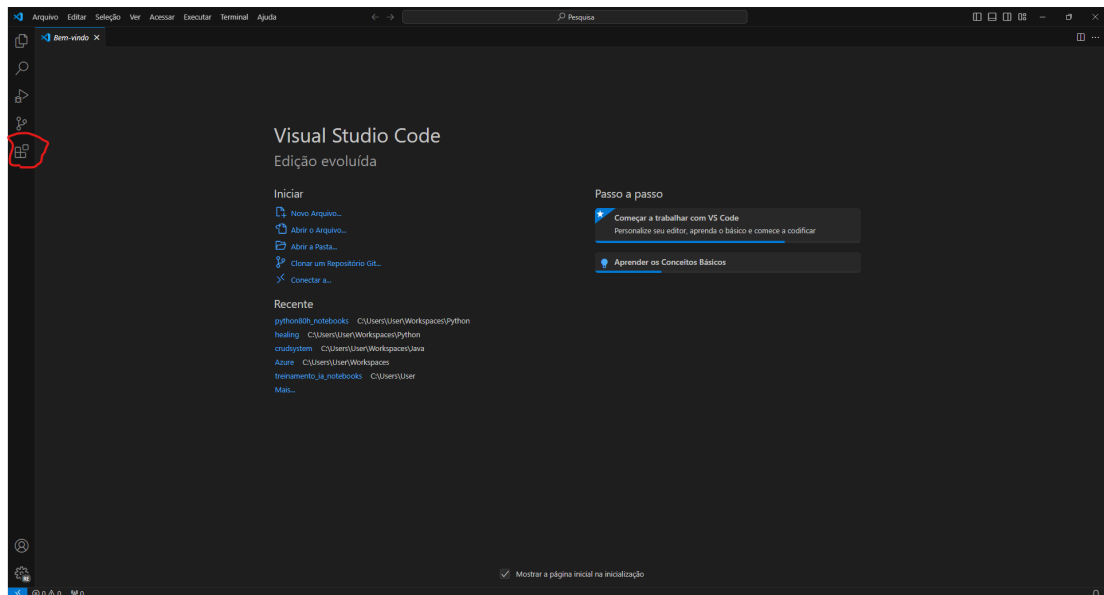
Após a reinicialização do seu computador, abra o VSCode para começarmos a configuração do programa.

Obs: não é objetivo desse tutorial ensinar a mexer no VSCode. Isso será ensinado pelo professor em sala de aula. Entretanto serão ensinados como acessar os recursos conforme precisarmos deles durante as aulas seguintes.

1. Ao abrir o VSCode pela primeira vez, essa será a tela a ser exibida:

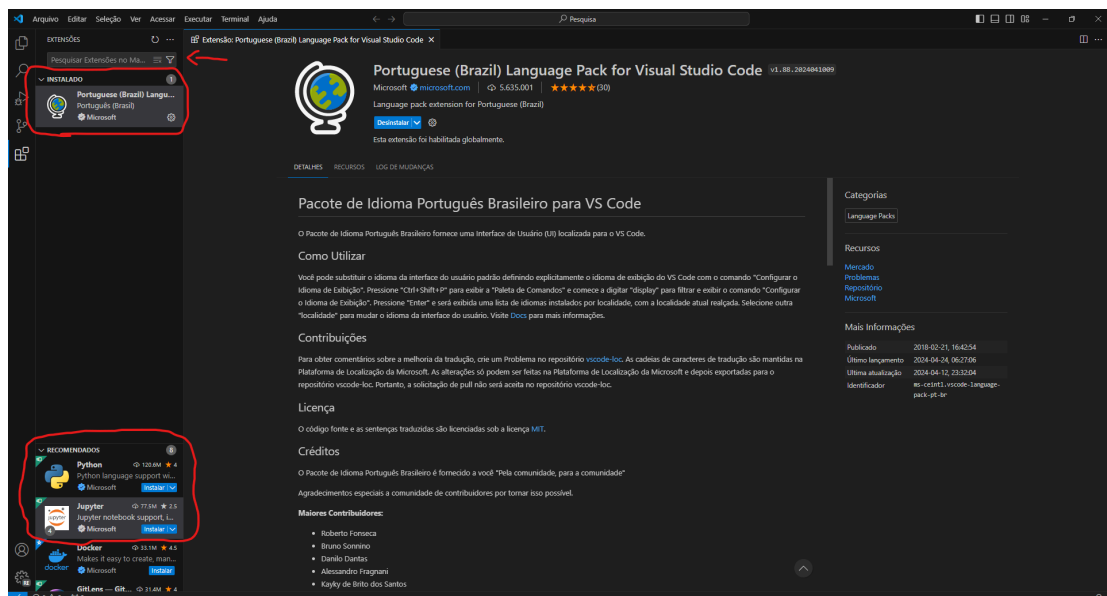


2. Logo de cara iremos instalar os plug-ins necessários para que possamos trabalhar com Python no VSCode. Portanto, clique no ícone **Extensões** do lado esquerdo da tela, conforme imagem abaixo:



3. Logo de cara, iremos instalar as extensões da Microsoft necessárias para o desenvolvimento de programas Python. São elas:

- Portuguese (Brazil) Language Pack for Visual Studio Code
- Python
- Jupyter



4. Essas extensões são suficientes, mas podemos incrementar ainda mais o VSCode para aprimorar nossa experiência, e não dependermos de nenhum outro programa externo para o desenvolvimento das aplicações. Portanto, procure e instale as extensões abaixo:

- Comment Anchors
- Create Jupyter - Buttons
- Django
- Git Graph
- Live Share
- Live Share Whiteboard
- Material Icon Theme
- Polacode
- Python Environment Manager
- Python Resource Monitor

5. Aproveite e instale também um tema do VSCode à sua escolha, para uma melhor personalização.

Após essa configuração, seu computador finalmente vai estar preparado para codificar em Python!!!

Opicional: Anaconda Navigator



O Python é uma das melhores, se não for a melhor, opção atualmente para ciência de dados e inteligência artificial, e junto com ele vem diversas ferramentas para ajudar o desenvolvedor nessa tarefa. Uma forma opcional de configurar o ambiente da sua máquina para o desenvolvimento Python, caso o objetivo seja mineração de dados é instalar o **Anaconda Navigator**.

O **Anaconda Navigator** é um *launcher* que funciona como um pacote de aplicativos voltados para ciência de dados, ou seja, ele centraliza em uma única solução tudo o que você precisa para trabalhar com ciência de dados. Uma das vantagens de se instalar esse *software* é que ele já instala automaticamente muitas das ferramentas a serem utilizadas por esse tipo de profissional. Por exemplo, ele já instala uma versão do Python, do Jupyter Notebook e do Jupyter Lab, sem necessidade de instalação prévia dessas aplicações. Ele também pode reconhecer algumas aplicações previamente instaladas e reuni-las dentro do *launcher* do Anaconda, como o VSCode e o PyCharm.

Por ser um *software* relativamente pesado, e mais voltado para ciência de dados do que para outros tipos de atividades, inicialmente não iremos trabalhar com ele, mas sinta-se à vontade para instalá-lo e utilizá-lo para desenvolver suas soluções em Python. A instalação desse programa é feita da forma tradicional para os usuários de Windows: indo no site de download (<https://www.anaconda.com/anaconda-navigator>), baixando o executável e instalando normalmente, clicando em **Next** -> **next** -> **Install** -> **Finish**, conforme o instalador for pedindo, e ao final da instalação, você terá o Python e todas as soluções que precisar reunidas dentro do Anaconda.

Exercícios

1. Qual programa iremos utilizar para desenvolver códigos Python?

- ☐ VSCode
- ☐ Python
- ☐ Git
- ☐ Visual Studio IDE

2. Qual a função do Git?

- ☐ Desenvolver códigos Python
- ☐ Versionar códigos
- ☐ Executar códigos
- ☐ Preparar uma Receita de Bolo

3. Qual a extensão de um arquivo Python?

- ☐ .java
- ☐ .python
- ☐ .py
- ☐ .exe

4. Marque abaixo o nome das extensões necessárias para codificação Python no VSCode:

- ☐ Python
- ☐ Live Server
- ☐ Synthwave 84 Theme
- ☐ Jupyter