

What is Programming?



Programming is like giving instructions to a computer to perform tasks



We use a recipe to cook a delicious meal, programmers use a set of instructions also known as code, to make a computer do what they want.

Programming is like giving instructions to a computer to perform tasks. Just like we use a recipe to cook a delicious meal, programmers use a set of instructions also known as code, to make a computer do what they want. These instructions can range from simple calculations to complex data manipulations, and they are executed by the computer step by step.

So, at its core, programming is a means of communication between humans and machines. And programming languages are the tools we use to communicate with computers.



We have various spoken languages for communication between people



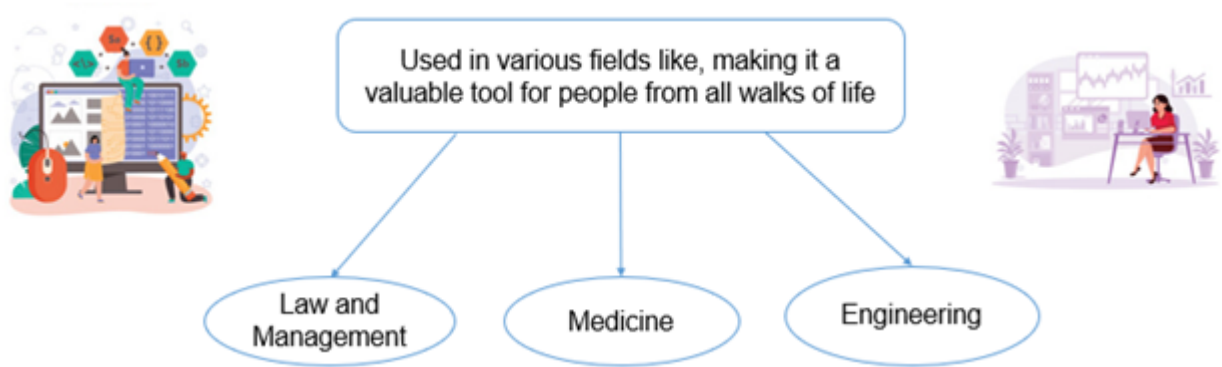
There are many programming languages designed to interact with computers



Every programming language has its own syntax, rules, and capabilities, making it suitable for specific tasks.

We have various spoken languages for communication between people, there are many programming languages designed to interact with computers. Each programming language has its own syntax, rules, and capabilities, making it suitable for specific tasks.

Python is one such programming language which is the most popular and beginner-friendly programming language.

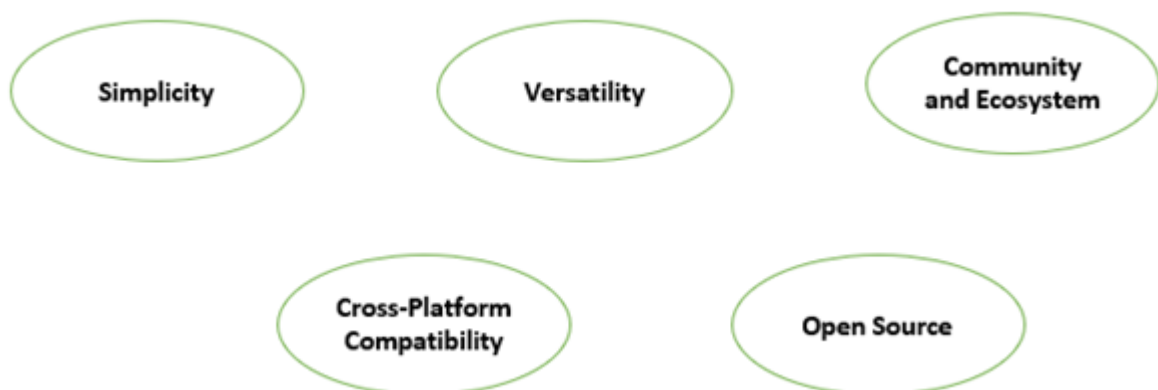


It has found its way into various fields, making it a valuable tool for people from all walks of life. Whether you're interested in law, medicine, engineering, management, or any other domain, Python can be your key to unlocking endless possibilities.

History of Python:

Python was created by Guido van Rossum and was first released in 1991. It was designed to be easy to read and write, emphasizing code readability and simplicity. Python's name is not related to snakes but is inspired by the British comedy group Monty Python. Over the years, Python has evolved, becoming one of the most popular and widely used programming languages in the world.

What Makes Python Stand Out:



Simplicity: Python is like a recipe written in plain and clear language. You can follow it easily because it's not filled with complex jargon or long-winded instructions.

It's clear and concise syntax means that the code you write is easy to understand and read. This encourages good programming practices because it's harder to make mistakes when the code is straightforward. It also reduces the cost of maintaining the program

over time, as it's easier for different people to work on and understand the code.

Versatility: Python is a versatile language that can be used for a wide range of applications. Think of Python as a Swiss Army knife. It can be used for many different tasks, just like a Swiss Army knife that can have various tools for different purposes. You can use it for web development, analyzing data, creating artificial intelligence, automating tasks, and more. Its extensive standard library and third-party packages, which we'll learn later in the program but for now consider them as additional tools, make it suitable for various tasks without needing to start from scratch each time.

Community and Ecosystem: Python has a large and active community of developers. This community is like a bustling marketplace where you can find help, resources, and tools. If you have a problem or need advice, you can ask others in the Python community for support. Additionally, the wealth of available resources and libraries (tools and products) created by this community can simplify your projects and save you time.

Cross-Platform Compatibility: Python is available on various platforms, which means you can use it on different types of computers and operating systems. Just like a universal language allows people from different backgrounds to communicate, Python allows you to write code that works on Windows, macOS, or Linux, making it a portable choice for development. It is like a universal language that people from different countries can understand. It doesn't matter which country you go to; you can still communicate.

Open Source: Python is open source, which means it's free to use. It's like a playground that anyone can access without paying an entrance fee. You can also see how Python itself is built because its development process is transparent. This openness encourages collaboration and innovation within the Python community, and you can use it without any cost.

Python's Relevance in Different Fields:



Python's Relevance in Different Fields



Python serves as a **versatile tool** for professionals from various academic backgrounds.

Management

Commerce

Economics

Finance

Law

Technology

Python serves as a versatile tool for professionals from various academic backgrounds. In management, it streamlines data analysis, project management, and decision support through its readability and cross-platform compatibility. In commerce, it automates financial tasks, enhances business intelligence, and aids in market research.

For economics, Python facilitates econometric modelling, statistical analysis, and policy assessment, enabling robust research and forecasting.

In finance, it's critical for risk management, quantitative analysis, and asset pricing. Python is equally valuable in law for legal document processing and analysis. It's also indispensable for statistics, offering advanced statistical modeling and data visualization. In the tech realm, Python is a cornerstone for web development, data science, and automation, providing an interdisciplinary advantage.

Proficiency in Python offers a competitive edge in various career paths and has become a vital skill in today's dynamic job market.



Python in Tech Industry



NETFLIX

Python is also a highly popular language in the tech industry. Many companies, including Google, Instagram, and Dropbox, use Python in their products and services. So, if you're interested in programming or looking to expand your skills, Python is an excellent language to learn. With its clear syntax, a wide range of applications, and thriving community, Python is the perfect choice for anyone looking to get started in programming and especially data science.

Installation of Python

Python does not come inbuilt in Windows so we must install Python in our computers to use it. It is like installing any other software or mobile application.

To work with Python, we need to have two things installed in our computers.

- The Python interpreter
- The Python IDE

To work with python, we need to have two things installed in our computers.

The python interpreter

The computer cannot understand python code Directly. code written in Python needs to be interpreted by the Python interpreter to run

Example: someone visiting France and not knowing French



To understand You need to translate it in your native language

Python IDE

'IDE' stands for Integrated Development Environment. To be efficient and hassle free it is best to use an IDE

Example: A mechanic always has his tool box that contains all the tools required to fix a problem



IDE comes with a text editor, a debugger, a code checker, Code testing tools and a lot more.

Python Interpreter

The computer cannot understand Python code directly. The code written in Python needs to be interpreted by the Python interpreter to run. Take an example of someone visiting France and not knowing French. To understand what people are saying you need to translate from French to your native language.

The Python interpreter is software that translates Python code into machine code, which is binary in nature and is run on the computer so that the computer can understand the instructions given in Python.

Python IDE

'IDE' stands for Integrated Development Environment. After installing the Python interpreter, we can run the Python code even using a notepad or any text editor but that will not be very efficient and involves too many steps. To be efficient and hassle free it is best to use an IDE.

A mechanic always has his toolbox that contains all the tools required to fix a problem. Similarly, an IDE contains all the tools required at different stages of coding at a single place. There are many IDEs available on the internet to use for free. Some famous ones are Spyder IDE, PyCharm etc.

In our course of program, we will be using Anaconda. You can also use your preferred IDE

Anaconda Navigator:

Anaconda is a powerful platform for data science and scientific computing. When you install Anaconda, you do not need to install Python or any IDE separately. Inside the Anaconda platform, you will have the option to install different IDEs and tools required for data analysis.

What is Anaconda and why are we using it?



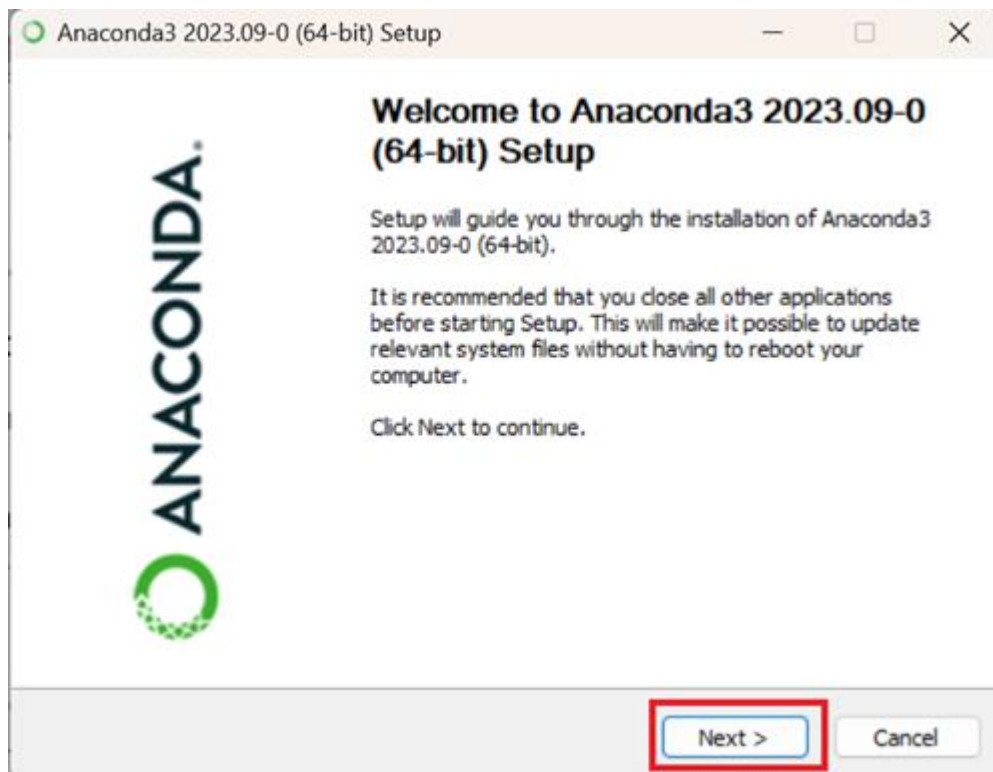
Anaconda is a powerful platform for data science and scientific computing

When you install Anaconda, you do not need to install Python or any IDE separately.

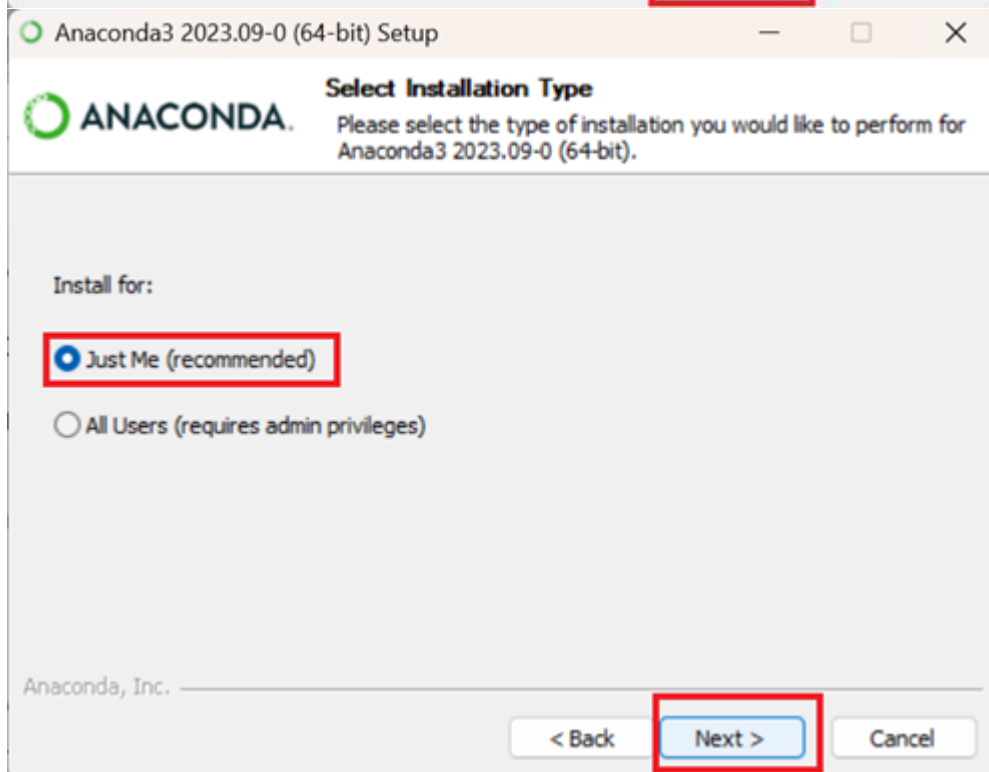
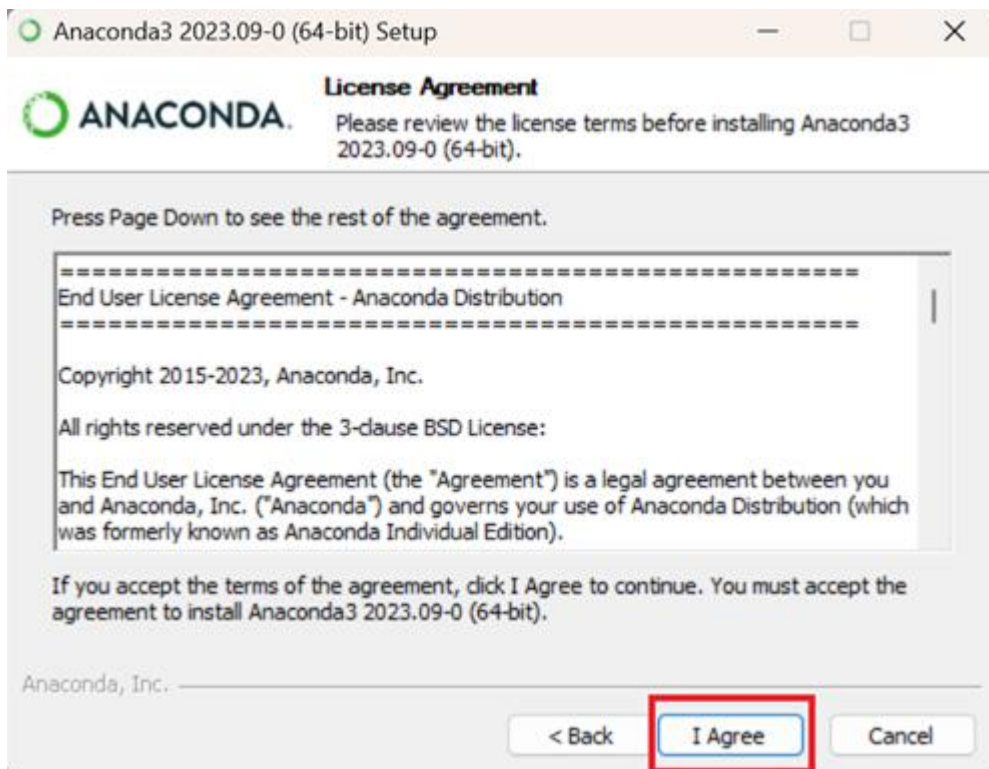
Inside the Anaconda platform, you will have the option to install different IDEs and tools required for data analysis.

let's begin by downloading Anaconda:

- Go to <https://www.anaconda.com> and click on the "Download" button on the page.
- Wait for the download to finish, and then run the downloaded file (**double-click the file**). You will see an installation window something like this.

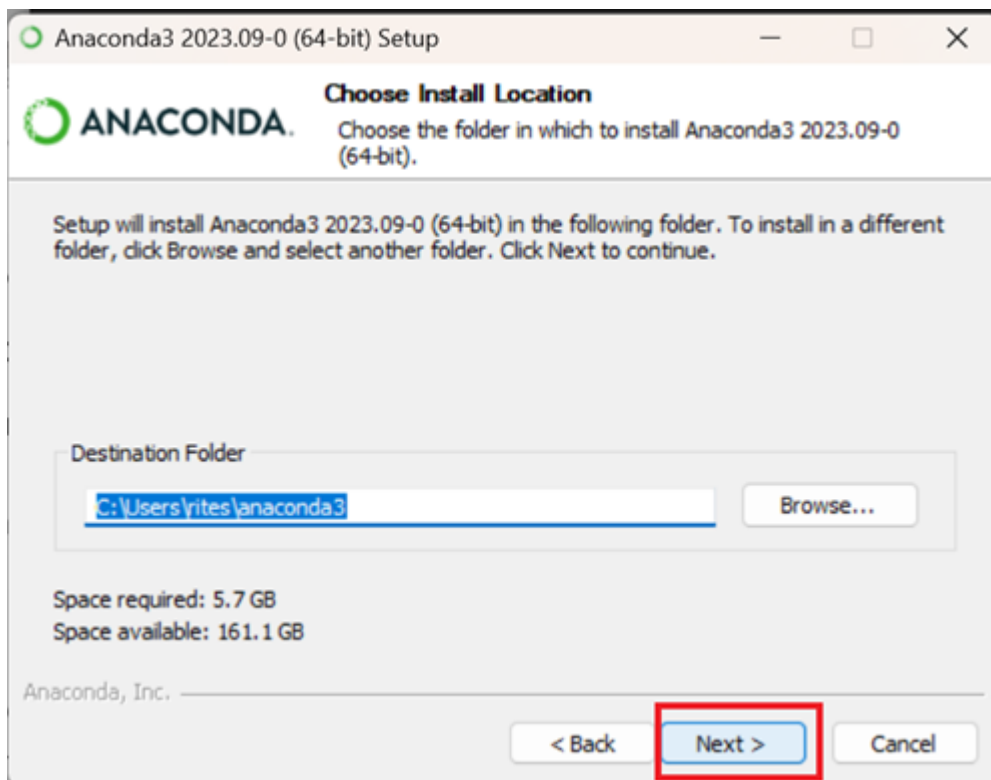


- Click **Next** to proceed further. You will see the next window.

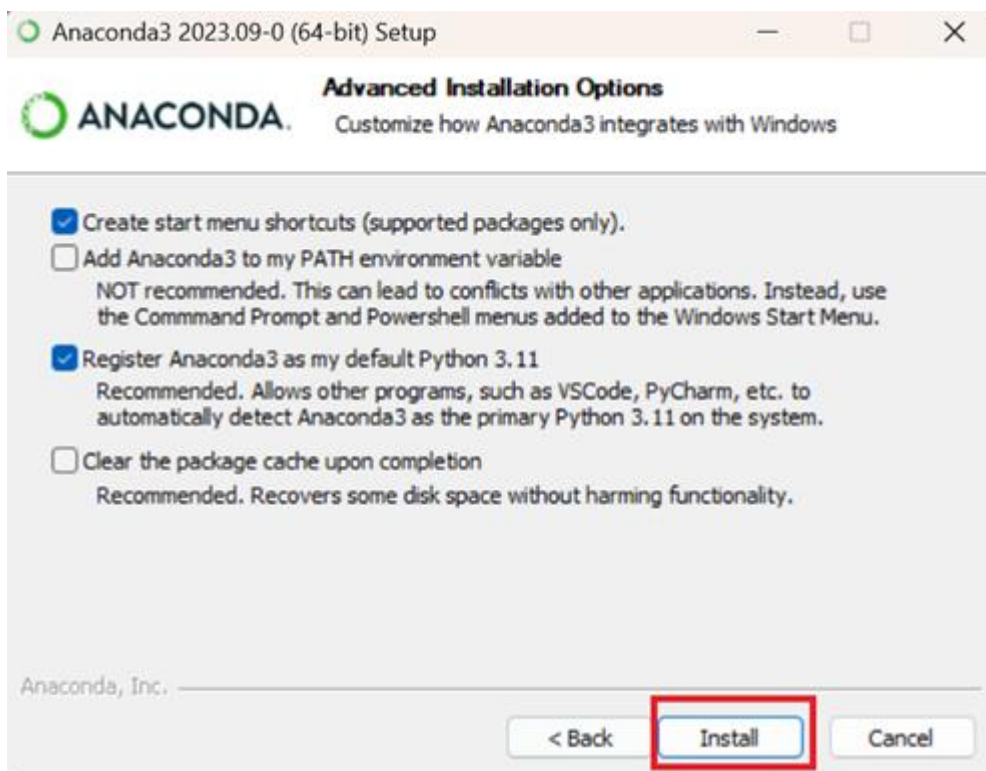


Read the terms and license agreements if you want, and click on “**I Agree**”

In the next window select the “Install for**Just me**”option and click **Next** to proceed to the next window.



Now click **Next** to proceed further to the final installation window.



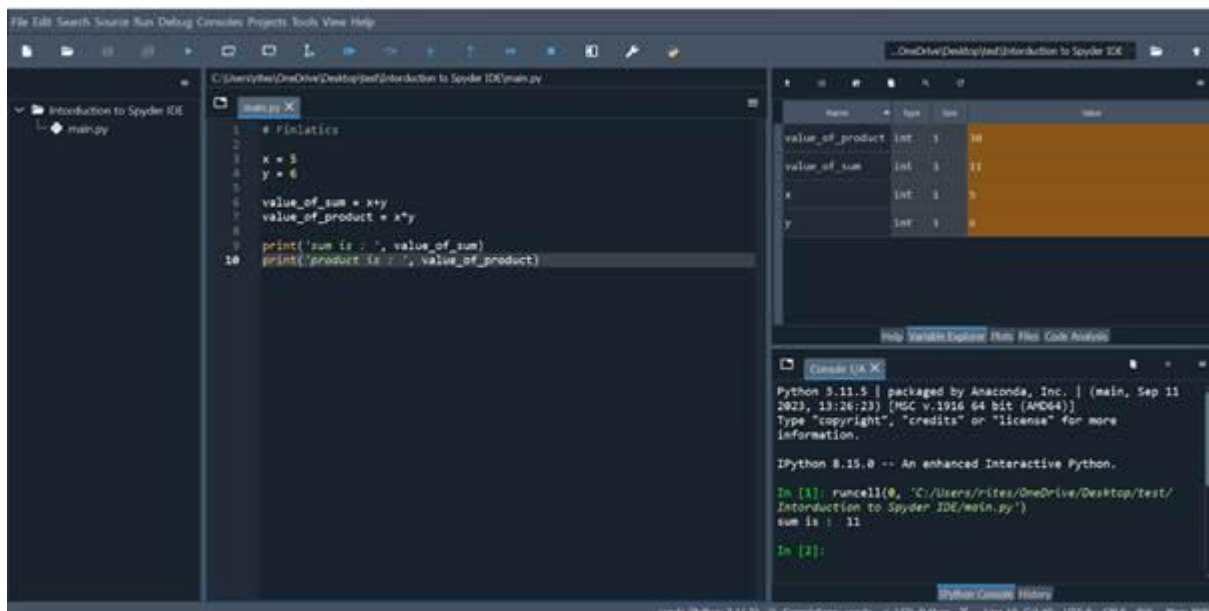
- Click Install and wait for the process to complete.

And that is it. You have successfully installed Anaconda Navigator on your computer and now you can use the Anaconda Navigator platform to install Spyder IDE if you'd like to work with Spyder.

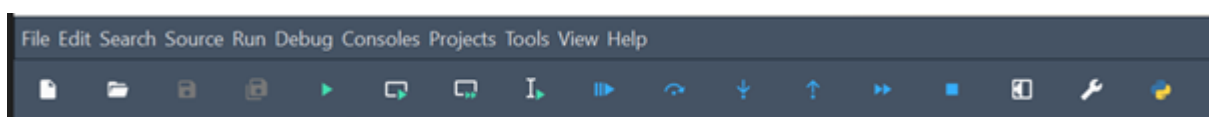
To do so, follow the instructions:

- **Open Anaconda Navigator**, which is the user-friendly interface for Anaconda.
- Click on the "**Environments**" tab on the left-hand side of the window.
- Click on the "**Create**" button to create a new environment.
- Name your new environment and select the Python version you want to use and wait for the setup to finish.
- Now, in the "**Packages**" list, change the default of the searching list from **Installed** to **All** in the searching list and then search for "**Spyder**" and **select the checkbox next to "Spyder"**.
- Now click on the "**Apply**" button to install Spyder in your new environment.

Spyder IDE At a glance:



Starting from the very top, first comes the menu:

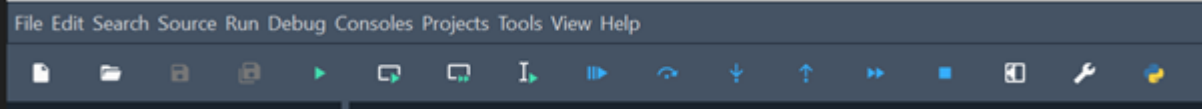




This menu has various options for opening, creating, and editing files and has the option to change the IDE settings and preferences. In the

Edit section, It has the option to undo or redo any edits made to the code, it has the option to indent or unindent the code. Apart from that it also has different options for running the Python code.

In the Projects menu, we get the option to create new Python projects and open, close or delete existing ones. In the File menu, we can create, edit, save, and open Python files. The Run menu provides us with various options for running the Python code.

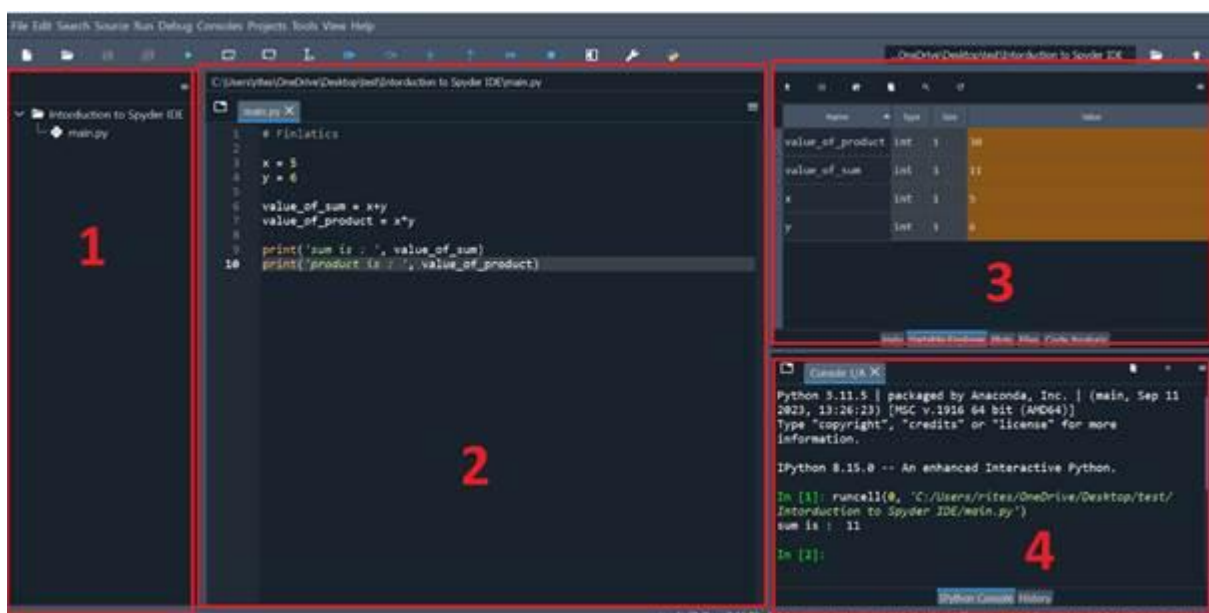
The most frequently used options are also listed below as shortcut icons.

	
	This is the Run File icon; it runs the entire python code at once.
	This is the run current line option icon; it runs a single selected line of code.

There are many different things to be explored but this is what we will be using regularly. Let us move on with the other features for now.

The 4 Main Panels:

The Spyder IDE has 4 main panels as can be seen in the following image marked as 1, 2, 3, 4.



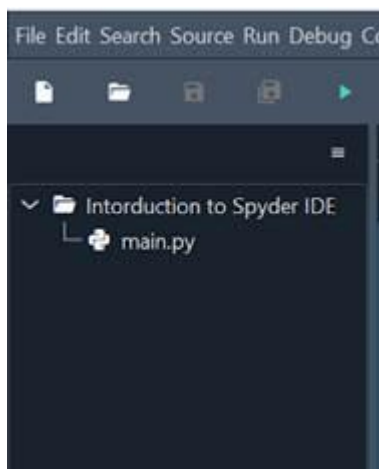
Number 1 is the file explorer, number 2 is the text editor, number 3 has different sub panels such as help, variable explorer, plots, files and code analysis, number 4 is the python console. Let us understand each component in detail.

Panel Number 1 (The File Explorer): The panel marked as number 1 is the file explorer option in the Spyder IDE allows the user to browse through the files and folders in the working directory. It allows us to open files in the text editor as well.

If you don't know what working directory is, don't worry its nothing fancy. Let us learn about working directory.

In Python or Spyder projects, think of the working directory as the main folder where the computer looks for and stores files related to our coding projects.

It's like the special folder where Python expects to find the files it needs, such as your Python scripts or data files. When we run our code, Python looks in the working directory to locate everything it requires for our project.

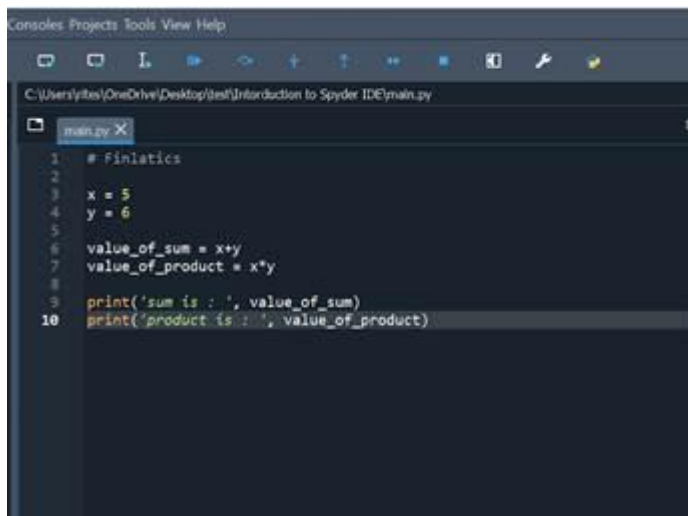


So in simple words, the Working directory refers to the folder inside which you have your project. Suppose if I create a folder named 'Introduction to Spyder IDE' and start coding in the folder then this folder 'Introduction to Spyder IDE' is my working directory.

As we can see our project "Introduction to Spyder IDE" in the file explorer which has one python file (main.py).

Another crucial thing that we need to understand is that we need to run the code for any project from the working directory of that project or else we might run into errors.

2. Panel number 2 (Text Editor): The number 2 panel is the text editor where we can write and edit Python code. This is where we will be writing the Python code.



```
1 # Finlatics
2
3 x = 5
4 y = 6
5
6 value_of_sum = x+y
7 value_of_product = x*y
8
9 print('sum is : ', value_of_sum)
10 print('product is : ', value_of_product)
```

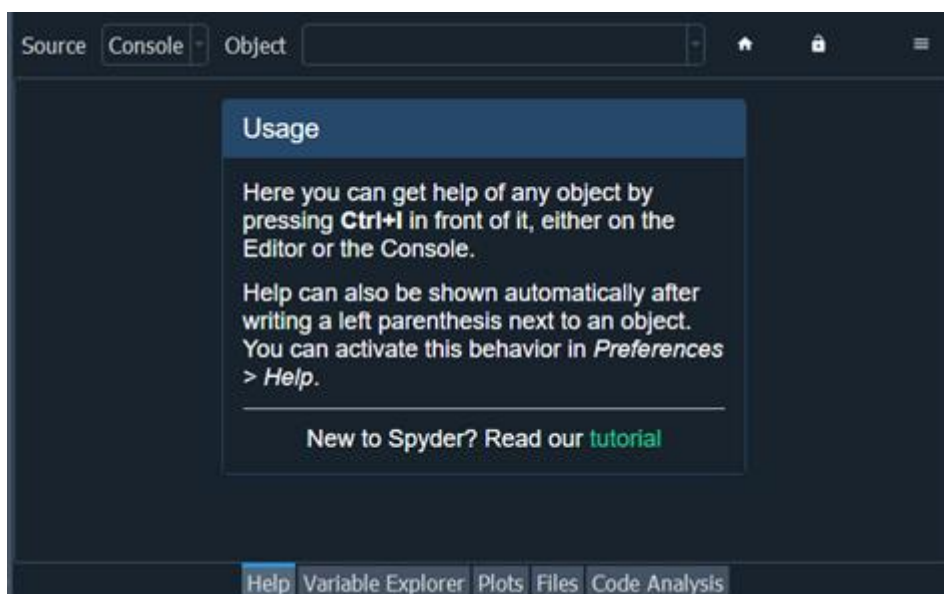
''' We can see our main.py file is open in the editor, and we have written some code in it. ''' We can open multiple files in the text editor. Here we can also select a single line of code and run it individually using the Run Current Line option mentioned above.

Line numbers are displayed on the left side to help navigate through the code easily.

3. Panel Number 3: The third panel has different sub-panels such as help, variable explorer, plots, files, and code analysis. Each has a different utility. Let us understand them individually.

Name	Type	Size	Value
value_of_product	int	1	30
value_of_sum	int	1	11
x	int	1	5
y	int	1	6

a. Help Panel: Here we can get help of any object by pressing Ctrl + L in front of it, on the Editor or the Console. We can understand different functions and commands in Python and get information about what they do and how can we use them, by pressing Ctrl + L in front of the code chunk.



b. Variable Explorer:

In programming, think of a variable as a storage box with a name. You put something (like a number, a word, or other information) in the box, and you can take it out and use it whenever you need. For example:

age=25

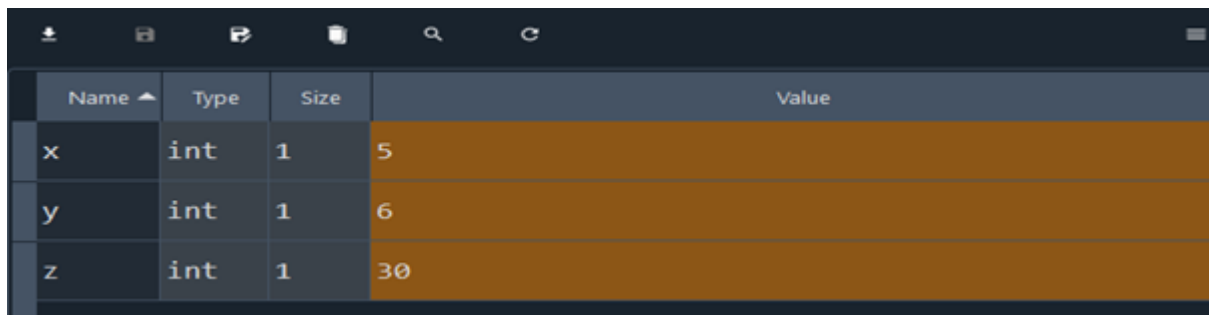
Here, age is like the name of the box, and 25 is what's inside. Later in your code, you can use age to refer to the value 25.

The Variable Explorer gives us information on the name, size, type, and value of each Variable. Let us understand this practically.

If we run the Python code written below where we are assigning the variable x a value of 5 and variable y a value of 6 and storing the product of x and y in variable z.

1. `x = 5`
2. `y = 6`
3. `z = x*y`

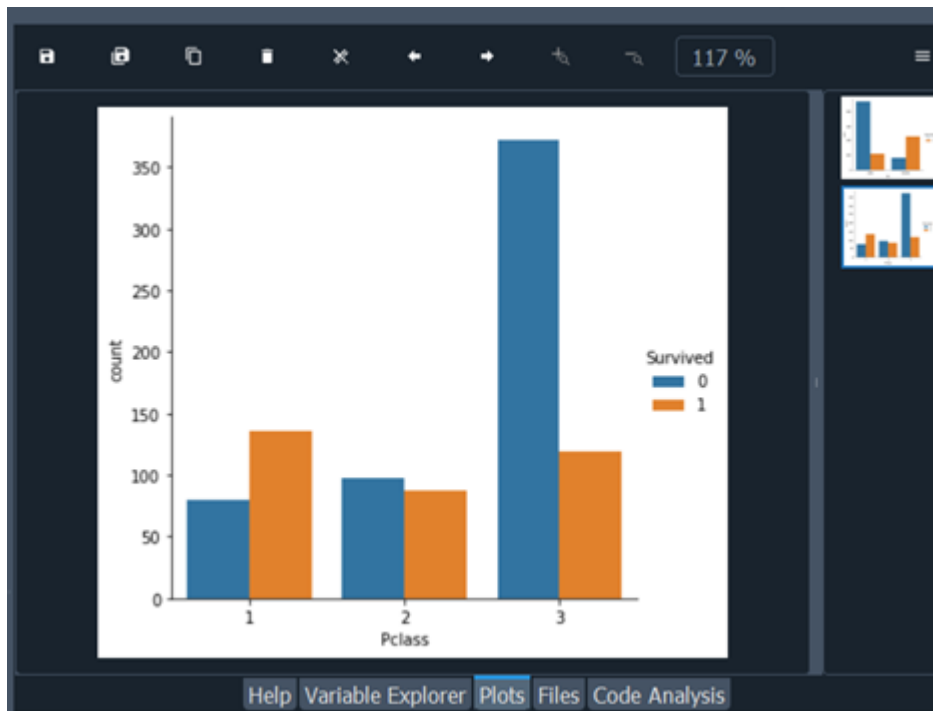
We can see the variable explorer will have these variables defined in it.



Name	Type	Size	Value
x	int	1	5
y	int	1	6
z	int	1	30

It also tells us the name, type, size, and value of the defined variables. We can also edit these variables in the variable explorer itself by double clicking the variable.

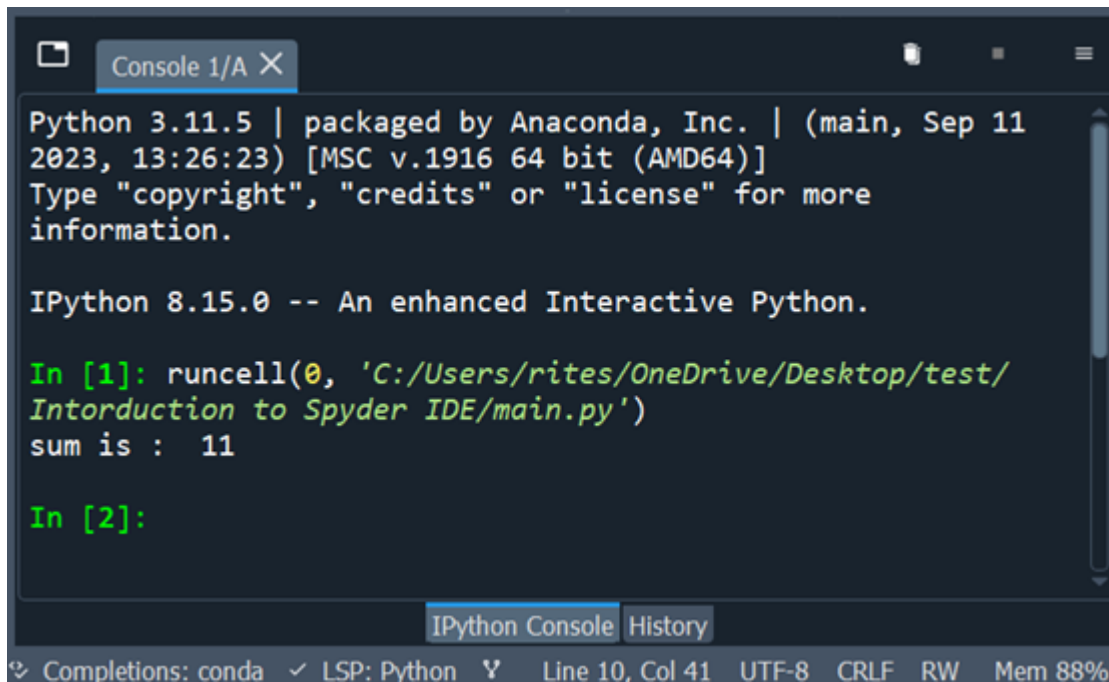
Plots: The Plots pane shows the static figures and images created during our session. It will show you plots from the Python Console, produced by our code in the Editor or generated by the Variable Explorer allowing us to interact with them in several ways. Here is an example of a plot shown in the plots pane.



We can also save, copy, or delete these plots using the save button in the above menu. We will see it in action further in the course.

d. The files pane: This pane shows the files of the project. Here we can delete, create, rename, or edit a file. This is very much same as the file explorer pane. It also gives an option to browse and change the working directory.

4. Panel Number 4 (Python Console): This is where the Python code is executed and we can test small parts of code in a console but anything we type here is not saved. We write the code in the editor and when we execute our code it runs in the console. If there is an error in the code the console will print the error or if everything works fine it will show the output.



```
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more
information.

IPython 8.15.0 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rites/OneDrive/Desktop/test/
Intorduction to Spyder IDE/main.py')
sum is : 11

In [2]:
```

IPython Console History

Completions: conda ✓ LSP: Python Y Line 10, Col 41 UTF-8 CRLF RW Mem 88%

You can also type Python commands directly into the console, and the computer executes them right away. It's like giving your computer direct instructions on the spot. For example, if you want to calculate the sum of two numbers, you can type `3 + 5` in the console, press Enter, and the console will immediately respond with 8.

Now suppose while writing a code we might be stuck at any point and are not sure about the output of the single line. We have the option to test the code by copy pasting the code to the command line in console and see the output of the code.

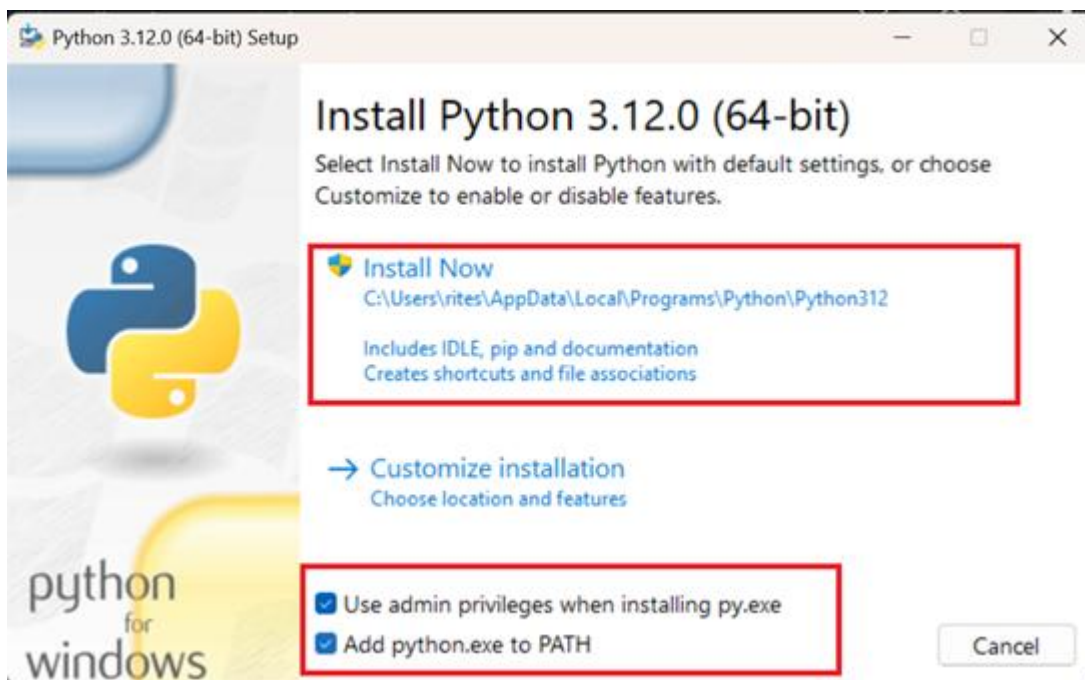
As we can see in the image there is also a “History” pane in the panel. This pane has the history of all the codes run in the console.

These panes collectively provide a comprehensive environment for Python development, offering features that support coding, debugging, data exploration, and project management. The layout and availability of these panes can be customized based on your preferences and workflow.

You can use the following instructions if you want to work with any other IDE Apart from Anaconda. For that, you'll need to first install Python in your system.

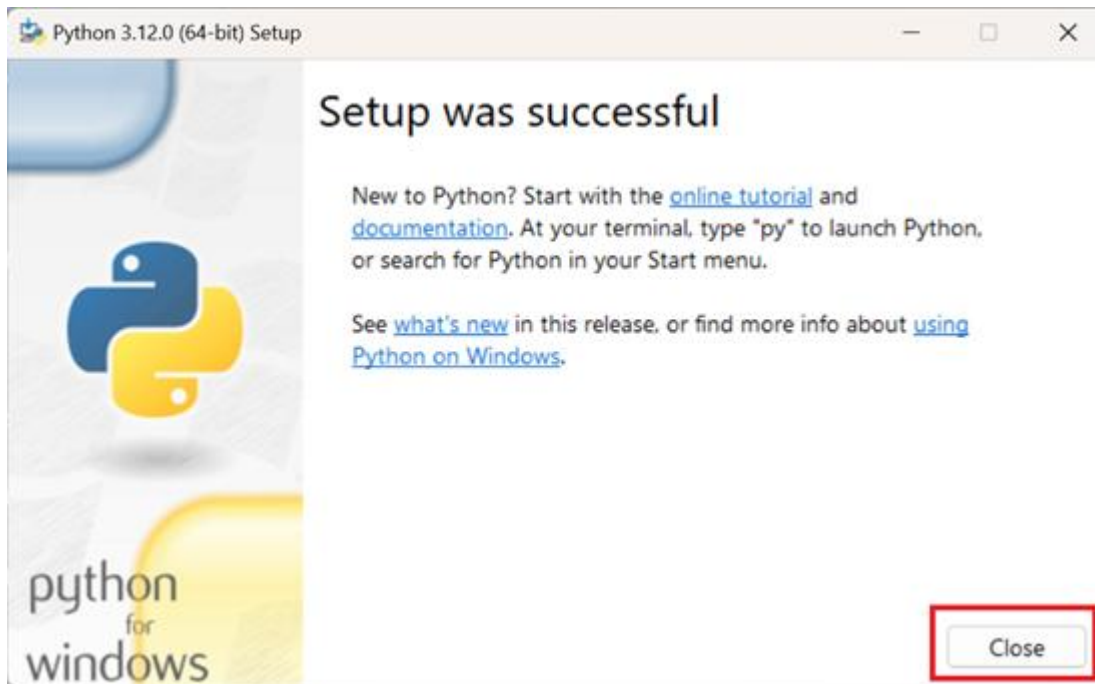
Installing Python:

- First, we need to download Python by visiting <https://www.python.org/downloads/> and clicking download.
- We recommend that you download the latest version from the link.
- After the download, we need to run the downloaded file(double-click on the file).
- When you run the file, you will see something like this:



The UI might change at times but this should be the same overall.

- Now, check the “*Use admin privileges when installing py.exe*”and “*Add python.exe to PATH*”.This is very important as **python might not work properly if you leave these unchecked** .
- Now, all you need to do is to **click “ Install Now ”** IDE comes with a text editor, a debugger - which is used to check bugs in programs, a code checker, Code testing tools, and a lot more.
- and wait for the installation process to complete.
- When the installation is over a “Setup was successful” Message will be displayed.



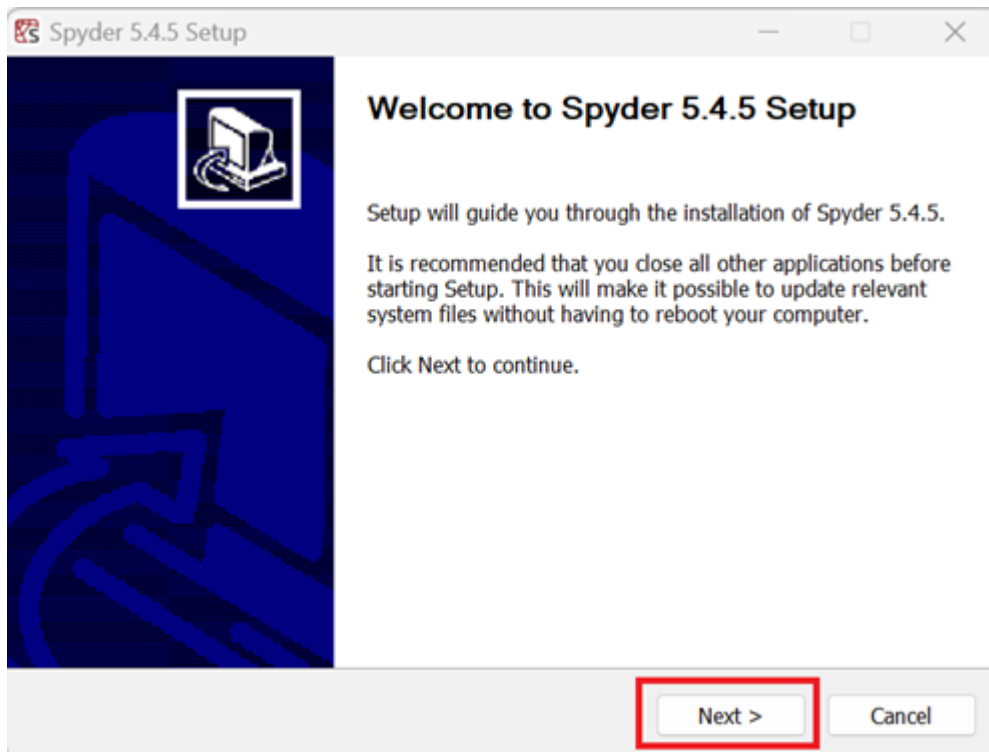
- Now you can click the “Close” button on the bottom right corner of the installation window.

Installing Python IDE:

There are many IDEs to use as mentioned earlier. Throughout this program we will be using Spyder IDE.

Let us install the Spyder IDE:

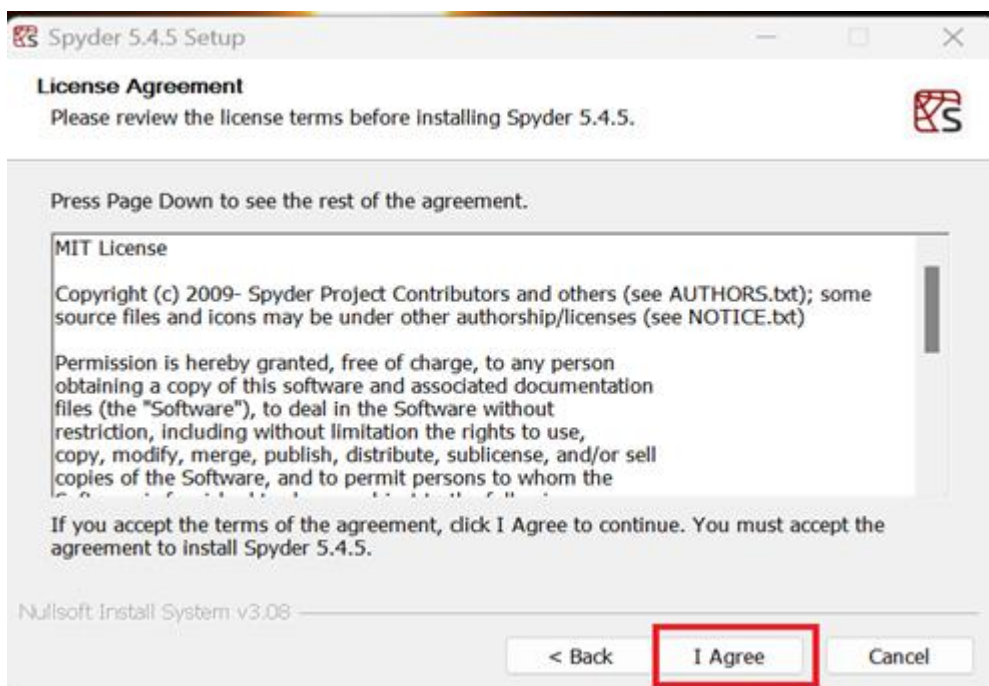
- We can download the Spyder IDE by **visiting** <https://www.spyder-ide.org/#section-download> and **clicking the download button** and wait for the download to finish.
- After the download is complete, we need to run the downloaded file(**double click on the downloaded file and allow**). You will see the installation window something like this:



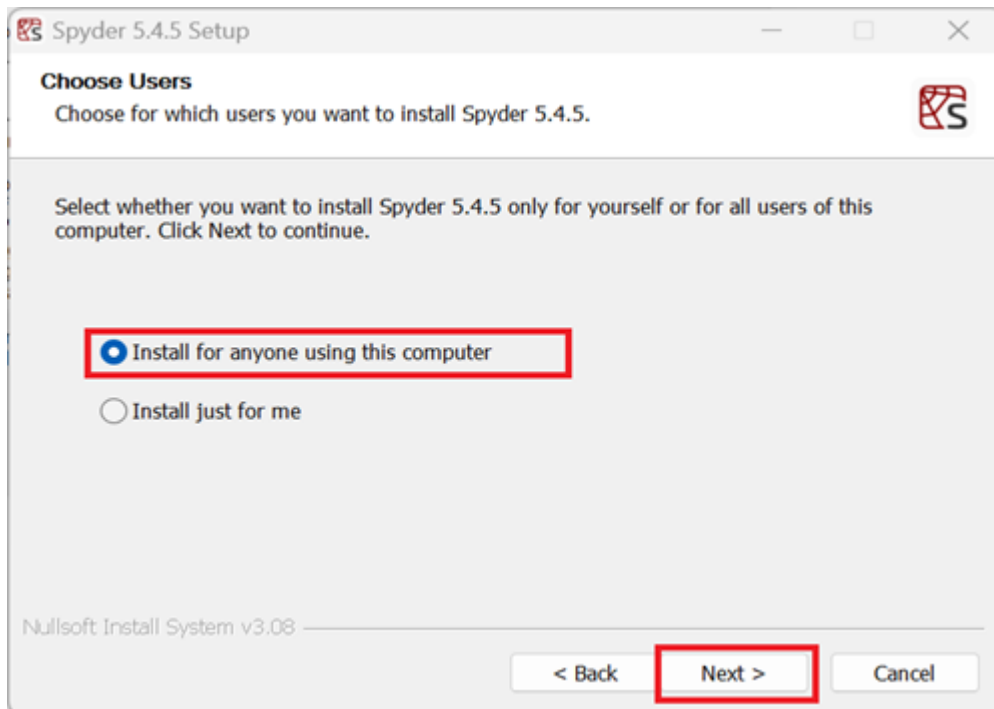
This UI might change at times but the process will remain the same.

- Click **Next** to proceed.

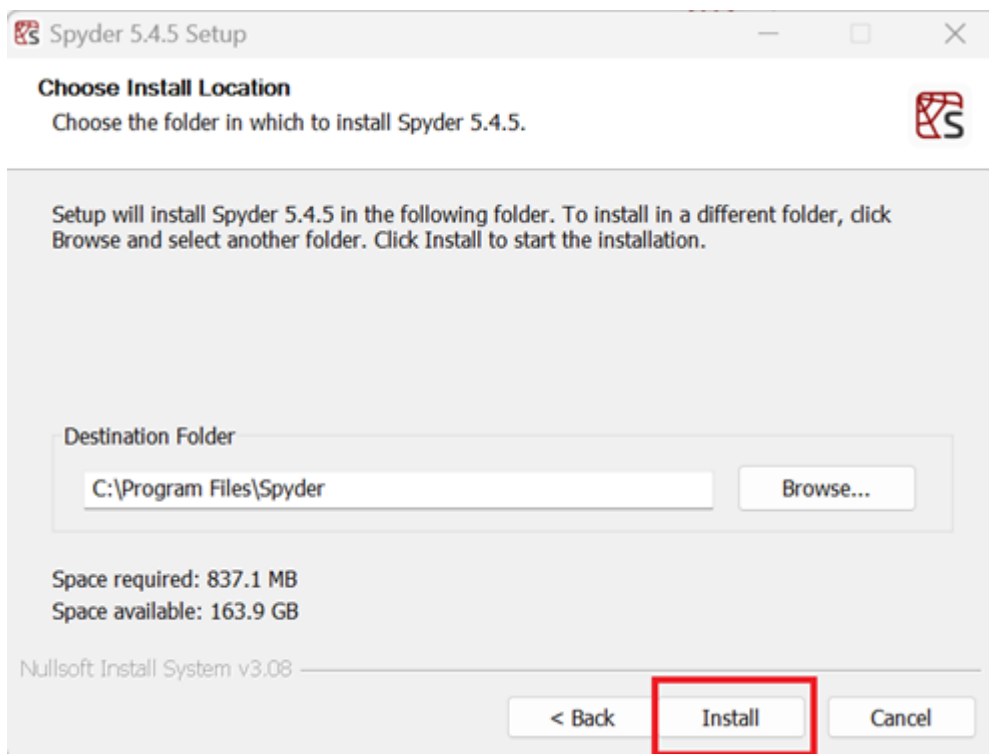
Now read the terms and license agreements if you want and click on **"I Agree"** to proceed as shown below:



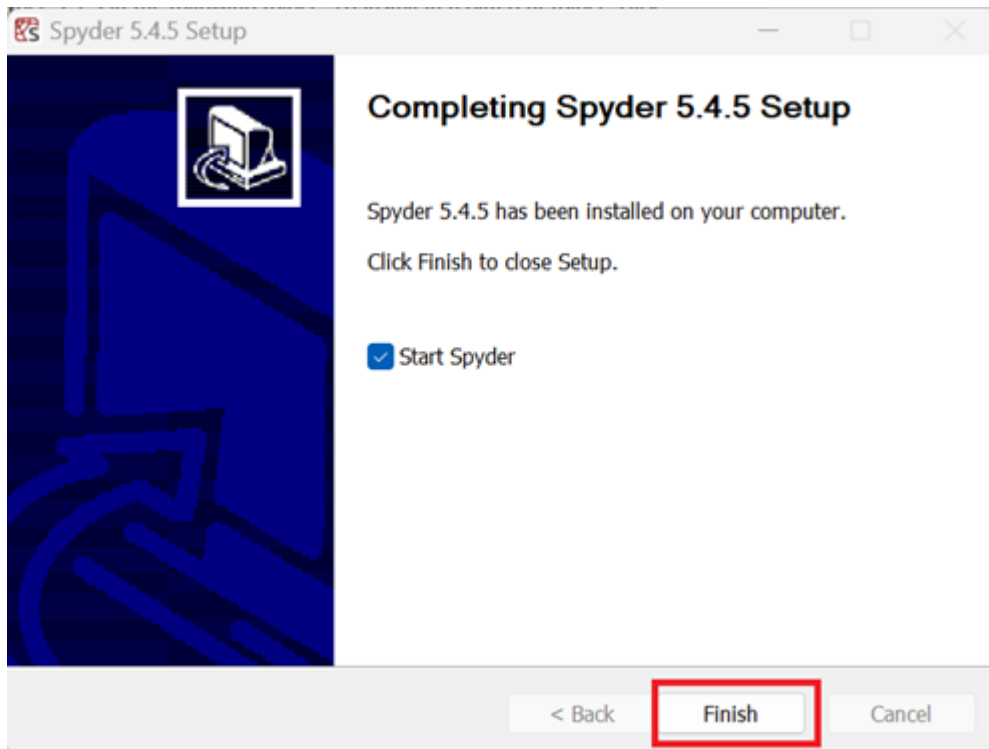
Now in the next window you have to select **"Install for anyone using this computer"** and click **"Next"**



Now click **"Install"** and wait for the installation process to complete. It will take a few minutes.



After the installation is complete you will see the final window something like this:



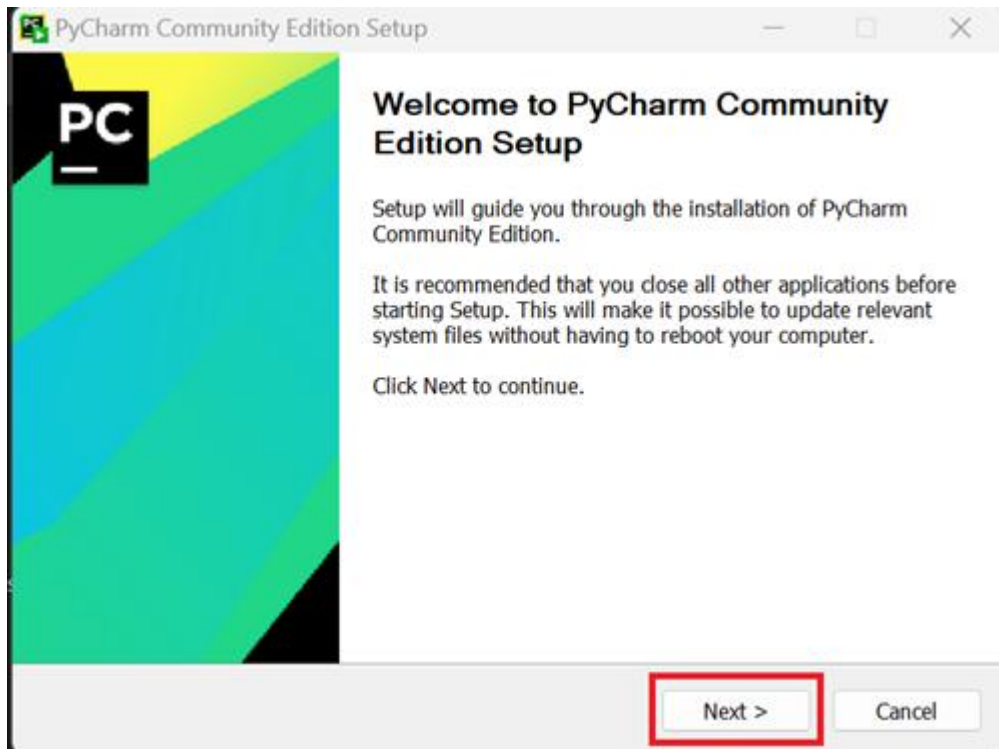
- Now you can check or uncheck the “Start Spyder” option (**checking this will launch Spyder IDE**) and click on **Finish**.

Other Methods

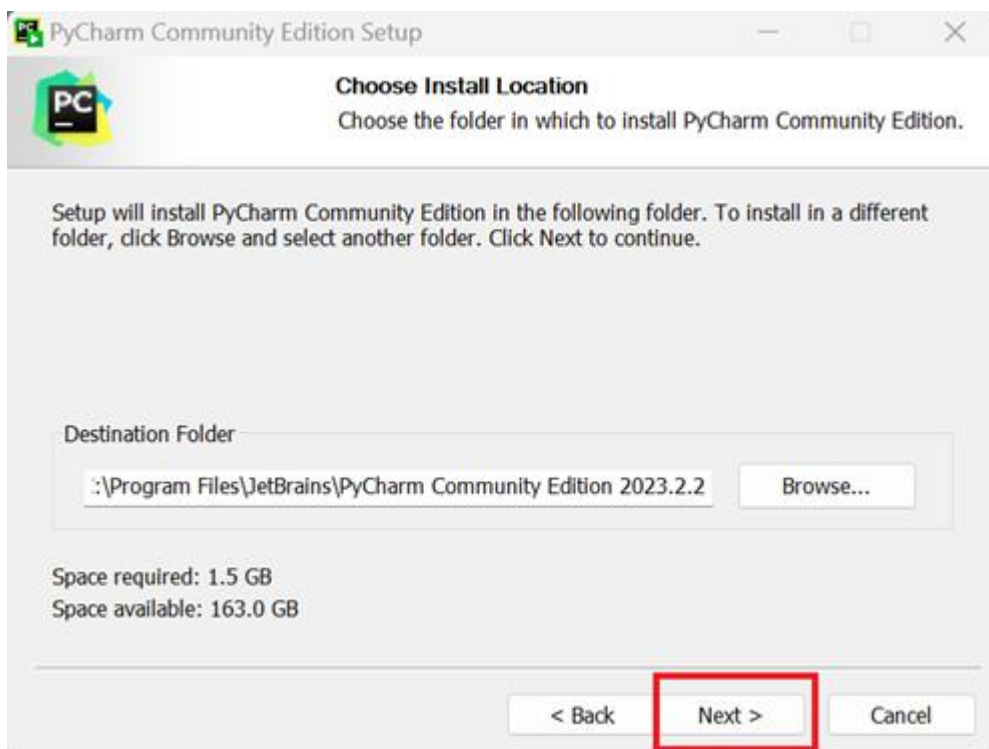
PyCharm:

Now, if you do not want to use the Spyder IDE or want to try some other options. You can also install PyCharm. **For using PyCharm you need to have python installed in your computer which you can do by following the python installation instructions mentioned earlier.** To install PyCharm go through the following steps:

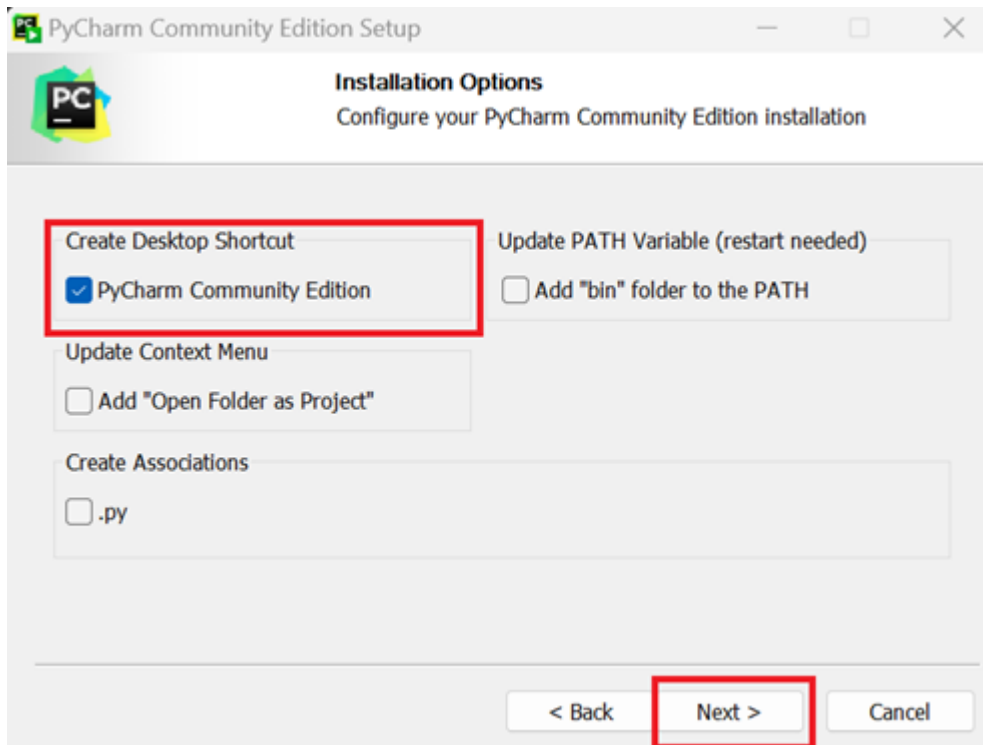
- To download PyCharm visit <https://www.jetbrains.com/pycharm/download/> and scroll down to PyCharm Community Edition and click on the “**Download**” button and wait for the download to finish.
- After the download has finished, we need to run the file (**double click on the downloaded file and allow**). On doing so we will see a installation window something like this:



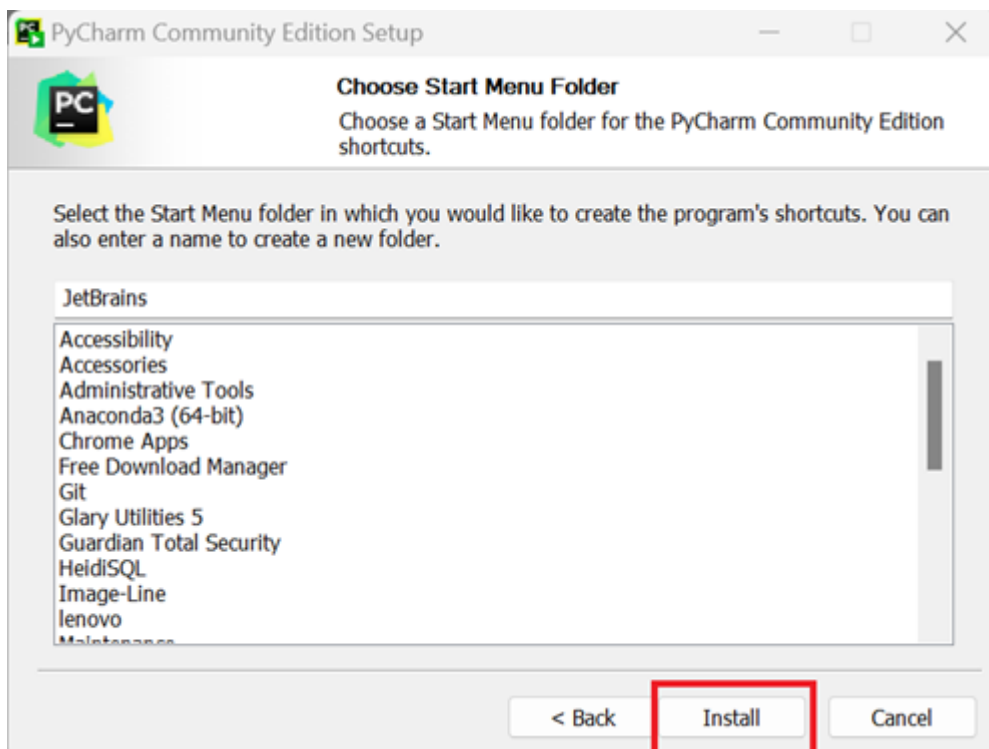
- Click **Next** to proceed with the installation.



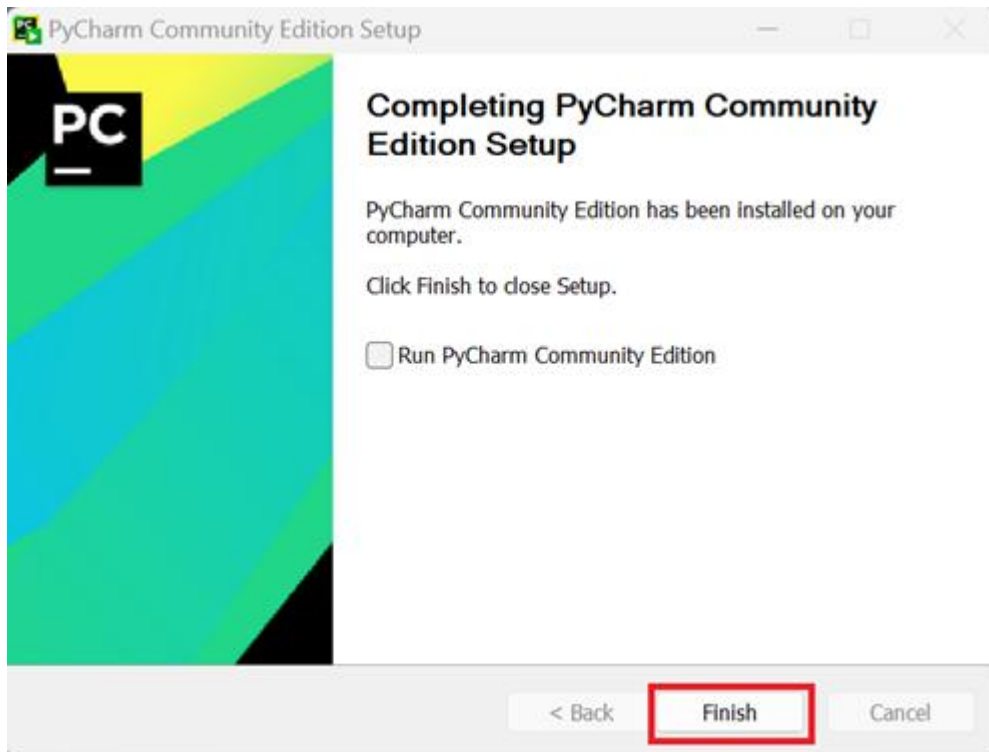
Again we have to click **Next** to proceed.



In this window we need to check the “Create Desktop Shortcut” checkbox and click **Next** to proceed with the installation.



Now click **Install** and wait for the installation process to complete.



- Now click **Finish** and that is it. You have successfully installed PyCharm IDE on your computer. You can check for PyCharm icon on your desktop.

VS Code:

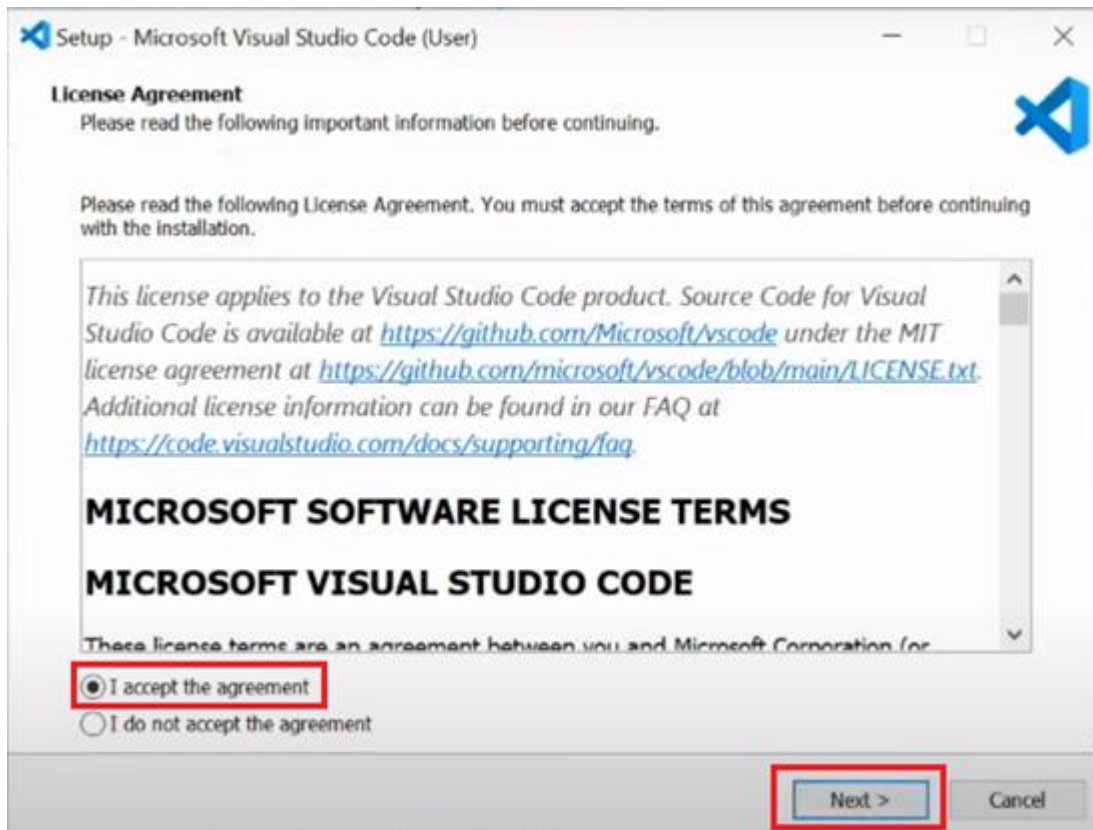
Visual Studio Code, also commonly called VS Code, is a source-code editor made by Microsoft. This is also a preferred editor amongst the coders because of its extensive addon extensions that make coding easy and fun.

To use VS Code, you need to have Python installed on your computer, which you can do by following the python installation instructions mentioned earlier.

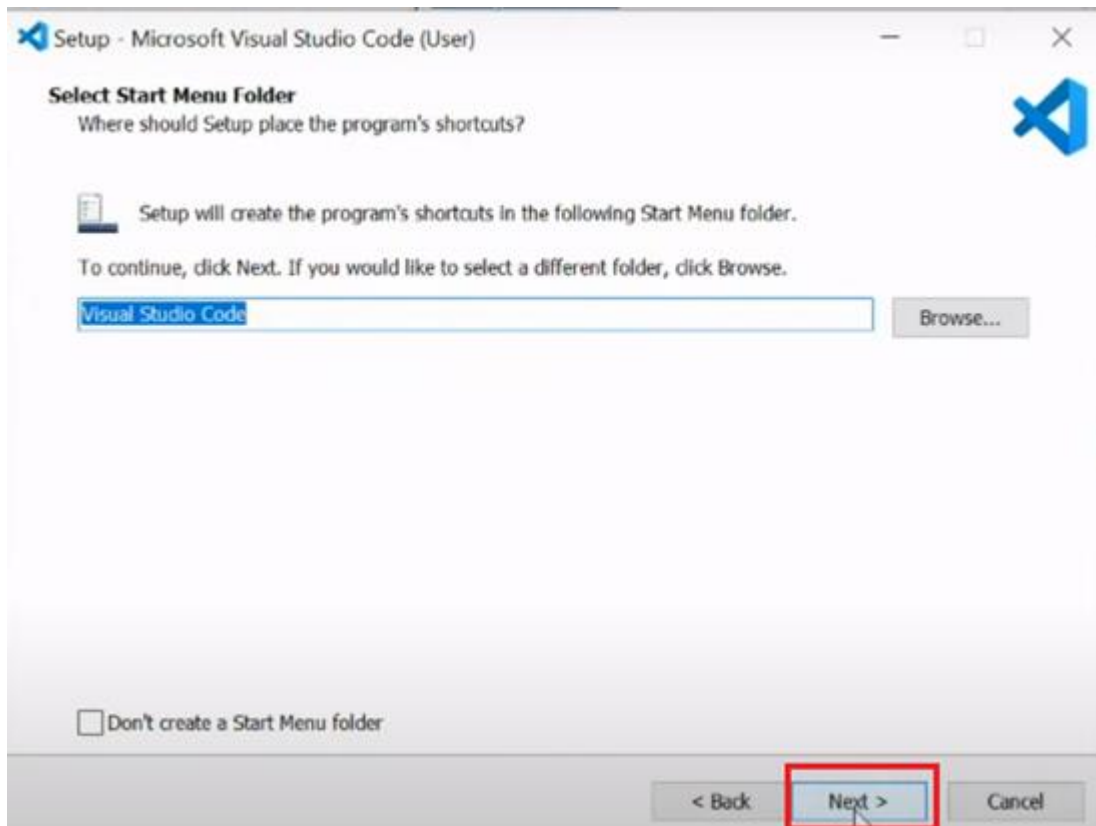
Now let us install VS Code:

- First, we will download VS Code by visiting <https://code.visualstudio.com/> and clicking the **Download** button and wait for the download to finish.
- Once the download is complete, we need to run the downloaded file which we can do by **double-clicking on the**

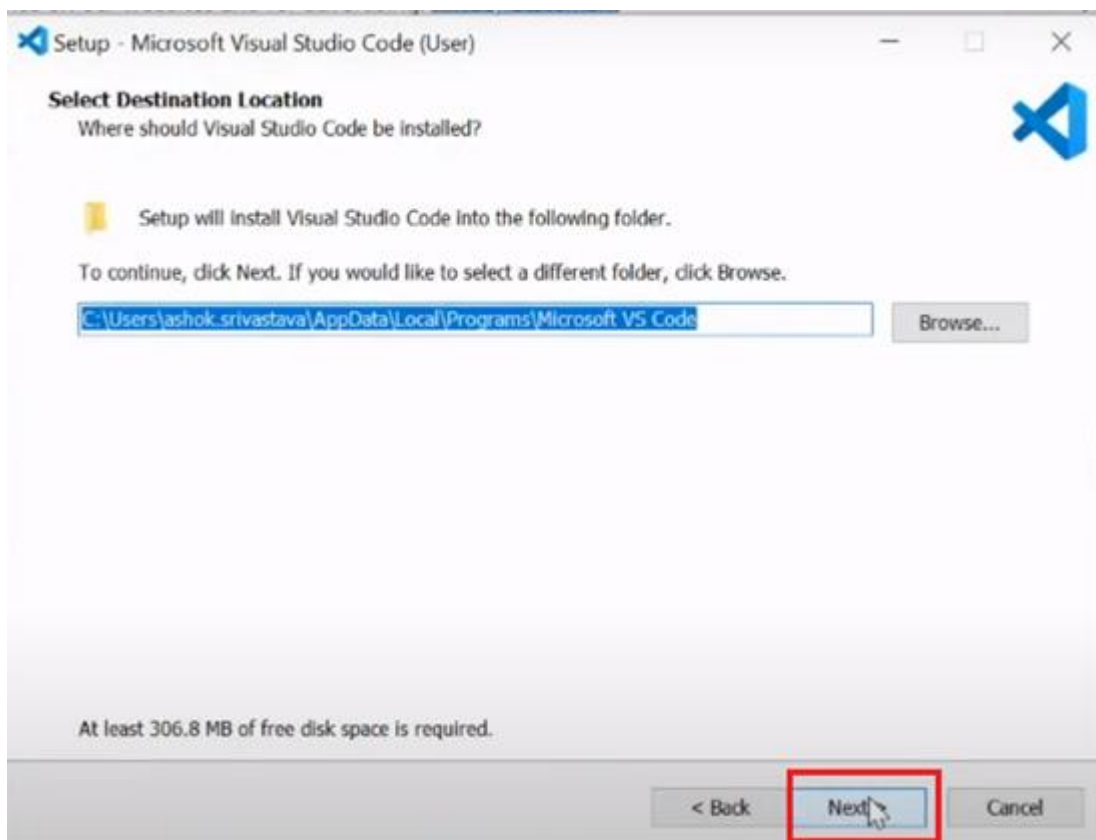
downloaded file and allow . After doing so, you will see an installation window something like this:



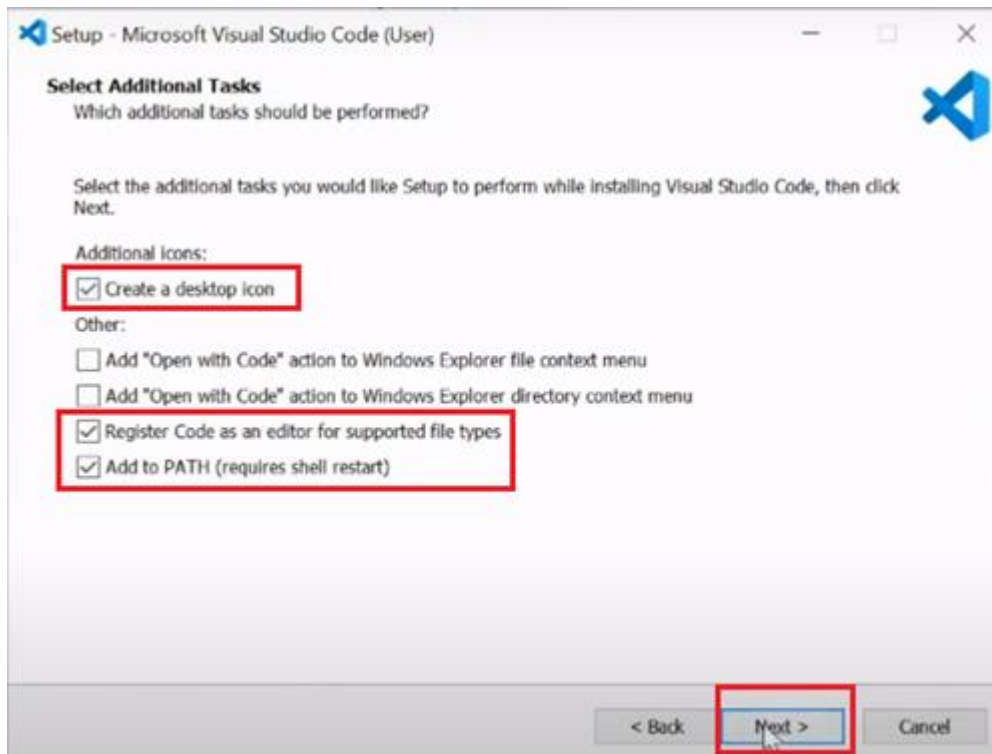
In this window, read the license agreement if you want and select “ / accept the agreement ” and click **Next** to move forward.



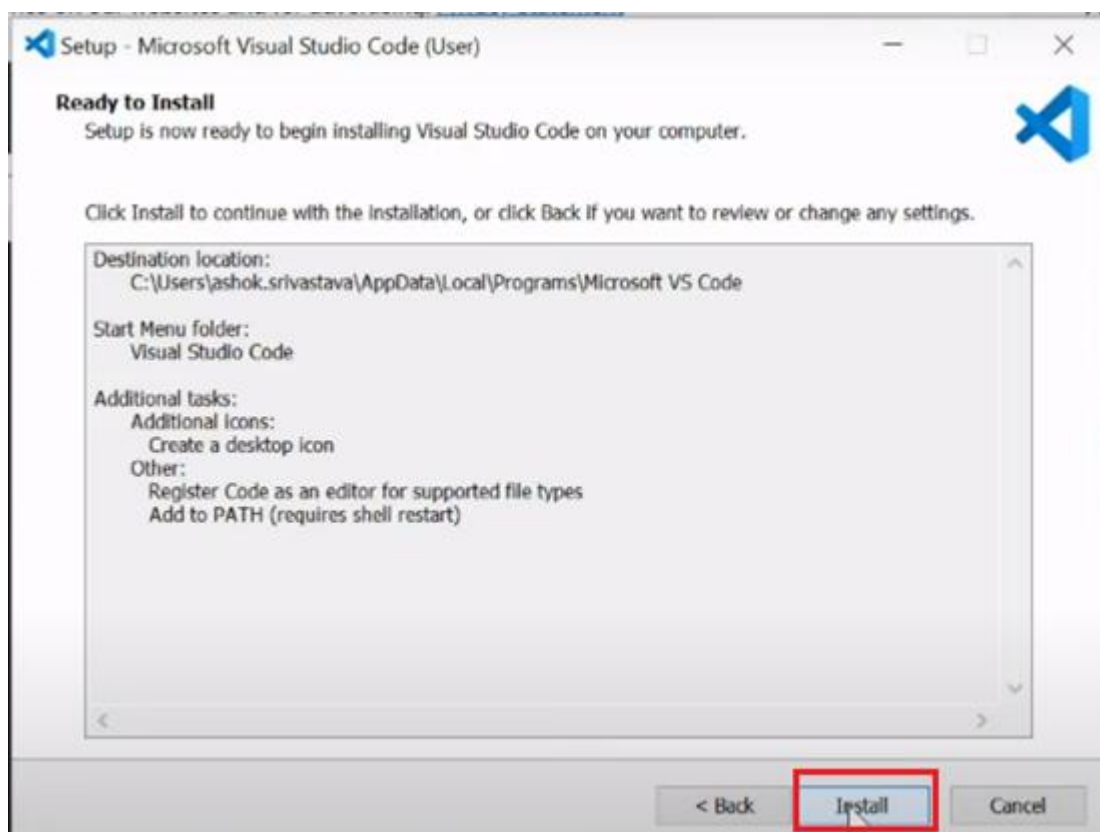
In this window, we do not need to change anything. Click **Next** to move further.



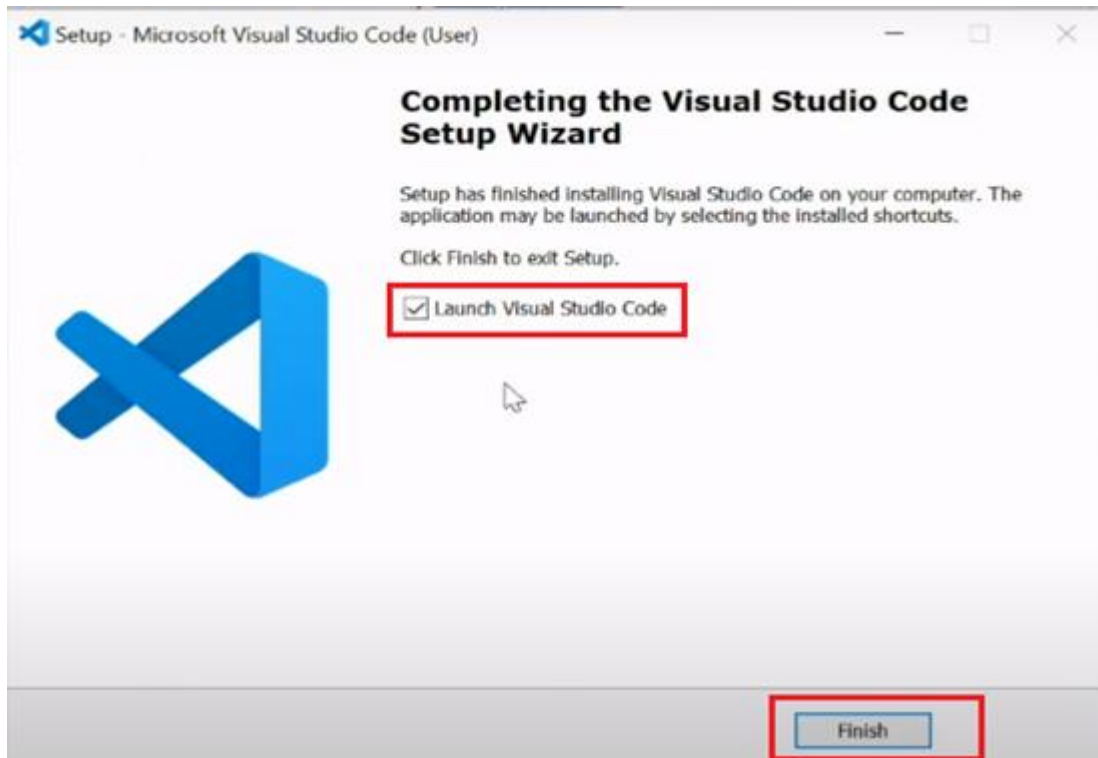
Now again we have click **Next** and move forward to the next window.



Now in this window, we need to **check** the “*create a desktop icon*”, “*Register code as an editor for supported file types*” and “*add to PATH*” options. **It is important to select these options** and then we have to click **Next** to move forward.



- Now we must click Install, and wait for the installation process to complete.
- After the completion you will see something like this:



- Now the VS code has been installed successfully. Here you can check or uncheck the “Launch Visual Studio Code” option, as per choice, and click Finish. Selecting the *Launch Visual Studio Code* option will launch the VS Code.

Now that we have both Python and VS code installed on our computer, we can use them to code in Python.