Advanced Business Decision Support Systems

Professor Deepu Philip
Department of Industrial Engineering and Management Engineering
Indian Institute of Technology, Kanpur
Professor Amandeep Singh
Imagineering Laboratory
Dr. Prabal Pratap Singh
Indian Institute of Technology, Kanpur

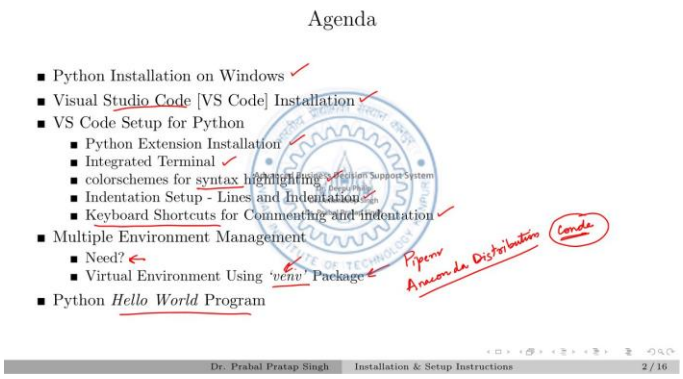
Lecture 27

Python Development Environment Setup

Hello everyone, I welcome you all to the Advanced Business Decision Support Systems lecture series and today, we are going to set up our development environment using Python and a code editor.

So, until now, we have covered the basics of what is a Programming language, the Elements of Programming language, various Classifications. We have also discussed the tools: Compiler and Interpreters and how they differ from each other and finally, we have also seen that Python is an Interpreted Programming language with dynamic typing.

So, to further get the hands on implementation details of the Business Decision Support Systems, we need to start developing our code base. So, to do that, we need to first get ready with the development environment on our machines.



So, today, let us start by installing the Python. So, usually, there are three most popular operating systems like Linux, Windows and MacOS. So, we are doing this installation instructions on windows only, but the instructions will only slightly differ for Linux and MacOS. Usually, Linux and MacOS comes pre-installed with Python as a system library but windows does not provide this language as a pre-installed. So, let us first see, how to install Python on Windows, then we will install another software on windows which is Visual Studio Code.

This is also from Microsoft, then we will set up that code editor, this is a code editor. So, we will set up that using different extensions available. The most popular extension which is officially supported by Visual Studio Code team is Python and then we will look how to use the Integrated Terminal available in this code editor. After that, we will try to set up the coding environment by changing or getting to know how to change the color schemes for Syntax highlighting. We have already learnt what is syntax in a Programming language.

After that, we will see, how to set up the Indentation. We need to stay consistent with the indentation setup because python follows white space indentation as we have seen in the last lecture. After that, we will also see the some of the most used keyboard shortcuts while developing a code in a python environment. So, we need to set up the keyboard shortcuts in our code editor. After that, we will look into one of the most discussed topic in the Python development environment is the development of a virtual environment.

So, we will look into it what is the need of creating a virtual environment with for each project and then, we will see, how to do it. So, here we will use the VN package which comes pre-installed with the as a standard library in the python installation, but there are many different versions or different packages that can perform the same task differently like Pipens, etcetera. Also, in the Data Science community, we usually use Anaconda distribution. This Anaconda distribution provides a different package manager known as 'Conda' which can also perform the environment management task for python packages, but we will look into this conda environment after some time, if we build those Data

Science packages using that Anaconda distribution. Today, we will only stick to the VN library.

After that, we will create our first hello world program which is the classic program which will print 'hello world' on your terminal.

Figure: Initializing Python Installer

So, let us first go to this link (https://www.python.org/downloads/) and download the python package from the internet. So, this is the python official website and on this, we need to click this download button. Here, you will see different kinds of python versions like 3.12 which is a pre-release. We only need stable versions today. So, let us just click on this button download python 3.11.5. So, this will download and get stored in your downloads folder.

So, go to the directory downloads and here we can double click this file. Now, you will see this new window. So, here there are two main options Install now or customize this installation. So, for beginner, we just click on Install now, but do not forget to check this these check boxes add python .exe to path.

If you use admin privileges when installing pi.exe, then it will install system wide. So, you can tick this as well if you are the admin of your system. Now, the setup is completed for basic Python installation on your windows. So, you can close this and after that, to check whether it is installed or not, you can click on windows, right click on windows and click on run.

So, here you can type this 'cmd' command to open a Command Prompt and here you get this Command Prompt window. Now, to check whether there is python or not, you can write Python and it will open up. So, it is this is the interpreter.

So, here your Interpreter is saying, this is python version 3.11.5 which we have just downloaded and installed and after that, here he is the interpreter is waiting for our commands. So, you can write like, in the last lecture, we have seen the 'Zen of Python'.

So, you can write here import this. So, this is the 'Zen of Python' like beautiful is better than ugly which we have talked in the last simple is better than complex. So, now our Python is installed.

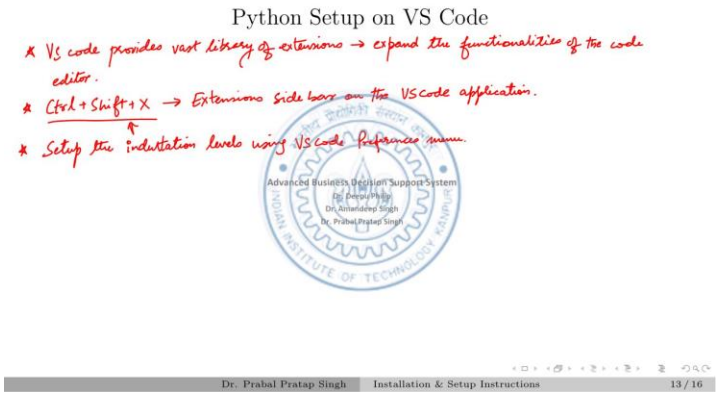
The next thing we need to do is, to get a code editor and we will see, why do we need that code editor after installing it. So, open Google, write 'Visual Studio Code', click on the link as code dot visual studio dot com and from here, download the stable build. It is in the subtitle, it is showing this is the stable build. So, download for Windows. So, we are using windows 11 here in this.

If you are using windows 7 or 10, it will work. So, this is also downloaded now, you can double click this file. The first window asks for you to accept an agreement with the Microsoft. So, you can click on 'I accept the agreement' and click on next. After that, check this address, this is the directory where the VS code will be installed.

So, let us just keep it to default next, it will ask for a start menu folder. So, start menu folders are these folders like end note here is a start menu folder. So, it will create a folder here for visual studio code or you can use this check box to not create a start menu folder. After that, you can use these check boxes also like, add open with code action to Windows Explorer file context menu. So, what will happen if you use your file explorer.

So, on right click this is the context menu. So, here if you click on a file, it can show the Visual Studio code here as well. So, for that, you can use this option and similarly this option as well. So, the first one is for the file context menu, the other is for directory context menu. So, if you want to open a complete directory in VS code then you can use the second option.

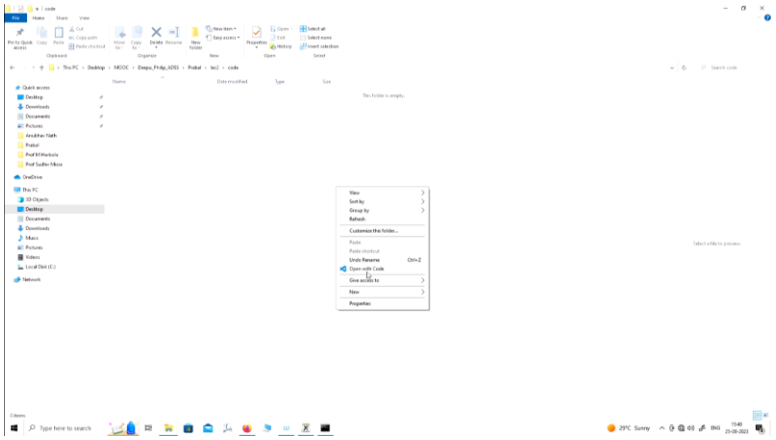
After that, you can also create a desktop icon. Now, the installer will summarize all those things that you have selected and then click install. So, you can launch visual studio code now. This is the first window which you will see, it will ask you how would you like to see your visual studio code. So, these are some visual customizations, you can do like this is dark modern in which, it opens by default, you can use this theme as well and these are some options that you can check and do things, but let us just check to the default. Now, our two main programs are installed right. So, what can we do is.



So, let us first see, what we need to do on visual studio code. So, for as a short term, we will use the word VS code from now on. So, VS code provides vast library of extensions.

These things can expand the functionality of the code editor. So, the very basic shortcut command, you can use is control + shift + X, this will open the extensions sidebar on the VS code application. The beautiful thing about this VS code editor is you can customize anything as per your liking. So, you can customize the shortcut command as well, but by default, this is the command. So, we can also set up the indentation levels using VS code preferences menu.

So, let us get a deep dive of VS code now. So, first we will create a directory desktop mocks, lecture 2, new folder code.



So, here you can see 'open with code'. So, this is the feature which we have selected in the installer menu. Now, it will ask you, whether you trust the authors of the files in this directory or not.

So, you can click yes, I trust authors. Now, you can create a new file from here or open a file if there is a existing file. So, let us first get to know, what are different things in this

editor. So, on the first row of the editor, it is a menu selection row. So, you can use the file option to either create a new file or open an existing file or add a different folder to this workspace.

Otherwise, you can use these edit commands, these are great out right now, but you can see, they are working when a file is open. Similarly, this is a selection menu, you can change you can change your appearance, you can change the layout of the editor by splitting this editor into half or right or you can even get three columns. Terminal is something like, you have already opened the command prompt using Windows + R shortcut key and then writing command, but here you can just use this control shift, this is tilde command to open the terminal. So, let us see control shift. So, this is the terminal which is open now, you can write here Python.

So, this is the same thing which we have earlier seen. Now, we can directly open this Python interpreter in inside the Visual Studio Code. Next is known as the Activity bar. So, here this is the explorer which is by default open now, it is showing that there are no files existing in this directory. So, if you want to create a new file, you can right click here and create a new file.

So, I am giving a name first dot this is the extension of Python (.py). So, if you just enter, this is the first Python file, now it has already opened in the right side of the Visual Studio Code, you can see here, we can close this welcome tab. So, here you can open multiple files here and each file will get its space in this tab bar. So, let us create a different file with different extension like read me dot MD.

So, this is a different kind of file. MD stands for Markdown, it is nothing but different kind of markup language and it can use to decorate your documents easily. So, we can also create a different file here like text file (.txt). So, all these files which we are creating, they are directly getting opened in the Visual Studio code and these files are being created in the same directory from where, we have opened this Visual Studio Code. So, you can see here, in this code directory, these are the three files, we can see which we have created first .py is a Python file the read me.

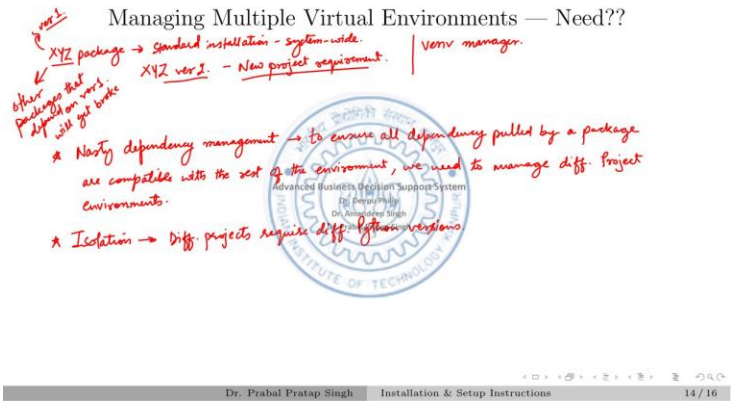
So, this is a Python file first, .py this is read me, this is a Markdown file and this is a text file. These three files we have created. Now, open the Visual Studio Code again. So, let us close these two files and this is coming back to the activity bar. So, this is the search bar, you can search anything and replace it with something else, if you have a large code base in your file, then you can do a simple search and replace option from here also using click endpoint options otherwise, you have shortcut keys for those as well.

Next, this is the source control window, we will touch upon this options as well in the later part of the lecture series. This is the debugger and you can use a debugger inside the

VS code as well and we will see, the how it works in the final decision support system in the complete decision support system which we will create. The last is the Extension. So, you can see that, here it is showing in brackets, control + shift +X.

So, this is the default keyword shortcut. So, you can either click on it and it will open otherwise, you can be here, open this Explorer sidebar and now I am using the shortcut key control + shift +X. So, it will open directly here. Now, either you can search Python here and it will search for all the extensions with the name Python in it and this is the first official release from Microsoft for this programming language. You can click on this install button to install this extension inside this VS code.

So, it will say, that it is installing. So, our extension has been installed. Now, how to check is let us just see. So, it is showing three different options like, switch to pre-release version by default, it will install the stable release and the first option is disable, you can either enable or disable this extension from here only otherwise, you can also uninstall it if you are not going to develop using Python in VS code, you can use many different programming languages inside VS code by using their particular extension. So, we have now installed this extension.



So, now since, we have almost installed our development environment, now we need to start coding inside the development environment.

So, before that, we first need to know few things about Python which is kind of controversial because it needs manual installation efforts. So, for that, we know that I have discussed earlier that some of the operating system usually comes pre-built with Python like Linux and MacOS and now we have already installed Python in our system. So, what happens is Python comes with some standard libraries and its packages. So, let us say if you install other third-party package, you have two options, either you install it system-wide or you can install it in a different environment by creating a secluded space in your developing machine and then install that package.

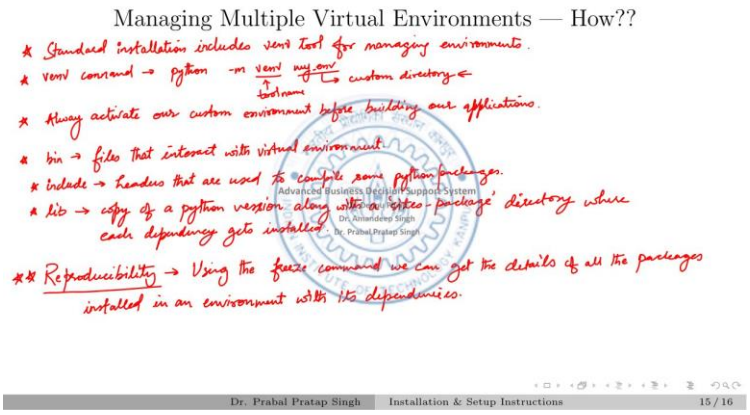
So, that we can isolate it. So, let us understand it with an example that if we all have visited a library in our life, so let's say, if librarian keeps on buying the books but do not arrange it in a well manner. So, what happens is, you visit the library and you will try to find a particular book in the library and if you ask the librarian, he or she will ask you to search for it because the library is still completely disorganized. So, it will be very hard for you to find a particular book and the particular edition will be very hard to find. So, even if you find a book, it will be very hard to get a particular edition like, version, edition 3 or edition 4 of that book.

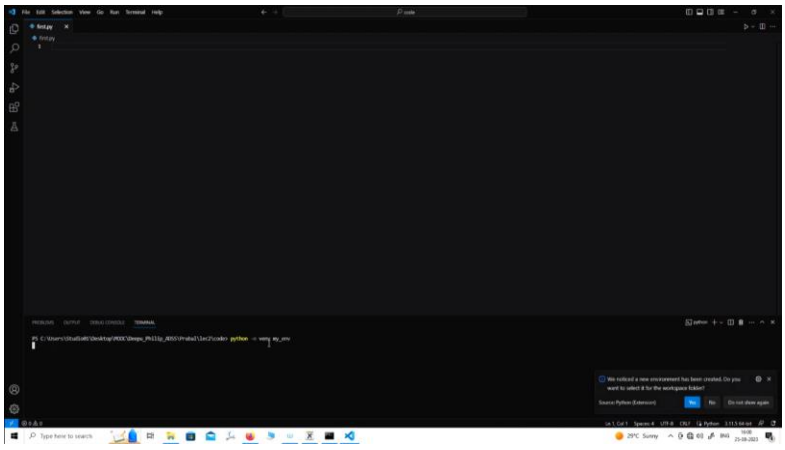
The same goes with the Python as well. So, what happens is, the standard installation of Python has some packages let us say, XYZ package. This package is available in standard installation which is system wide. So, when you have installed XYZ package, this would have come with its own version, let us say ABC. Now, after some few days, you start creating a different project and for that project, you again need this XYZ package but an advanced version of XYZ package which has been updated in the library by its developer. So, what you will do is, you will install a new XYZ package with version DEF, this is just term this will be usually in number.

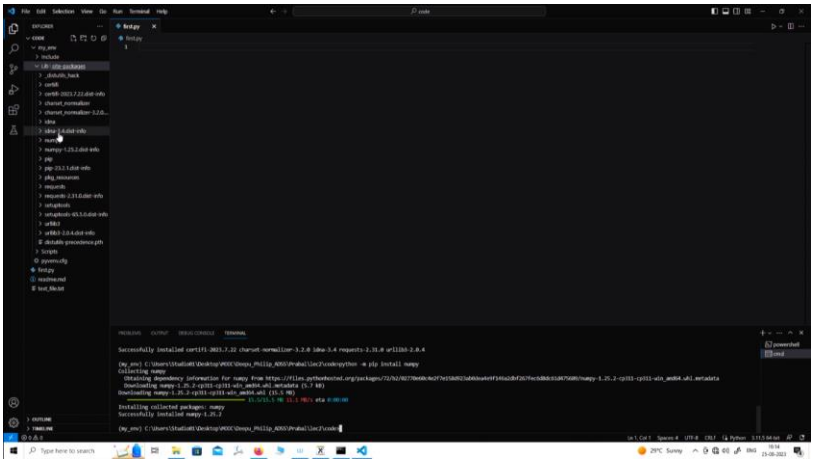
So, let us go to version 1. So, this you have installed in version 1, then say, you installed version 2 of this for the new project requirement. Now, your new project will work but the problem stays with the XYZ package version 1 because it has been overwritten in the standard installation. Why? Because you have installed this version 2. So, all those other packages that depend on version 1 will get broke.

So, here is the main problem. So, what happens is developers of this language tried to create a solution for this. So, what they suggest is always when you are trying to create a new project, you start with a secluded environment in your developing machine. So, for this we will use Vn manager. So, summarizing all these things, Python has a very nasty dependency management. So, to ensure all dependency pulled by a package are compatible with the rest of the environment, we need to manage different project environments.

So, this is the property which we are talking about is isolation. We need isolation of various packages that are being installed because different projects require different Python versions.







So, next is, how we are going to manage these multiple virtual environments. So, we can use the Vn command on the terminal which is Python 3m Vn myn. So, here this is the name of the tool and this is our custom directory or the environment name that we are defining.

So, user can define any name here at myn. So, these are the options of these command Vn and myn. So, we will always activate our custom environment before building our applications. So, let us create first custom environment. So, we will open our Visual Studio Code and here, let us first exit from our interpreter by writing the command exit and open and close brackets.

Now, you are again into your directory. So, you can use cls command to clear. So, let us write the same command which we have discussed in the terminal of the VS code here. So, this is Python. Let me click it again.

This is python -m vn my_env and it will create your new Python environment. So, what this command has performed is, let us open the explorer from here and we can see, there is new directory my_env in the same directory which from which we have opened this Visual Studio code. So, let us look into the directory. So, this is my_env. Now, here are three different folders and we will discuss about these three folders, what are the contents of these three folders.

So, here this scripts folder is very important. These activate files and deactivate files will be useful for activating or deactivating your environment. So, let us go again to Visual Studio code. Now, here we have just created the environment, but we have not yet activated the environment. So, to do that, you can write here.

So, to activate it, write this command on your terminal. So, let us activate our environment. Write- \my_env\Scripts\activate. And, here you can see that, our environment is activated. Now, write here Python. So, it is showing three different addresses. The first one is the current directories address, the next is the system wide location of the Python.

So, now we can see that, we have different pythons in different environments. So, let us clear the screen. Now, to install a different package let us say, a request package very widely used. You can write Python m pip install. So, it will collect the request package from the internet and install it in our environment which is my underscore env.

Now, you can also open this my env directory, open the lib sub directory in the sites package, you can see request. So, it was not earlier here. So, let us install one more new package and see, whether it is installing here only or somewhere else. So, you can first see that, we are going to install num p from the internet, but here no sub directory has been named as num p numpy.

So, let us install it. You can use the previous command by using the up arrow and we need to just change the name of the new package num p, press enter. So, it is now downloading and now installing the num p package. Now, it is showing that, it has been successfully installed and you can see here that, the num p directory has been created. So,

this is, how you are going to install or the required environments in your custom build environment. And, you have the option to keep this directory inside your project or keep in a single location where all your project environments reside.

So, these are two different kinds of options you have. So, let us now see, what this code editor allows us to do. So, first of all, we can open command pellet using control shift and p command. So, this is the command pellet, you can write most of the commands to search and directly open it. So, first of all, we will be opening up the color schemes.

These are the different color schemes. Now, just changing using the arrow keys on your keyboard, you can see that, the complete colors of these of your code environment will change. Now, why this is useful is because this will change the colors of the syntax while you are writing the code. So, syntax highlighting is a very important property that is being provided by these code editors. So, see these are the different options, we will stick to Monokai for our work.

And, we can also see the various command shortcuts by writing keyboard shortcuts. So, these are all the shortcuts that are defined by default in the VS code. You can search here for opening extensions. So, this is the extension, show extensions this is control + shift + X. Now, you can change this by double clicking it.

It will ask for press the desired key combination and that is enter. So, you can change it here as well. So, to check for any kind of keyboard shortcut, you can come here. So, let us discuss about the three different sub directories created by the environment command. So, these were the bin directory which holds the files that interact with virtual environment.

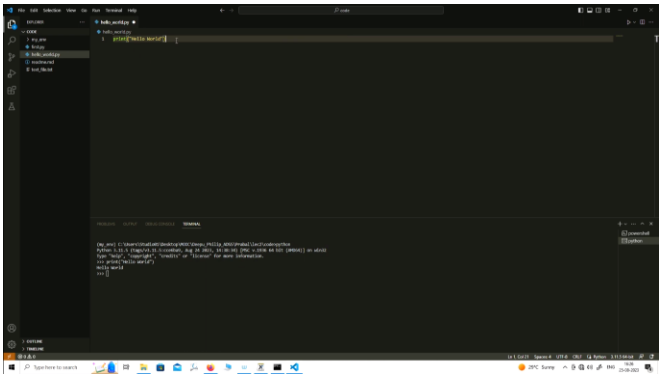
The next sub directory was include. So, it stores three headers that are used to compile some python packages. The last sub directory was lib. So, it holds a copy of a python version along with a 'sites-package' directory where each dependency gets installed So, by using these virtual environments, we get this important property of reproducibility. Using the freeze command, we can get the details of all the packages in the packages installed in a environment with its dependencies. So, what it is saying is that, let us say, you have created a project and you are using custom built python virtual environment for that project.

And, now you have completed your project and you want to share it with your team. So, what happens is, your team does not know which version of the numpy or request package you have installed in the environment. So, to ease out this process, you can either share your complete project directory which can get very large or you can just tell the team members that, this was the actual version number which I have used in my project environment. So, they will only use that file which stores all these version

numbers and they will install those packages. So, that they can get the exactly same python virtual environment for their use to get to see or review your code. So, all this process can be automatized by using the freeze command available in the python environment.

So, let us see how it happens. So, we can use this command python m pip freeze. So, it is showing that these are the different things that we have installed like, we have installed the request library here and it was having the version of 2.31.0 and numpy was also installed, which was 1.25.2. So, you can use these details and store it in a different requirements file and share this with your team. So, that they can install all these packages.

So, let us now create our first hello world program. So, there are two ways to code in python, either you can write everything in the interpreter mode by calling this command python on your terminal and it will show these three arrows and is asking for the next command. So, you can write any command here and it will execute by first checking the Syntax and Semantics of that command and then will show the output.



So, let us write the very basic hello world command by using the print function which will print the string which we provide to the terminal.

$$print(\text{"Hello world"})$$

So, by pressing enter, it is now showing that, Hello World is being printed on the terminal. So, this is the most basic example of a command on terminal, the same thing you can do by creating a script file. So, let us now create the first script file hello world as hello_world.py. Now, just write the same command here, print hello space world. So, we have written our first command and the tab bar is showing, this circle filled circle here, which is showing that, we have modified this script, but have not yet saved it. So,

to save this file, we can press control + S on our keyboard and now it is showing the crossbar.

It is indicating that, we have saved our file. Now, this file is also available in our explorer and is also available in the directory hello world. So, to run this file, you can first need to get out of the interpreter by writing exit, open and close brackets and here you can write python hello. So, I use the tab command to auto complete the name of the file python hello_world.py.

By pressing enter, it is showing Hello World. So, these are the two different ways of using of writing the code in python. Now, you may ask, why do we have two different ways? So, what happens is, when you are testing and developing your code line by line, you can use your interpreter by using this Python interpreter available in terminal you can write some short codes and test it whether it will work or not and then go to your script file and create a large code base here.

So, scripts are usually for large codes and you the interpreter will then, interpret each code line by line. So, also you can see here that, this is the function, built in function print and this is the string which we wanted to print on our terminal. So, we can do all of these stuff all of creating our scripts in a basic notepad file as well like, I have created this text file dot txt here. So, let us just create a new file here document and give this a file name new code.

Now, this is a text file, if you change rename this file, it is not showing. If it is not showing the extensions in your explorer, what you can do is, you can click on view then, go to options, change folder and search options go, click on view tab and uncheck this bar, hide extensions for known file text type. So, now you can see the complete file name right now, rename this file and change the extension of the file by changing it to p file.

So, now this will ask you if you change a file name extension then file become unusual, but it will be usable, we have just changed the extension name. Now, it is using the python extension. So, if you try to open it in notepad, now we have opened the same new file, new code .py in notepad and just write the same command, print hello space world space from space notepad, use control S or the file menu to save this file.

So, we can see that, we have created a new file using notepad and if you try to run this file in the same environment by using the command python new_code.py. So, now we are running new_code.py from our terminal inside VS code, but you can also use this command frontier which we used earlier. So, from anywhere, you can just start a python environment and run this file by pressing enter and now it is doing the same thing hello world from notepad. So, we created this file from the notepad and it is running as usual.

So, now why install a new software like VS code, then configure it to your liking and then start coding. The reason is that, all these color things, this print is in green color then, hello world string is in yellow color and the brackets is in dark yellow or some light brown color. So, these are the syntax highlighting colors and this will be very useful when you have thousands of lines of code and you want to understand by just looking at the code, what this is.

So, this is differentiating between print and hello world as saying that, print is a built in command that is known by the interpreter, but hello world is just a simple string which you can edit as per your liking. So, with time, as we increase, the lines in our code base, we will get to know the functionalities of visual studio code better and how it helps us in more than one ways and why we should not keep on writing our code in notepads.

So, thank you all and we will meet in the next lecture where, we will understand basic statements and variables available in python and further things that we need to learn to get the understandings of how python code works and how we need to create much better and improved code to create a complete decision support systems. Thank you.