

# Day 7 Let's Begin Python

## Summary

- The speaker is introducing Python, a programming language that is similar to Visual Basic, but with a different syntax. The speaker is demonstrating how to use the command line to navigate files and folders, create Python scripts, and run them in the terminal. The speaker is emphasizing the importance of using the command line, and is demonstrating how to install and use Anaconda, a terminal environment that makes it easier to work with Python.
- The speaker is explaining how to use the command line interface (CLI) in GI Bash, a Linux-based operating system. The speaker is demonstrating how to use the CLI to navigate to the desktop, create new files, and delete files. The speaker is explaining how to use the CLI in Mac, and is demonstrating how to open a file in a preferred program.
- Speaker 1 explains how to use the terminal, including how to create folders, write commands, and run programs. She recommends using Visual Studio Code, and says that it is easy to use. She explains that the terminal is a useful tool for debugging and troubleshooting.
- The activity guys? Perfect guys. So let's, let's work in this. We have 10 minutes to work in this part and then we can, we can review it. Under, I'm in a, in a editor with quick Python and have a print statement, but there's just icons and I have no idea what any of them do. I thought we were, you were gonna go over how to

install it. It says on instructions as well. You just went over the commands. I've got some weird graphics at the top of the screen that I have no idea. Adjust editor options. I must be in a totally different editor. Are we working in exercise number one or two? You just go. You're muted by the way, f dear. The wrong command is Python. I actually do Python three quick Python, BYN Mac and it works. Try to do that. Alan Python three. I am Python three. The wrong command is Python. The wrong command is Python.

- Speakers 1, 8, and 11 discuss an issue with Anaconda, and Speaker 11 offers to create breakout rooms to help. It is recommended to work in virtual environments, because they will not affect anything on your local machine. To create a virtual environment, you can use the `conda create` command, and name it anything you want.

Summarized by AI | June 30, 2023



Select text to create highlights [Learn more](#)

## Transcript



S1 Speaker 1 ▶ 0 00

We can Thank you Andrew. And so for this part guys, we are not going to need the virtual environment. I am going to explain to you the main

purpose of using virtual environment so you actually can understand more about what it is and why we are gonna need it. Okay? So let me start sharing my screen right now.

S1 Speaker 1 ▶ 0 38

Okay. I am going to go to the presentation note and please let me know if you can see my screen with a thumbs up. Perfect. Okay guys, let's start. So today's introduction to Python. Python. That is basically a programming language that the most significant change from Visual Basic, it'll be the syntax. Okay? Because the fundamental context about coding, about the process flow and about object programming, like subjective against real context, remember about that part is basically the same. Okay? So class objectives, we are gonna be able to perform Python three installation. We're gonna navigate through folders and files via the terminal. We are gonna create Python scripts and run them in the terminal. And we're gonna stand, we're gonna understand like basic programming concepts in Python. So today, basically we are gonna cover variables, arrays for loops. We're gonna type our first type program. But yeah, today it's gonna be more about getting to know to your terminal because it's important on how you, you navigate through that.

S1 Speaker 1 ▶ 2 07

Okay? So few things before we start on Python, it's a traditional programming language. The difference between, like I mentioned from BBA and Python is the syntax. So terminal, in terminal, I want to show you this slide before I start working on it. Basically in this slide you will find this commands that actually I share already into the resources in A T X T file, okay? Due to the Slack channel resources, you will find the T X T file to this information. Basically CD is to change the Atory CD and the symbol is changed to the home director and it goes like that. Okay? So I am going to exit this because I am, I want to present my terminal and I want to keep, keep going on this before I present my terminal guys, we have to terminals here. We have Max and we have Windows.

S1 Speaker 1 ▶ 3 18

So basically for Windows you can, if you have Windows 10 like I do, you can go here and just click CMD and you open, you come and prompt. Okay? For Mac, if you go to the Finder and then you click terminal, I mean you type terminal, you're gonna be able to access your terminal. Okay? So for this part, to be honest, right now in my computer, because this computer, I use it to work sometimes and I use it sometimes as well to, for this course, I have everything set up okay, in your computers, maybe you will not have everything set up and you're gonna see something different, but that doesn't mean that it's not working. Okay? That's one of the points that I want to tell. So basically, if you install an Accon correctly, and if you have a Windows or a Mac, when you go to the Finder and click Anaconda, you will see an application like this one.

s1 Speaker 1 ▶ 4 29

Okay? When you open an Accon, and I jumped to Anaconda because I'm going to explain about the terminals, okay? When you open Anaconda, you're gonna see the application, the the application hub where you can actually install Visual Studio Code Spider, you can install Jupyter Lab, all these applications and you can launch them that they work purify when you are like doing the science or data analytics. So just wanted to show you this, oh, sorry guys, are you seeing my screen? Yes, yes. Perfect. Okay. So in my windows, that's why it's a little bit confusing. I have three terminals. I have C, M, D, okay? I mean cmd, that looks like this common prompt of windows I have GI B, GI Bash is the one that I'm gonna be using a lot, okay? Because GI B I can connect GI to my Visual Studio code and to my GitHub as well. So I can clone repositories, I can pull requests. So that's why I'm using GI Bash in Windows. Hugo, can I use the terminal? Yes, you have to set up the terminal with Bash in order to clone a repository, for example. And this one is embedded, you don't need to do anything, you just go to git bash and then just clone the repository here or pull or push or or merge or whatever you want to do, okay? That's why I use gi. Another terminal that I have is the conduct. The conduct terminal.

s1 Speaker 1 ▶ 6 29

Here is my conduct terminal, okay? The conduct terminal installs A, as soon as I, it is Anana Pro install. As soon as I install my Anaconda, it's

installed. Let me explain a little bit about this one. Yes, it's a terminal different from the Windows terminal, but eh, it's embedded inside that window terminal and we can activate or deactivate conduct terminal, okay? And I'm gonna show you how to do it. This is Windows users. Guys, remember with Mac it's easier because with Mac you just have one terminal, and with that terminal you do everything. Okay? Here in Windows you can get confused a little bit with all terminals and sometimes you activate a virtual environment here in the Anaconda, but it's not activating it Bash for some reason. And then you have to look out what's going on, okay? But that's why I try to use all the time my gi, okay?

S1 Speaker 1 ▶ 7 41

Actually, I am going to do my demonstrations today in gi and I am going to, oh, Ancon for the, an environment that it's a little bit easier because you are actually doing the virtual environment in Anaconda, but you can do it here in Giba as well. But maybe you can get some errors here in GI Bash, okay? That you can fix, but it's more straightforward in Anaconda. So for the first part guys, for the first part, what I'm gonna do is I am going to show you the commands, okay? Before I start showing the commands, any questions about terminal?

S1 Speaker 1 ▶ 8 22

All clear in terminal aspects, what is the terminal in your computer? Anyone else? Nobody. Guest a guess. Maybe a wild guess. Anyone? No. Okay. The terminal is gonna be basically the, like the control command of your ship, okay? Your computer is a ship and basically this is is like the control command where you can do whatever you want to do in your computer, okay? You can install programs, you can delete programs, you can create files, create folders, delete files, run files. You can actually, well do whatever you, you think here, increase the storage space, separate your hard drive, you can do it everything through commands, okay? And actually in Windows you have something that is called run. And this run up helps you to, to create comments that goes directly straight to the common prompt. Okay? So that's basically the terminal. So here, when you access to your terminal for the first time in gi, it's gonna go to the default home folder.

S1 Speaker 1 ▶ 9 52

In my case, it's going to this folder that it says in, in folder C users and the name of the user of this machine, okay? In your case it will say your name or it goes, or if that goes to the C, then you can just move from C to users to desktop to whatever. And then you can change that default location, okay? In this case, whenever you create a Git Bash or you download the Git Bash is gonna open in your users. Why there? Because whatever you're gonna do to your machine, create a folder, whatever it's gonna be. Storing information, caches memories, whatever information you put there, it's gonna be storing it over there. You can change that, of course. How you're gonna change that. You put `cd`, short for, change directory, and then you put like `cd ..` and you say, I don't know how to spell desktop or I don't know if I'm gonna spell it incorrectly.

S1 Speaker 1 ▶ 10 59

So you stop tap basically. Okay? The tap key, the same in Windows, the same in macOS. Okay? So when I click enter, now you're gonna see this yellow part where it's saying that you are investor question so far, guys the terminal side, okay? So the other part is `ls`, like it says here, list files in the folder. So I don't have visibility here like I'm used to, right? With my graphical user interface. So when I click `ls`, I can see everything that is inside here, my desktop, okay? I have all that safety, right? So I can actually go to whatever folder, file, whatever thing I have here.

S1 Speaker 1 ▶ 12 01

Okay? I want to go back because I don't want to go to desktop. Then just click change directory space and to Dots. And then you just go back, okay? Or maybe for example, you're in `CD documents` and then you want to see what it's in documents. And then I want to go to `CD` inside documents where I have my work space. So right now I am in documents and inside WordSpace, and I don't want to do `CD space`, `cd`, spaceless Door. So I want, I just want to go to my home directory. `CD` in my keyboard is out of control. And this symbol, my keyboard is a Spanish keyboard with `any` in the keyboard. So maybe for you is different, but I'm pretty sure that all the keyboards have this. So I click here and I go directly to my home director. Okay? I don't want to, I want to go to the

workspace again and I don't want to type everything again. Just go with the up arrow on your keyboard and you can see all the commands that you have been read here. If you go down, you can see. So I want to go to documents. Then after that I want to workspace umbrella without typing anything.

s1 Speaker 1 ▶ 13 36

Question so far guys? So far so good. Okay, so I am going to my home director because I want to go to desktop. Where is desktop? Here's desktop. So here in the desktop, I want to show you the, well, we have P W D that shows the current directory. And I wanted to show you this Make a dear, right, make Dear. And then the folder name that is gonna be called my case example. That's it. So CD example exists there and it should be empty with Ls. I'm checking that is empty, but I'm in example, right? So I am going to create a new file there. How I am going to create a new file. I'm gonna write to, okay, because it says STO creates a new file with the file. And then I'm going to put, I dunno, first pie, my first Python file. Okay, enter. If I click Ls, I am going to see my first dot pie Python.

s1 Speaker 1 ▶ 15 05

Okay? Any questions so far? No. Okay, perfect. So now for example, in Mac you're gonna use, before deleting, deleting my files, I am going to do something in Mac. You use Open and opens the current folder in Windows, you type Explorer, and then space dot. And that opens the current folder in Windows. Or if you want to open in a specific file, that's what I'm gonna do right now. Exactly the same open, but without the dot an Explorer and without the dot. And you just add the file name. So in my case it's gonna be Explorer through what?

s1 Speaker 1 ▶ 15 53

Explorer. Okay. And then my file name is First five and it's gonna open it in my default I D I D E program that I have to open this kind of files. I explained to my machine that whenever I'm gonna use a Python Dopy extension file, open it in Visual Studio Code so I can start doing things right away. Okay? How you do that? Normally when you're gonna open in Windows specifically, you're gonna open an extension that your

computer does not recognize. It's gonna give you a message. How do you want to open this file? I have these programs available to open your file and then you select it. Do you want to select this option as a preferred option? Yes. Right? You have any questions on that? I mean, I, I am pretty sure that I don't need to show you how to do that.

S1 Speaker 1 ▶ 17 01

Yeah. Questions so far guys? No. Okay, so then I'm just gonna print something here in Python. I'm gonna print my first Python and I'm gonna save it control less and it's save. And then I'm gonna go to Bash. And if I want to run that, I just have to put five and then my F and I finish with the top and then run. And that's it is gonna print my first part. Maybe in your canvas when you were reading your, doing your study and reading it, saying that you will have to run everything with GI B, every Python program, you're gonna be running it in Giba and in market says in the terminal, okay?

S1 Speaker 1 ▶ 18 09

The theoretical speaking is true because Giba is running it. But we're gonna do a trick here that we are going to try to run this in my, in my visual studio. Let me check if I have everything set up for running. It is activating ways, blah, blah, blah. And if you have problems here, I will, I will help you how to, and that's it, right? Why? Because I add my terminal to my visual studio code and I will, will help you how to add terminals, for example, split the terminal, add the terminal, and I add my bash here.

S1 Speaker 1 ▶ 19 09

Okay? And it's pretty easy because you can start doing things right away here, print or text, save it, remember, you have to save it, it's important. And then just run. Okay, how are you saving it? Controls, that's it. Controls or you just click here, save. Okay, and then you're, you're running first Python, other text. And it's easier because you can see what are you doing if you, you have to debug. For example, when I put print, it's showing me something here like show code actions. And it's giving you like different things. Not really useful to be honest, but when you run it, you see the error name print, it's not defined what is print. It's not a function invite. Okay? Questions so far?



s3 Speaker 3 ▷ 20 34

Hug. I have a question.

s1 Speaker 1 ▷ 20 36

Yes, Jerry?

s3 Speaker 3 ▷ 20 37

So, so currently on your screen, you're in Xcode, right? Or

s1 Speaker 1 ▷ 20 45

No BS code? Yes. D S BS code.

s3 Speaker 3 ▷ 20 49

Okay, so you said that I have a Mac, so everything should be in my terminal.

s1 Speaker 1 ▷ 20 56

No, you actually can have a BS code in your Mac as well. This is a different program. Look, look in my, it's an application.

s4 Speaker 4 ▷ 21 04

I don't think you started yet. A lot of us are confused cause of that. I think he was just showing his commands.

s1 Speaker 1 ▷ 21 12

Oh, okay. Yes.

s5 Speaker 5 ▷ 21 14

Yeah, I've got a, I've got a window in a Python, but it's not, it doesn't have any of your icons or anything. It's just like a, a command lines where I can type pi Python commands.

s1 Speaker 1 ▷ 21 27

Yes. I mean this Visual studio code is, whenever I open a dot pi file, it's open here. It's like, you can actually do this as well. This block, this notepad, right? You can put here print, and then you put my first Python. Oh, because it's a text. You just save us and then you save it as not txt, but five Python note, note, python note, save it here in my example

folder. And that's it. Okay. I, I didn't open it here. I mean, this is not nothing fancy. I'm not doing anything like, wow, here opens a new, new application or anything. No, we're gonna, we're gonna talk about this one. Don't worry about that. But it's just, when I, whenever I open a Python file, it gonna go straight to the Visual Studio code and we're gonna, we're gonna see how to do it. Okay? So for example, here I in ls, and you'll see that I have my two Python files, one that I created in my notepad and the other one that I created in my Visual Studio code. But at the end, the thing that is gonna make your file a Python, it's the dot pie extension.

S1 Speaker 1 ▶ 23 21

Okay? That's clear or not Questions. Do we need to download Visual Studio? Yes, you will. I mean, it's, it's my recommendation and when I'm, when I'm start talking about an a condom, I'm gonna tell you how to do it. It's my recommendation because it's really useful. Whenever you start a job or, or anything like that, many companies use Visual Studio Code, but some others, some others, well, I, I, I work in, in Pie Charm, I work in Visual Studio Code and I work in, what's the other one? Spider. Spider as well. And it depends on what you like to be honest, but all, they all have quite the same purpose. Okay? Okay, thank you. I'm, I'm gonna show you, I didn't want to confuse you there, we just want you to go to the program and show you how everything runs. But basically it's like that I'm still working in the terminal as you can see. Okay, thank you. No problem.

S6 Speaker 6 ▶ 24 28

Hugo, for some reason, my anacon, it's not downloading. I think that there's, there's an error with my Anacon for some reason.

S1 Speaker 1 ▶ 24 37

Okay? We can, we can check on the errors. I mean, after I pass this terminal part, we're gonna see the Anaconda part step by step. So you can, you can check because there is one step that is really important in Windows users. Okay? And then we can do some debugging there, Kelly.

S6 Speaker 6 ▶ 24 57

Okay, sounds good. Yeah, I have a Mac.

s1 Speaker 1 ▶ 24 59

Oh, you have a Mac

s6 Speaker 6 ▶ 25 01

Code already has Python, but for some reason like Anaconda, it's not, it's not working at all.

s1 Speaker 1 ▶ 25 06

Same. Okay. Okay, perfect. Anyone? Adrian, do you have any questions? No. No, no. Okay, so, so now I'm going to show you, for example, I have that. So if I want to remove a file, specific file, I'm gonna put `rm` and then my first Python, then I include `ls` and I remove that file. And if I want to remove the whole folder, right? I'm just going to, I need to go back one because I have to work in desktop and I have to put `rm` then notices slash `r` and then name of this, the folder that is sample.

s1 Speaker 1 ▶ 25 59

So that's it. `ls` and I don't have example in my desktop anymore. Okay? So that's how you work with the terminal. Any questions on the terminal part? Perfect guys. So I'm gonna give you some time, like 10 minutes for you to actually start working on your terminal, okay? The activity, you can find it here where it says Python three day number one activities, and then you open this second one. Now is your turn to do some work in your terminal. You will create three folders and a pair of Python files to print strings of your own creation to the console. Okay? Use the following instructions to write commands in your terminal. Create a folder, call Learn Python, navigate into the folder inside that learn python. Create another folder called assignment one. Inside assignment one. Create a file, print a statement, run your Python, return to that folder, and well just have to do all, all these instructions. Okay? Questions about the activity guys? Perfect guys. So let's, let's work in this. We have 10 minutes to work in this part and then we can, we can review it. Okay?

s5 Speaker 5 ▶ 27 55

Its like, this is where I was before when you were going through. I'm, I'm under, I'm in a, in a editor with quick Python and have a print statement, but there's just icons and I have no idea what any of them do. So I don't even know how to run it.

s7 Speaker 7 ▶ 28 12

Yeah. Wait, I thought we were, you were gonna go over how to install it. It says on instructions as well. You just went over the commands.

s5 Speaker 5 ▶ 28 19

Like I've got some weird graphics at the top of the screen that I have no idea. Adjust editor options. I must be in a totally different editor.

s8 Speaker 8 ▶ 28 32

I'm sorry. Are we working in exercise number one or two? Sorry, I messed up.

s9 Speaker 9 ▶ 28 41

You go. You're muted by the way, f dear.

s1 Speaker 1 ▶ 28 45

Oh, thank you very much. Yes, number two. Thank you Andrew. Alan, if you want, you can share your screen please.

s1 Speaker 10 ▶ 28 55

And also, I forget too, like how to download the, download the whole file with s h ssh key. Like I forget to come in June. Anybody know about that?

s1 Speaker 1 ▶ 29 08

You just put GI clone in your GI badge or terminal and then you just put the s go to GitHub and copy not the H D P. Do you copy the s s h?

s1 Speaker 10 ▶ 29 24

Okay, thank you. I got it.

s1 Speaker 11 ▶ 29 27

Or, or GI pull if you already have it cloned.

s1 Speaker 1 ▷ 29 30

Yes.

s1 Speaker 11 ▷ 29 31

Okay,

s1 Speaker 1 ▷ 29 34

So Alan, you are here. Yes. You have to put the brackets there. Okay, now save it, save that file. Sorry, you're speaking in, in mute if you want to face, no, I think the safe part is gonna be up right on the Mac menu. Yes. File and save. Yeah. Okay. Go to your terminal. To your your terminal in Mac. Yes. Okay, go to that file wherever you have that file quick. No, no, it's Python, quick python pie. If you're in the right folder. Oh, well this, you can, it's Python type. Python space. No Python space. Quick the name of your Python file.

s2 Speaker 2 ▷ 31 35

You're muted. Alan.

s1 Speaker 1 ▷ 31 37

Yes, you're muted Alan,

s5 Speaker 5 ▷ 31 39

Thank you.

s1 Speaker 1 ▷ 31 40

No problem.

s5 Speaker 5 ▷ 31 41

All Python, quick Python, dopy. So the wrong command is Python.

s1 Speaker 1 ▷ 31 46

Yes. Not found,

s3 Speaker 3 ▷ 31 52

I actually do Python three quick Python, BYN Mac and it works.

s1 Speaker 1 ▷ 31 57

Okay, try to do that. Alan Python three.

s5 Speaker 5 ▶ 32 01

Oh yeah, I am Python three.

s1 Speaker 1 ▶ 32 03

Okay.

s5 Speaker 5 ▶ 32 06

When I installed it, I remember now. That's what it named it.

s1 Speaker 1 ▶ 32 12

Thank you Gupta. Okay, let's

s5 Speaker 5 ▶ 32 13

See. Yeah, thank you very much. Yeah, cuz when I first installed it, it, it's what, when I first installed it, you probably did the same thing. We had to put it in a different directory and, and then it named it Python three.

That's must be the current gen Gen. Oh, okay. For 3.1, right?

s1 Speaker 1 ▶ 32 29

Yeah. Correct.

s2 Speaker 2 ▶ 32 32

I have a very silly question. Thank

s1 Speaker 1 ▶ 32 34

You. No problem. Alan. Yes, Denise, you

s5 Speaker 5 ▶ 32 35

Are the hardest part is all these little things. Cuz then you can't test,

s1 Speaker 1 ▶ 32 38

Right? Yes. No, no, no. And

s1 Speaker 11 ▶ 32 40

It's not intuitive like,

s1 Speaker 1 ▶ 32 41

You know,

s8 Speaker 8 ▷ 32 43

So I was, I'm on my desktop where I created my folder and I put cd learn Python because I wanna open it and it says that not such to file Oratory learn Python. But when I, yeah.

s1 Speaker 1 ▷ 33 00

Yes. If you want to share your screen really quick.

s8 Speaker 8 ▷ 33 02

Sure.

s1 Speaker 1 ▷ 33 03

I'll stop here online. Thank you Alan. I dunno. My camera, it's like, whoa.

s8 Speaker 8 ▷ 33 12

So that is, I say I put, I'm in my desktop and I put, you know, look the files and I see Learn Python. And when I put cd, learn Python, it is a non-social file or Atory,

s1 Speaker 1 ▷ 33 28

Well learn Python. It's a folder or what it is.

s8 Speaker 8 ▷ 33 31

Yeah, it's a folder. It's right here.

s1 Speaker 1 ▷ 33 33

It is there.

s8 Speaker 8 ▷ 33 34

And it's on my desktop.

s1 Speaker 1 ▷ 33 37

Learn. Okay, try to click LS please. There.

s8 Speaker 8 ▷ 33 47

So I see it here.

s1 Speaker 1 ▷ 33 49

I dunno why you see, you are in a different desktop. I know that Mac can handle different desktops. I don't know if it's that or

s8 Speaker 8 ▷ 33 57

Oh,

s1 Speaker 1 ▷ 33 58

Really?

s1 Speaker 11 ▷ 33 58

It's a spelling issue. In this case.

s1 Speaker 1 ▷ 34 00

It's a spelling issue. Is it? Okay? Oh yes, it's there. So I think it's l e r r put cd, then space.

s8 Speaker 8 ▷ 34 13

Oh, CD in space.

s1 Speaker 1 ▷ 34 15

Well, you are not in desktop anymore.

s8 Speaker 8 ▷ 34 19

Okay, so should I go back

s1 Speaker 1 ▷ 34 21

To desktop?

s8 Speaker 8 ▷ 34 25

I'm in users now.

s1 Speaker 1 ▷ 34 28

No, go back to desktop.

s2 Speaker 2 ▷ 34 31

Mm.

s8 Speaker 8 ▷ 34 37

Oh, so I have to go back. Okay, let me



S1 Speaker 1 ▶ 34 42

CDs, space two points. Now see the desktop, if you go with the up arrow, you can see Yeah,

S8 Speaker 8 ▶ 34 55

Yeah, it's weird.

S1 Speaker 1 ▶ 34 59

Did she do a PW D to see where she is? Exactly. PW D. Okay,

S1 Speaker 11 ▶ 35 09

Go back to your home directory with the CD and then the little squiggly line. You you're, you're up a couple folders up.

S8 Speaker 8 ▶ 35 17

Yeah, that's why I'm trying to do the te, so

S1 Speaker 11 ▶ 35 20

CD tell, tell, I was trying to remember that one. Yeah.

S8 Speaker 8 ▶ 35 24

Oh, there it is.

S1 Speaker 11 ▶ 35 25

And then you can, you should be able to go through the desktop from there.

S1 Speaker 1 ▶ 35 32

Then cd, then click only L, no, no cd. CD space L and then Top, top key.

S8 Speaker 8 ▶ 35 46

It's not working.

S1 Speaker 1 ▶ 35 48

Tap is not working. Okay. Not capital L.

S1 Speaker 11 ▶ 35 51

Sometimes it can be like that.

s8 Speaker 8 ▷ 35 53

Capital L.

s1 Speaker 1 ▷ 35 54

Yes, capital L Top Tap. That's it. And, and yeah. Yeah.

s8 Speaker 8 ▷ 36 05

So you mentioned that in, in Mac it's two desktops.

s1 Speaker 1 ▷ 36 10

Well you can actually have different desktops, but not in the folders.

s8 Speaker 8 ▷ 36 15

Oh yeah, you are right. There's different, you're right, you're right. So, okay, is there any way to know which one I'm, I'm at? Just,

s1 Speaker 1 ▷ 36 23

No. Well, to be honest, it's just one folder where you can actually say everything and it's that, that desktop one that you are. Okay.

s8 Speaker 8 ▷ 36 31

Okay, thank you.

s1 Speaker 1 ▷ 36 33

No problem. Okay guys, two more minutes, we can check. Well, it's not, we can check the installation part. Okay, perfect guys. So Time's Up. So any questions about that activity? Terminal activity?

s2 Speaker 2 ▷ 38 18

I

s8 Speaker 8 ▷ 38 20

Do, when you said that we are gonna create a print statement, I'm already in my touch with Python, which it means that I already created the file. Did I have to go against CD or I just put the run the print Quick Python pie.

s1 Speaker 1 ▷ 38 39

Yes. If you created the file already with the print statement, you can run it directly in the terminal Python and then the name of your file.

s8 Speaker 8 ▶ 38 47

So the file is there, but I have to write inside the file, is that right?

s1 Speaker 1 ▶ 38 51

Yes. You have to write inside the file, the print, the statement.

s8 Speaker 8 ▶ 38 54

And how do I write the file inside the name, inside the file?

s1 Speaker 1 ▶ 39 00

How you do it. In any, if you have Notepad or you have any text editor or you can Oh, okay. Yes, in any text you can do it.

s8 Speaker 8 ▶ 39 12

Awesome. I thought I, I could, I cannot do it directly on the terminal, right?

s1 Speaker 1 ▶ 39 18

Yeah.

s8 Speaker 8 ▶ 39 20

Okay, cool. Thanks.

s1 Speaker 1 ▶ 39 22

No problem.

s1 Speaker 1 ▶ 39 26

Okay, perfect. So what I'm gonna do now say, I'm gonna speak about the Anaconda installation, okay? How we go on this? That's the other part. So basically you have the instructions on your canvas. I'm pretty sure that they give you like step by step on how to do it, eh? So here actually, the only thing that we have to do is just to make sure that you have an Anaconda installed or if I go to an Anaconda in Google, an Anaconda download, it's free, it's an open source. This Anaconda, it's really cool to be honest. You can download it for Linux, for Mac, for Windows. Okay? And it gives you like different instructions on how you can take more advantage of this.

One thing is that for Windows, in order for an account to work, you have to set up the, in the environment environment variables.

s1 Speaker 1 ▶ 40 38

Okay? In Mac, you don't need to do that to put Anaconda in the environment variables. You just, you just have to ping. I'm pretty sure that in Canvas they give you where you just have to ping or or thick the box where it says include an accon in my path variables. And then, and then I think that's pretty much it on that part here, right? So does anyone here has an ACCON installed already or you went to the, through the process of installing an accon? Yes. Yes. Okay. Mine is not working. It's failing. Who is not working? Calvin is not working. Who else is having problems with installation of Ancon?

s1 Speaker 1 ▶ 41 45

Okay, then raise your hands. Okay, in this case, let's do something. Let's create, I'm gonna create the breakout rooms. I'm gonna create four breakout rooms. And if you are having problems installing an accon, please one of the TAs and of Andrew, Andrew or Joe or Bill, if you can go to one of the rooms or maybe two of them and try to help the people that it's the minority with their problems on the installation side, that will be awesome. Okay? I'm gonna open the rooms now and just go to one of the rooms with one of the TAs if you have problems installing an accordance, okay? If you don't, please stay here.

s1 Speaker 1 ▶ 42 42

And then after you fix your problems, you can come back and well you can, you can check out because we are just gonna do some installation. Installation checkups. Okay, perfect. So in order to do the installation checkups, what I'm gonna do is I'm gonna close my GI Bash. Okay? If you go here and then you type an Accon or you go to find their in Mac and type an aconda, it's gonna open your an ACONDA application. This an Aconda application, like I mentioned before, it is really powerful and really helpful because I mean, you just need to have the permission of your company to get it. But I, I haven't heard any companies that they don't allow the use of a free open source. And here is what I was talking about, right? You can download different, different text editors like

Charm. It's like a visual studio code. So you can write your Python there. BS code, the one that I show you, spider, it's another powerful Python. I d e for graphs. And actually we're gonna use this one, our studio later on in this course. Question so far about ancon, okay, another way to check that I have ancon installed. I'm gonna open my GI badge. You open your terminal. Hugo,

s1 Speaker 12 ▷ 44 31

I have one question.

s1 Speaker 1 ▷ 44 33

Yes,

s1 Speaker 12 ▷ 44 34

I don't, I went through the document, the module, and I installed the extended version of Anaconda. I already have a bunch of them, bunch of tools already installed. I don't have R Should I be installing that separately or is that for you? Oh

s1 Speaker 1 ▷ 44 52

No, you can actually look it out here. Okay. When you have your ANCO navigator, you can look R here.

s1 Speaker 12 ▷ 44 59

R Studio. Okay. All right. Yeah, because I didn't, I don't ha, I didn't see that in my applications, so I just wanted to check, but we don't, we are not going to use it right now, right?

s1 Speaker 1 ▷ 45 09

Not right now. Okay. And we're gonna go over that.

s1 Speaker 12 ▷ 45 12

Okay, cool. Thank you.

s1 Speaker 1 ▷ 45 13

No problem. So whenever I'm here, I can just initiate my conduct with this command conduct in it. And in this case, I'm gonna use Bash because I can use other shells. So Con is installed, they recommend you

to actually close your command prompt and open it again GI Bash. Okay, here you just put Conda slash slash version, and you see my version is four point 10.3. It's all, I know I have an update my version, but it still works.

s5 Speaker 5 ▶ 45 59

Mine's 2, 3, 5 0.0 that I did yesterday.

s1 Speaker 1 ▶ 46 03

Which 1, 2, 3

s5 Speaker 5 ▶ 46 04

0.5 0.00,

s1 Speaker 1 ▶ 46 07

Okay.

s5 Speaker 5 ▶ 46 08

Is that the current version?

s1 Speaker 1 ▶ 46 10

No, that's okay on the versions, because for the moment, it doesn't matter too much about what version we have in Conda. The Python one, yes, because of the libraries, but Conda not so much for the moment. Okay? But if we want to fix that, we just go and update, upgrade now and then just upgrade your version to the latest version. Okay? So now you have your version and I am gonna explain a concept here. Everyone is good in this part version.

s1 Speaker 13 ▶ 46 46

What if it says Conda command found?

s1 Speaker 1 ▶ 46 52

Because

s1 Speaker 13 ▶ 46 53

You put Conda, you run Conda in it Bash,

s1 Speaker 1 ▶ 46 56

And it's not coming up. Anything Eric there?

S1 Speaker 13 ▷ 47 00

No.

S1 Speaker 1 ▷ 47 02

Do you have a Windows or a Mac?

S1 Speaker 13 ▷ 47 04

Windows?

S1 Speaker 1 ▷ 47 05

Okay, try in here. Try to look for an Accon prompt, please.

S1 Speaker 13 ▷ 47 12

All right,

S1 Speaker 1 ▷ 47 13

You have it?

S1 Speaker 13 ▷ 47 16

Yeah.

S1 Speaker 1 ▷ 47 17

Yeah. Okay. Open that one slash slash version.

S1 Speaker 1 ▷ 47 30

It's working. Yeah, yeah. Okay, perfect. So in your case, that's exactly what I was explaining. In your case, you will have to link, or not link, but to, to start Conda in your gi. Okay? But that's a different process for the moment. You can use your Anaconda prompt for this example, okay? They're all connected, but in different shells. So the next part is we're gonna, we're gonna explain about the virtual environments. Yes. Any questions? No. Okay. So we're gonna do a virtual environment. Okay guys, why we need a virtual environment. Let me put this like as clear as possible. Virtual environments work as different rooms in your computer, okay? Environments can hold different tools. Let's see it like that. For example, your computer right now, it's an environment. And in your computer you can have the tools to work as a data scientist, right? But

maybe you need more room or you need another environment to work as, as a marketing guy.

S1 Speaker 1 ▶ 48 56

Why? Because that's your side, your side job, okay? But you know that you're gonna need some of the data science tools as well, but you want to keep everything organized, right? So that's why you create a virtual environment. One thing that I want to remind you here, guys, we recommend you to work in virtual environments. Why? Because the virtual environment is not going to affect anything on your local machine, okay? The virtual environment is virtual, it's something, it's a machine created momentary whenever you start using it. And it's a workspace that you create to do your test, to put up your code, to just keep things about marketing and data there. Because in your computer you want things about data only. So that's the whole use of virtual environments. I, I know people in the industry that says, I like to work everything in local and whenever I'm gonna do a deployment in production or, or, or in u a t or in in per, or something like that, I create my environments because I will do all my deployments in environments, but I will like, I like to work locally every time, right? And that's fair enough. This is your choice, okay? And so here I'm gonna show you how to create a virtual environment. Any questions about virtual environments, guys?

S1 Speaker 1 ▶ 50 31

Okay. So if your con commands are not working in it, but try to use the Anaconda prompt in Mac, use your terminal, okay? So you're gonna write Conda and then create and then slash N. And then what you're going to put here is the name of your virtual environment, okay? In my case, I am going to call it depth. And I'm gonna, I'm gonna hit enter. So it's gonna be creating the environment. Yes. Okay. So apparently it's created and it's telling me to activate depth. You just put Conda, activate depth, and to reactivate it, you just put Conda deactivate.

S1 Speaker 14 ▶ 51 20

How do we know the name of the mutual environment?

S1 Speaker 1 ▶ 51 24



You actually choose that name. Any name that you want to put there.

s1 Speaker 14 ▷ 51 28

Oh, okay. Okay.

s1 Speaker 1 ▷ 51 29

It can be your virtual Mimi environment, something like that. Okay. Right. I just put it there.

s1 Speaker 15 ▷ 51 35

Hey Hugo, can you scroll up just to see what you typed in? Conda?

s1 Speaker 1 ▷ 51 40

Yep. Conda create

s1 Speaker 15 ▷ 51 43

Conda create Awesome. Dash and Dev.

s1 Speaker 1 ▷ 51 47

Okay. And then here's the name of the environment. Okay, so then I want to go to that environment. As you can see here in my terminal, it says base, that's my local environment. So I click Conda, activate Def, because that's my name of the, my virtual environment. And now I am in my virtual environment.

s1 Speaker 12 ▷ 52 13

You go one question.

s1 Speaker 1 ▷ 52 15

Yes.

s1 Speaker 12 ▷ 52 15

Is there a way for us to check if you already have a virtual environment set up?

s1 Speaker 1 ▷ 52 21

Yes, there is a way. And right now, to be honest, I will need to check here. Well,

S5 Speaker 5 ▷ 52 27

I tried to create Dev again and I didn't realize I created yesterday. And so it told me, already created and then asked me if Juan wanted to replace, and I said no. Oh,

S1 Speaker 12 ▷ 52 35

So you did the same create dev.

S5 Speaker 5 ▷ 52 37

Yeah. So I must have done this yesterday. I didn't realize I, yeah, I, I've done it. I tried to create dev again and I said replace, I said no. And then Dev,

S1 Speaker 10 ▷ 52 46

It's also my questions like how can we just, you know, change the, change the environment here,

S1 Speaker 1 ▷ 52 56

Here in how you can change the environment.

S1 Speaker 10 ▷ 53 00

No power share prop. I already just create a dev in the, in dev like a couple, couple hours ago. Bill told me to do that. But like, since I want to do another dev, and they said a Conda environment already exists at blah, blah, blah. And you want to remove existing environment, why? Or no?

S1 Speaker 1 ▷ 53 26

Yes. Basically you can remove that environment and create your new environment or you can assign a different name to the new environment. Okay? So you can put dev one, dev two, dev three, dev four, and then you can create as many envi virtual environments as you want.

S1 Speaker 14 ▷ 53 47

Do we remove the environment by deactivating it?

S1 Speaker 1 ▷ 53 52

No. Actually I have to check how to, well, definitely if you want to remove an environment, for example, I am going to put the bait, I'm gonna go to base, and if I go again to create in depth, it's gonna tell me Dev environment already exists. You want to remove it and I can remove it, click yes. Okay. But I owe you the commands to actually remove environments to check how many environments you have created and all those commands that I know they exist because I used them before, but I, I owe you that part. Okay. I'm gonna check where I can find that and I send you,

S3 Speaker 3 ▶ 54 41

Okay, thank you.

S1 Speaker 12 ▶ 54 42

And Alan, thank you Hugo. Thank you. I've, I actually got the same message that Dev already exists and I just said no. And I was able to activate contact, activate dev, and get into dev environment.

S1 Speaker 1 ▶ 54 55

So Exactly. You can do that. Yeah.

S1 Speaker 12 ▶ 54 57

Thank you.

S1 Speaker 1 ▶ 54 58

Perfect. Any

S1 Speaker 16 ▶ 55 00

Questions? I just found, if you wanna know the name of the environment that you have on give list of environments, you have

S1 Speaker 1 ▶ 55 13

Perfect list. Maybe it's working for you, maam. Because for me

S8 Speaker 8 ▶ 55 34

Is it con info list or just con that list?

S1 Speaker 1 ▶ 55 39

Let me try con that list and that's it. Oh no, it's not that easy.

s8 Speaker 8 ▷ 55 49

It's not that it's not it,

s1 Speaker 1 ▷ 55 51

It's not it. It is giving me everything that it's in contact.

s3 Speaker 3 ▷ 55 55

Okay.

s1 Speaker 1 ▷ 55 56

It

s1 Speaker 16 ▷ 55 56

Worked for me, so that's why.

s1 Speaker 1 ▷ 55 58

Okay, so then maybe you can share your, your command and they We can try, you're using a macro Windows.

s1 Speaker 16 ▷ 56 05

Windows,

s1 Speaker 1 ▷ 56 06

Windows. Okay.

s3 Speaker 3 ▷ 56 09

You go. I just found out that if you go to Anaconda Navigator,

s1 Speaker 1 ▷ 56 14

Oh yes.

s3 Speaker 3 ▷ 56 15

And then click on the environment, then you can see the environments here. Just from the left side, you see environments?

s1 Speaker 1 ▷ 56 23

No, I'm here in Applications one. Oh, you actually can see all the applic. Yes. Thank you for reminding me. Ok. Yeah, this is all the environments

that I have been created. Yeah. As you can see here. And it's, it's user friendly and you can delete them here as well. That's why I don't remember the commands. Yeah, thank you for that.

s8 Speaker 8 ▶ 56 51

You're welcome. In list also worked for me. It's working. And I have a Mac.

s1 Speaker 1 ▶ 56 57

Okay. So yes. Can you base the command?

s8 Speaker 8 ▶ 57 04

It's E N V? No.

s1 Speaker 1 ▶ 57 07

Oh

s8 Speaker 8 ▶ 57 08

Yeah, yeah. E N V list. Yeah. There you go. That's

s1 Speaker 1 ▶ 57 12

It. Yeah, I was showing on S Slack so you can see it. Thank you very much. Yes, that works. Thank you. Thank you very much. Perfect guys, you're getting experts in this. That's great. Any questions? Okay, great. How do you get to Anaconda Navigator? I just go to your finder in Mac because you're a Mac user. Go to the Finder and then you just try type Anaconda Navigator. You will find it there. Perfect guys. So let me jump to the next activity of today's class. We're gonna actually jump right away to Python. Okay. Work in any editor that you want. I am going to work in BS code because I have it here already. That's why it says on the launch here doesn't say install or anything. So I'm gonna work in that one. So I recommend you for this course to work in Visual Studio. It's friendly, it's really easy to use. So I'm gonna go to Visual Studio Code here. You can see

s1 Speaker 12 ▶ 58 39

Hugo, I have one question. I'm sorry.

S1 Speaker 1 ▷ 58 41

Yes,

S1 Speaker 12 ▷ 58 42

I do have Visual Studio installed and I was able to launch it, but I think the preliminary steps of what you did in the beginning of the class where you

S1 Speaker 1 ▷ 58 51

Mm, well I will repeat it now.

S1 Speaker 12 ▷ 58 54

Yes, thank you.

S1 Speaker 1 ▷ 58 56

No problem. Here in Visual Studio, it's giving you like the welcome page, right? Where you can open a file, you can create a new file, open a folder or whatever you want to do. The first step that you, that I recommended you to do is go to this square extensions part. Try to find for this, no, just type Python and this Python you, you install this one, Python extension pack B 1.7 0.0.

S1 Speaker 14 ▷ 59 30

So I have to download Visual Studio first to be able to do this.

S1 Speaker 1 ▷ 59 35

Yes. Or you can go to an condo and a condo navigator and then launch it from here. Okay. Okay. But you can download it as well or you can use any other text editor. I'm explaining this one because I, this is the one that I use and I recommended for the group. For the course maybe to be honest. Yeah. Yes. Actually to install Python. What Python? This Python extension pack. And I'm gonna be, I mean the rest of the course I'm gonna be using Visual Studio to be honest. So it's gonna be better for you to follow up with this one.

S1 Speaker 1 ▷ 1 00 28

Okay. So once it's installed, we can actually start doing things here. I'm gonna go to the explorer and I'm gonna open a folder here, open a folder.

I am going to go to my folders where I have everything here, users, this one. And then I'm going to my folder, highton day one activities. And I'm gonna open all the activities. Okay. Basically you have here I trust the auditor. Yes. I trust everything. Great. And basically what you have here is all the, well, I will have all the activities for today's class. Okay. I'm here variables. And I am going to actually open this and I can see the variables. Okay. You are

S1 Speaker 12 ▶ 1 01 38

Not variables. Yes. I'm sorry. You're not creating a folder, you're just exploring and you're going to the folder that you want to open the file from. Right?

S1 Speaker 1 ▶ 1 01 46

Exactly.

S1 Speaker 12 ▶ 1 01 47

Okay.

S1 Speaker 1 ▶ 1 01 48

You can create folders here in Visual Studio Code J.

S1 Speaker 12 ▶ 1 01 52

Okay. Alright. But right now you, we don't need to create a folder at the moment

S1 Speaker 1 ▶ 1 01 57

We don't. But sorry, sorry. I was wrong. You can create folders here. Look, okay. You folder in your file. Okay,

S1 Speaker 12 ▶ 1 02 04

Got it. But this is the way that you navigate to wherever you want. If your Python files are already present somewhere and you wanna open them, that's how you navigate.

S1 Speaker 1 ▶ 1 02 13

Exactly.

S1 Speaker 12 ▶ 1 02 14

Got it. Okay.

S1 Speaker 1 ▷ 1 02 14

Thank you. Yes. I mean you will have all your activities here, right? You're gonna work with them soc once, right?

S1 Speaker 12 ▷ 1 02 20

Ok. Right. And this is for Python, right? This is not something that will be applicable for Yeah, obviously vba, VBAs a different thing altogether.

S1 Speaker 1 ▷ 1 02 29

Ah, exactly. But here you can open css js, you have a script html, you can open like a lot of of programming languages.

S1 Speaker 12 ▷ 1 02 41

Got it. Okay. Thank you.

S1 Speaker 1 ▷ 1 02 43

No problem.

S5 Speaker 5 ▷ 1 02 44

I'm in the opening, but I'm not seeing the same thing you have there.

S1 Speaker 1 ▷ 1 02 49

Oh no, this one is just for the instructor, the soft part to go through the variables. Alan. So this one I'm gonna, we're gonna publish as soon as I finish. We, we can push these so versions.

S1 Speaker 14 ▷ 1 03 07

Okay. Can you me again how you got there?

S1 Speaker 1 ▷ 1 03 11

Yes, no problem. I'm gonna, I'm gonna close everything. I'm gonna open everything again. So I'm here open folder, I go wherever is my folder location in my case. I know where is it sir? Here. And it's gonna be here and I open it. Oh well I open everything.

S1 Speaker 14 ▷ 1 03 45

Okay.



s1 Speaker 1 ▶ 1 03 50

To look out and then I'm gonna close this one. I'm gonna close Trust, trust and then close everything.

s5 Speaker 5 ▶ 1 04 09

So we should start like do the walkthrough, get started with VS code and,

s1 Speaker 1 ▶ 1 04 17

Sorry, what was that Alan? It

s5 Speaker 5 ▶ 1 04 18

Sounds like we should do the walkthrough, get started with VS code and get real familiar with this since you're gonna be using it from now on for the coding exercises.

s1 Speaker 1 ▶ 1 04 27

Yes, exactly. I mean there is nothing to be honest, to give you like how you use BS code. Bs code because basically you can choose whatever you want, whatever text you want. I'm gonna be showing this, but they, they pretty much work the same. So that's why we don't go in detail on how to use it. But I mean we're gonna, we're gonna learn how to use it on the run. Okay.

s8 Speaker 8 ▶ 1 05 10

I have a question. My, I don't, I am not able to even type on my, on my VS code.

s1 Speaker 1 ▶ 1 05 20

Oh, I haven't checked that. Well in that case, Denise, let's, let's check on those errors after class. So we can just move on with the variables part or you can actually Slack one of the TAs. Maybe they can help you sharing your screen, I mean sharing your screen images or something like that. I'm pretty sure that we can help you on that things. Okay. Alright. Right, thank you. So we're gonna talk about variables. Basically we don't have much to talk about variables. It's like A B V A. But here you don't have to put ding or variable or anything. You just put the name of

your variable equal to and whenever you put like strings, it's gonna know that it's a string. Whenever you put an inch chair, it is going to know that it's an inch. Whenever you put a bullion, it's gonna know that it's a bullion.

s1 Speaker 1 ▶ 1 06 13

One important part about bullions is that whatever you put true or false, you always start with a capital letter. Okay? Because if not it's gonna, it's gonna give you an error. You can see this, right? It's gonna give you an error and it's gonna correct everything that you do. Print a statements, pretty much we did it a string with a string. You concatenate with a plus pretty much the same as B. A. In order to convert an integer into a string, you just put the word `s t r`, then you open parentheses and then you put `J` the name of the parable plus `J` this another string. Then you can convert a bullion into a string with `S str R` as well. And the `F` string accepts all data types without any conversion. The only thing that you have to know here is that you're gonna put this little `F` that is gonna paint before your message and whatever variable you want to show you just have to put it in fully brackets. Okay? And that's it. Any questions about the variables? What does the `F` do? The `F` do? Basically it's accepting all kind of variables is the `F` string. You don't have to convert the strings to `i` meanger to strings or bullions to strings. You just have to put whatever variable you want to use in your printing message in curly brackets. Okay? And this basically, it's gonna show you the variables. Nick is a professional frontier and all the printing. More questions about variables guys,

s3 Speaker 3 ▶ 1 08 08

How did you run this on the terminal?

s1 Speaker 1 ▶ 1 08 11

Perfect. No, here in the Randi bar, you click it, okay and then run on the bar and that's it.

s1 Speaker 12 ▶ 1 08 23

Okay. Thank you Hugo. One question, I think I already had Visual Studio installed before I installed Koda. Koda, does it install another vs code

when I do that?

S1 Speaker 1 ▶ 1 08 42

No, it's not going to let you, your machine is gonna tell you that that program already exists. Well, if it's another version, yes, it's gonna let you install it. Mm.

S1 Speaker 12 ▶ 1 08 52

Because I see two visual studio codes on my, I don't know what the bar and it is confusing which one to open. So I just wanted to check with you, but okay,

S1 Speaker 1 ▶ 1 09 06

I mean we can, we can check it as well in office hours, no problem. Okay,

S1 Speaker 12 ▶ 1 09 09

Sure. Thank you.

S1 Speaker 1 ▶ 1 09 11

Okay, great guys. So perfect. So we're gonna work in the next, the next activity of today's class. Okay? For the next activity of today's class, actually we have a one easy activity. Basically is a hello war variable. You will have to create Python two variables called name and country and well to follow all the instructions and then print it. Okay? If you are not using BS code, that's okay that you create it in a, because for example, guys, like I told you, oh, I don't know how to create that in BS code. I can create this in my notepad because Hugo didn't explain us how to use BS code, right? So I can use my notepad, any text editor, I can create my code here. Okay? I can type my Python here and then I can file save as the name of my Python. Okay? You know, variables with extension pie. Okay, I am going to save it here in my desktop save. Okay? And then I can go to giba. I am going to go to my desktop part. Okay? And then I'm gonna put Python and then ba, lemme check this variables. It's variable Python variable five and that's it, right? I can run at it there and I don't go to complications in the code. How do I run my code and all that kinda things.

S1 Speaker 1 ▶ 1 11 17

Okay, the whole point here is to you for use Python and you can use it in a text detail. Any text detail, okay? So please start working on this activity. We are actually have only 10 minutes to work on this and then we can start reviewing it.

S8 Speaker 8 ▶ 1 11 37

I'm sorry, I'm still waiting for a ta. I'm not able to type it. Just, I don't know what's going on.

S1 Speaker 1 ▶ 1 11 44

Oh, okay. Any TAs available to help Denise in one of the breakout rooms? Let me check. Maybe they are all in breakout rooms. We have, yes, if you go to room two, Denise, you can find Andrew Ecker there.

S8 Speaker 8 ▶ 1 11 59

Awesome, thank you.

S1 Speaker 1 ▶ 1 12 00

No problem.

S1 Speaker 14 ▶ 1 12 06

When I press on run and debug, it asks me to choose from a list.

S1 Speaker 1 ▶ 1 12 12

Yes, choose Python.

S1 Speaker 14 ▶ 1 12 14

It's not one of them, it's no JS or web app. Chrome web app Edge install extension,

S1 Speaker 1 ▶ 1 12 24

Install ex. Okay, share your screen really quick. Okay. Okay, go to the squares in the extensions.

S1 Speaker 14 ▶ 1 12 44

Oh sorry,

S1 Speaker 1 ▶ 1 12 45

Don't worry. No, not there. Into the squares.

s1 Speaker 14 ▷ 1 12 52

This? Yes.

s1 Speaker 1 ▷ 1 12 53

Here. Yep. Type Python. Oh, is there? Well yes. Install this Python. The first one. The

s1 Speaker 14 ▷ 1 13 05

First one?

s1 Speaker 1 ▷ 1 13 06

Yeah. And that's the only thing that you needed to do. Now go and run. Try to run it again

s1 Speaker 14 ▷ 1 13 22

From here?

s1 Speaker 1 ▷ 1 13 24

Yes.

s1 Speaker 14 ▷ 1 13 27

Yeah. Okay fine. This one right? Yeah.

s1 Speaker 1 ▷ 1 13 29

Okay.

s1 Speaker 14 ▷ 1 13 30

Thank you. Hi.

s8 Speaker 8 ▷ 1 14 51

Hi. Silly question. What? What activity are we working on?

s1 Speaker 1 ▷ 1 15 03

Sorry, what was that?

s8 Speaker 8 ▷ 1 15 04

What activity are we working on?

s1 Speaker 1 ▷ 1 15 08

We are working on Activity four, number four, S T U. Hello? Variable world.

s8 Speaker 8 ▶ 1 15 16

Okay. Yes. Awesome, awesome. I got that resolved. Thank you.

s1 Speaker 1 ▶ 1 15 23

Perfect. No problem.

s1 Speaker 14 ▶ 1 15 26

So it's still asking me to choose whether I wanna create a Python file or environment or run and debug your Python file or explore more resources,

s1 Speaker 1 ▶ 1 15 37

Run and debug your Python file.

s1 Speaker 14 ▶ 1 15 40

Okay, thank you.

s1 Speaker 1 ▶ 1 15 42

No problem. Please let me know if that works.

s1 Speaker 14 ▶ 1 15 46

Okay.

s1 Speaker 1 ▶ 1 15 59

If you want guys, after the class in office hours, I give you, you stay, I give you half an hour off. Be a walkthrough. Sounds good. You, I think we could all use that. Okay, perfect.

s1 Speaker 17 ▶ 1 16 29

One quick question. You go, so the read me file, when I click on that it, it opens into the same terminal, right? So if I want to open it into a new tab, how can I do it?

s1 Speaker 1 ▶ 1 16 43

Just,

S1 Speaker 17 ▷ 1 16 44

Yeah, I tried, I tried that.

S1 Speaker 1 ▷ 1 16 46

Drag it to the site, that's it.

S1 Speaker 17 ▷ 1 16 52

It actually creates a file on my desktop if I drag it and drop it there. But yeah, let me

S1 Speaker 1 ▷ 1 16 58

Check that again. Try to share your screen and I can try to help you there.

S1 Speaker 17 ▷ 1 17 05

How

S8 Speaker 8 ▷ 1 17 07

Can I refresh the VS code?

S1 Speaker 1 ▷ 1 17 12

How can you refresh the VS code? How is that like you want to close all the folders?

S8 Speaker 8 ▷ 1 17 21

No, just refresh because I had to like, I guess the reason why is because I had an extension was having like this effect when I, like when I was typing, so that was getting very lethargic, my typing, but it got to a point they couldn't write. So I uninstall it and disable it. But it says that I have to reload it and I did and I still, now I can't write type again.

S1 Speaker 1 ▷ 1 17 45

Okay. Then it is, to be honest, I don't know, I have to find out how can you reload via scroll?

S8 Speaker 8 ▷ 1 17 52

Ok,

S1 Speaker 1 ▶ 1 17 53

Yeah, that works.

S1 Speaker 17 ▶ 1 17 57

No, so if I want to have this file, if I, if I click it here, it replaces the one that's already in there open. So

S1 Speaker 1 ▶ 1 18 08

Yes, just grab the file and just put it to the next, not, not there.

S1 Speaker 17 ▶ 1 18 13

Oh, here.

S1 Speaker 1 ▶ 1 18 13

Yeah.

S1 Speaker 17 ▶ 1 18 18

No

S1 Speaker 1 ▶ 1 18 22

That's a Windows, right?

S1 Speaker 17 ▶ 1 18 23

Yeah, that's Windows.

S1 Speaker 1 ▶ 1 18 26

Okay. Try to open your entire window. Oh cause you're trying to open two files with the same name. Try another one.

S1 Speaker 17 ▶ 1 18 40

Okay. Yeah.

S1 Speaker 1 ▶ 1 18 45

Yes,

S1 Speaker 17 ▶ 1 18 45

But but then this are like in a same window. There are three tabs, right?  
What I meant is if I can just like play around and create one more window



here.

s1 Speaker 1 ▷ 1 18 57

Oh yes. You can open another BS code.

s1 Speaker 17 ▷ 1 19 02

So I have to go to my file and then open it, right?

s1 Speaker 1 ▷ 1 19 07

No, go. Go to your applications in the open BS code again.

s1 Speaker 17 ▷ 1 19 15

Oh, okay. From

s1 Speaker 1 ▷ 1 19 16

From scratch. Yeah. Yeah, exactly.

s1 Speaker 17 ▷ 1 19 19

Okay. Okay. Got it. And then I can play around here. Okay, got

s1 Speaker 1 ▷ 1 19 22

It. Open as many. Yeah, yeah.

s1 Speaker 17 ▷ 1 19 24

Thank you.

s1 Speaker 1 ▷ 1 19 25

No problem.

s1 Speaker 14 ▷ 1 19 30

Can I share my screen?

s1 Speaker 1 ▷ 1 19 32

Yes. Me.

s1 Speaker 14 ▷ 1 19 34

Okay, so here,

s1 Speaker 1 ▷ 1 19 48

Can you make that big?

s1 Speaker 14 ▷ 1 19 51

Did I make what?

s1 Speaker 1 ▷ 1 19 53

That screen bigger? Yes. Terry, we are in Hello variable wall.

s1 Speaker 14 ▷ 1 19 59

Okay, so what, what do I do then?

s1 Speaker 1 ▷ 1 20 02

Make the window bigger just to see.

s1 Speaker 14 ▷ 1 20 05

Okay,

s1 Speaker 1 ▷ 1 20 06

Yeah, sorry for that. Yeah, perfect. Okay, so go to the five that you want to run first.

s1 Speaker 14 ▷ 1 20 16

From where?

s1 Speaker 1 ▷ 1 20 17

Explorer go up, up, up, up, up.

s1 Speaker 14 ▷ 1 20 20

Oh, here. Okay. Classwork.

s1 Speaker 1 ▷ 1 20 22

Classwork. Yeah.

s1 Speaker 14 ▷ 1 20 26

Oh,

s1 Speaker 1 ▷ 1 20 27

Where do you have your Python?

s1 Speaker 14 ▷ 1 20 29

Here.

S1 Speaker 1 ▷ 1 20 30

Okay, go there. Lesson plan.

S1 Speaker 14 ▷ 1 20 34

Yeah,

S1 Speaker 1 ▷ 1 20 35

Python number one, two activities I guess. And then you working in number four? Number one,

S1 Speaker 14 ▷ 1 20 44

Four.

S1 Speaker 1 ▷ 1 20 44

Yeah. And so version. Okay, there. Stop there.

S1 Speaker 14 ▷ 1 20 52

Okay. Can

S1 Speaker 1 ▷ 1 20 53

You see up there that you have plus file over there? Like plus create a new file where it says desktop. You see where it says Yeah, that's a folder. Go to the left. Yeah, create a new file. Perfect. Name the Python file, whatever name you want to put it. Enter then print something.

S1 Speaker 14 ▷ 1 21 37

Oh, okay. Okay. Now I can start working.

S1 Speaker 1 ▷ 1 21 42

Yes.

S1 Speaker 14 ▷ 1 21 43

Okay. Okay. Thank you.

S1 Speaker 1 ▷ 1 21 46

No problem. Yeah. Okay, I'm gonna start sharing my screen.

S1 Speaker 12 ▷ 1 21 57

One question.

S1 Speaker 1 ▷ 1 21 59

Yes, Jay,

S1 Speaker 12 ▷ 1 22 00

Can I quickly share my screen? It's just a small question.

S1 Speaker 1 ▷ 1 22 03

Yes,

S1 Speaker 12 ▷ 1 22 04

Thank you. Sorry about that. Okay, can you see my screen?

S1 Speaker 1 ▷ 1 22 11

Yes,

S1 Speaker 12 ▷ 1 22 13

I, I think I've gotten the program right? I'm guessing I just, can we not have all of this show up and just show the output? I do this, will it work?

S1 Speaker 1 ▷ 1 22 32

Well it's showing an error now. Yes you can if you want on office hours, I can give you some, I can give you all guys some tips about how to navigate better through the terminal.

S1 Speaker 12 ▷ 1 22 44

Okay. Alright, thank you.

S1 Speaker 1 ▷ 1 22 45

No problem. So I'm gonna share my screen now and we are going to see the answer of that activity. Hello world.

S1 Speaker 1 ▷ 1 23 02

Hello, variable word pie. So create a variable called name that holds a string name equal. Jacob Deming. Create a variable called country that holds a string country United States an integer and another integer and calculate something to calculate it. Basically you just have to put call the variable. Then times eight, that intu intuitive is python, create a variable

satisfied that holds a volume. Then printing, print out, hello, hello, my name, blah, blah, blah. You live in. And then the Fs, the F string basically is the F first. Then you open quotes and then you put whatever string you want to put. And in order to put any variables inside, you just use quarterly brackets. Okay. And that's it. To run these, I am going to run it with my GIAs. Okay. Because we are learning GIAs, not bsco. So to run this, I am going to actually go to CV and then this symbol I'm going to home

s1 Speaker 10 ▷ 1 24 19

Then. Sorry Hugo, I have a question.

s1 Speaker 1 ▷ 1 24 22

Yes?

s1 Speaker 10 ▷ 1 24 22

Yeah. So what's wrong with the, what's the difference with the string and the F string?

s1 Speaker 1 ▷ 1 24 28

The difference is that with the string, for example H This one, yeah, it's an integer, right? And for me to concatenate this integer with another integer, I have to put this word, s t r, I have to call this function and then I convert my integer into a string because if I don't do this, look, it's gonna give me an error. Okay? It's gonna tell me something's wrong here. I cannot concatenate a string and an integer. So that's why I will use my sdr R here to convert it. The difference with the F is with nf, you don't have to use s str, you just have to put in quarterly brackets, whatever variable you want to use It doesn't matter if it's a bullion, if it's an integer, if it's a double, if it's a lung, if it's a whatever. Okay. That's the difference. Yes.

s1 Speaker 12 ▷ 1 25 31

So you don't need to put Colin before you put the variable name in curly braces.

s1 Speaker 1 ▷ 1 25 38

Exactly.

s1 Speaker 12 ▷ 1 25 39

Because in the previous one you had a column before you put the curly braces. So I was not sure if you needed that.

s1 Speaker 1 ▷ 1 25 48

Oh no, that was part of the strings.

s1 Speaker 12 ▷ 1 25 51

Okay, got it. Thank you.

s1 Speaker 1 ▷ 1 25 53

No problem. Okay, and so I'm gonna run it here, here. Where is it? I can go here. Yes. Perfect. I'm in sync. Let see, I'm going users and here I'm going here. I am going to date that database plans. Here I am going to, I don't know, but this is zero one or something like that. Lesson plans. Then I'm going to zero three with Python. Then I'm going to 0 1, 0. This one I'm mature. So yes, I'm gonna one. Yes. And here I am going to activities. Yes. Then here I am going to see the zero four. Then here I'm going to open the SO version. In the so version. I am going to actually, what's the name? See the, and now I can run the Python.

s1 Speaker 17 ▷ 1 27 19

Are you sharing something?

s1 Speaker 1 ▷ 1 27 23

Yes. You can see my gi

s1 Speaker 17 ▷ 1 27 25

No, no, we can see the video.

s1 Speaker 1 ▷ 1 27 28

So yeah. Oh, maybe yes. No, no GitHub, no ash. Oh, okay. Let me then share my screen. Okay. I wasn't doing anything like really important. I was just telling you that I am actually running this, this thing not in visual. Okay. Because I don't want to confuse you here. I type my code in Visual Studio, but I'm gonna run it in Giba. Okay. So I was just navigating the through my folder where I have my hello variable Python. So I just arrive

to that folder and in that folder I just need to put like Python, hello variable pie and it's gonna run. Okay. So the same thing, you just have to navigate to that folder where you have your, your file pie and just put Python and then the name of the file to

s8 Speaker 8 ▶ 1 28 32

Well, so yeah, on that. So you have to find first where your folder is and once you find it you go LS or how do you or CD on that? How do you enter to that file again?

s1 Speaker 1 ▶ 1 28 49

Yeah, exactly. So basically with CD you change directories. Yes. So for example, here I can change my directories. I ls here because I don't know what is inside here. So in LS I can see that it's a so and so version. I go to the so version here, I go to LS to see what is inside my so version and then I can run my hello variable pie

s8 Speaker 8 ▶ 1 29 12

And how do you run it?

s1 Speaker 1 ▶ 1 29 14

Python and then hello variable pie

s8 Speaker 8 ▶ 1 29 19

And just enter. Okay. And I have another very silly question. When you put the curly bracket, why are you the curly brackets in daily wage? Again,

s1 Speaker 1 ▶ 1 29 29

Because this is an F string, the F string function allows you to put whatever variable you want to put in a string. The curly brackets is for that.

s8 Speaker 8 ▶ 1 29 39

Okay. And the F is also an string. What, what is, I'm sorry, and I think you already mentioned this, but I missed what the F is different from the string. What is it? The difference

s1 Speaker 1 ▶ 1 29 51

The string, in order for you to concatenate strings and integers, you have to put `SSTR` to convert whatever other variable types you have here to concatenate the string. And here you don't, you just put everything together and whatever variable you want to use, you just put it in curly brackets.

s8 Speaker 8 ▶ 1 30 11

Oh, okay, I see. So that's only like just basically `MA` putting text and the `F` string is just putting like text to it? Yep. Like a `bullion`. But I cannot use the `F` string with numbers in `bullions`.

s1 Speaker 1 ▶ 1 30 33

`F` strings, you're gonna be able to use whatever type of variables. Doesn't matter if it's a `bullion`, if it's an `interior`, if it's whatever. But you just have to put it in curly brackets.

s8 Speaker 8 ▶ 1 30 45

Okay. So basically the `xr`, I can just mix any type of data.

s1 Speaker 1 ▶ 1 30 49

Exactly.

s8 Speaker 8 ▶ 1 30 50

Got it. Thank you.

s1 Speaker 1 ▶ 1 30 51

No problem. Okay, so I am going to the next, next activity of today's class. I'm gonna save, this is prompts. Okay. And here, for example, we are going to ask the user some questions. In this case, in order to ask the user questions, I am going to use my input function. Okay. So my variable name is gonna be equal to my input. And the question that that input is waiting is what is your name? So you can type your name. Then you're gonna collect the user's input for the prompt. And then you're gonna put again `int`, because we are going to pass on string or we want to convert that to an integer, I am going to find a variable that it's `H` equal to `int` input. How old are you? Okay. Then a `bullion`, the same thing



instead of in I am going to collect is the input true or false. Okay. After that I am going to call my three variables printing. My name is this one. I will be h plus one next year. That's why I am converting this into an interior because I can do an operation inside here and the input was converted two and then two or false. Okay. Is input thought?

S1 Speaker 1 ▶ 1 32 28

Yes. Okay. Let me run this. Any questions about the call prompts? No. Perfect. Let me run this. So I'm gonna go to the cd, go back to, I'm going see the, go back to, can you see my giba now? Yeah. Okay. I'm in activities, so I'm gonna click Ls, cd, I am going to zero five prompts and then Ls to see what is inside cd, the soft version here. And then LS and I can run my input. So I'm gonna go Python input, stop by, what's your name? Ki How old are you? I dunno. 23 is the input truth. Tru, the input was converted to true. Okay, so my name is Hugo. I will be 24 next year. And the input was converted to true. So basically because it is doing the other way around, right? It's going to, it is gonna my bullion, if I put false, it is gonna put true. If I put true, it's gonna put false. Okay. It's gonna convert it. Yes.

S1 Speaker 17 ▶ 1 33 58

One silly question, sorry if I might have missed, missed it out, but when we can run the file here, why are we using GI P to do it? I mean there are a lot of steps involved to run the file, right? So why? Why do we do that exactly

S1 Speaker 1 ▶ 1 34 16

Here we're doing it for educational purposes, purposes project. Okay. Because in this specific part, I am not teaching how to use BS code, but I am teaching right now how to use, what to do in Python, how to run your Python code. Okay. Because I have so many people having problems running the code. So we are doing it the way that we saw it at the beginning.

S1 Speaker 17 ▶ 1 34 43

Okay. Yeah, because it's, it's not at all an optimized way to run a course. Exactly. I mean there are many steps involved to go it to the director.

S1 Speaker 1 ▶ 1 34 52

So Exactly. You used to keep there and you see the one in here. Yeah, exactly. Okay, cool. Thank you. No problem. Any other questions?

S1 Speaker 1 ▶ 1 35 06

No. Okay, perfect. So what if, just, what if guys, if we do the next activity together, what do you think about that? Let's open the activity that says a student down to input. Let's see the read the readme say create to the different variables that will take the input of your first name and your neighbor's first name. Create two more inputs that will ask you how months each of you have been coding. Finally, display a result with both your name and the total amount of months code. Is that clear? So do you think we can do it together? Yes. What do you Yes, yes. Let's do it together then I'm gonna open my own. So version here, I'm going to create a new file here that it's gonna call my, what's the name of this? To down to. Ok. Did you,

S8 Speaker 8 ▶ 1 36 34

So sorry. Did you put that in in, how do you create it again? Sorry, I missed that

S1 Speaker 1 ▶ 1 36 39

Part. I go to unsolved here and then I just click here.

S8 Speaker 8 ▶ 1 36 42

Oh, okay. Thank you.

S1 Speaker 1 ▶ 1 36 45

Now I'm gonna use my steps here. Well, no, I'm gonna, I'm gonna create my comments because I'm used to create my comments here. The first one is I'm gonna take the input, input of of name and my name and my, my name neighbor, right? So here I'm gonna put my name, it's gonna be equal to input and then the question, right? What is your main space? That's it. I don't have to put anything like this. Com. Alright, then my neighbor. Neighbor is gonna be the input of what is your neighbor, right? All good here guys, if you have any questions, just let me know, okay?

S1 Speaker 10 ▶ 1 38 04

Questions. So while you just put name, it's as name, but so while you only pay, play an e i g as neighbor. So it's just for default they will just recognize guys name and e i g as neighbor or

S1 Speaker 1 ▶ 1 38 19

Yes, I can put whatever name of my variables. To be honest, I just think of one of the variable name and I just use it. But I can put whatever name, right? Whatever you want to identify here, just put it okay. Yes, no. Okay, perfect. So the other part, it says create two more inputs that will ask how many months, how long you have been coding, right? So how long you have been coding. So I'm gonna put months rolling. Oh is one thing here in the variables. Guys, you cannot define a variable with spaces, okay? It's gonna give you an error. So you have to put like for example, these I think do notation is not even accepted. So it just, this notation is accepted. Or if you put first just like this as well, it's accepted month coding, right? So just be careful with that. So here is, I'm gonna put input and my question is going to be how many months have you been coding

S1 Speaker 12 ▶ 1 39 56

Hugo?

S1 Speaker 1 ▶ 1 39 58

Yes. Jay.

S1 Speaker 12 ▶ 1 39 59

For the coding duration, do you have to specify the variable type or you just take the input? As As and when? As how we've, how the user has given it. Like for example, if I give an in integer because it's the number of months, it automatically take it as in integer. It doesn't ha, I don't have to specify anything speci for that. Right?

S1 Speaker 1 ▶ 1 40 25

Exactly. Ok, right.

S1 Speaker 12 ▶ 1 40 27

Thank

s1 Speaker 1 ▷ 1 40 28

You. Yes, here for the moment. I, I am, I haven't specified if it's an integer, if it's anything right, I can change that whenever I'm doing the printing.

s1 Speaker 12 ▷ 1 40 38

Got it. Thank you.

s1 Speaker 1 ▷ 1 40 39

Yeah, perfect. You go? Yes.

s3 Speaker 3 ▷ 1 40 43

Can I ask a silly question? Can we take multiple inputs from the user? Like instead of asking two twice, can you ask

s1 Speaker 1 ▷ 1 40 52

Like once and take two inputs from the user? Yes, but you have to embed the input here and it, it will be just one variable. So if you want to use the variables separately, like in this activity suggesting you are not going to be able to do it. So that's why we do two inputs. Okay. Okay. Thank you. No problem. We have that part done, no errors so far. So it says finally display a result with both your name and the total amount of coding, right? So I'm gonna create another variable that says at total months and I'm gonna be here total equal to, and here here's gonna be my gear because I want to transfer everything in gear and I'm gonna put, you see how it's gonna bring everything that I'm using right now. So I just can click it here, then I can do a plus here, another ink and then I can put my neighbors.

s1 Speaker 12 ▷ 1 42 18

So is the input taking the value that you're giving it as a default string?

s1 Speaker 1 ▷ 1 42 24

Yes.

s1 Speaker 12 ▷ 1 42 25

Okay, got it. Thank you.

S1 Speaker 1 ▶ 1 42 26

No problem. That's my total months between both of us. And the other part is do the printing, print, whatever we want to print. Print. And then I can put like I am, I am perfect plus here my name and then I can put here plus and then put and my, I have to put in space here so it look much better. And my knee, sorry, I just cannot put it like past this. Then I put a plus here and then I put the nail for this, right? I'm actually gonna go, no, it's not gonna this print. And then I'm gonna put our total mount of holding this and then I'm with the plus because I am returning an here I have to convert it to stream. You see that? So, so and then gonna put here months,

S1 Speaker 12 ▶ 1 43 55

Can't we just use F

S1 Speaker 1 ▶ 1 43 57

Print? Yes, we can use F as well. Good. How will we, how will we do, how will we do it in with Fs?

S1 Speaker 12 ▶ 1 44 07

It's pretty simple. I mean I've, so you basically just do print F.

S1 Speaker 1 ▶ 1 44 12

Okay, let me do it.

S1 Speaker 12 ▶ 1 44 14

Yeah,

S1 Speaker 1 ▶ 1 44 16

Print.

S1 Speaker 12 ▶ 1 44 18

Yeah, you start with F then your, okay the, let me just describe what I've written so that it's easier. So I just did print F your braces, I mean your quotation marks are gonna start. I just said your name, like the input that you've taken and the neighbor's name in curly braces because their

variables have been coding for curly braces. The total, total duration and months.

s1 Speaker 1 ▶ 1 44 47

Oh, perfect. Yes. Curly braces every variable, right?

s1 Speaker 12 ▶ 1 44 51

Yeah, all the variables in curly braces and then add in whatever you want to type for the print statement.

s1 Speaker 1 ▶ 1 44 58

Perfect. That will work.

s8 Speaker 8 ▶ 1 45 00

Do you have to still put Yeah, the string because whole point, just not to put the string right?

s1 Speaker 1 ▶ 1 45 05

Sorry, sorry, can you repeat that Denise? Denise

s8 Speaker 8 ▶ 1 45 07

Please. Like once that you have the F, you don't need to put the string, you just put the quarterly brackets and the F at the beginning. Okay, got it. That's much easier. Got it.

s1 Speaker 1 ▶ 1 45 16

And that's it. That's the activity that's run these

s1 Speaker 18 ▶ 1 45 19

I think we don't need, I think we don't need the plus signs with

s1 Speaker 12 ▶ 1 45 26

The

s1 Speaker 1 ▶ 1 45 27

S. Oh yeah, sorry,

s8 Speaker 8 ▶ 1 45 30

We don't need them.

s1 Speaker 1 ▷ 1 45 31

No,

s1 Speaker 18 ▷ 1 45 31

You're And the quotes and we don't need the quotes. Yeah, I think,

s1 Speaker 1 ▷ 1 45 36

No, no, yes, that's right. Yes.

s8 Speaker 8 ▷ 1 45 39

Okay. So the The plus sign, it's needed when there's no strings in there and

s1 Speaker 1 ▷ 1 45 45

When there is not F string.

s8 Speaker 8 ▷ 1 45 48

Oh, when there is no F string. Okay. But the spaces still need to be there, right?

s1 Speaker 1 ▷ 1 45 57

Exactly.

s8 Speaker 8 ▷ 1 46 11

My F string is not working for some reason.

s1 Speaker 1 ▷ 1 46 15

Sorry,

s8 Speaker 8 ▷ 1 46 16

My FST stringing.

s1 Speaker 1 ▷ 1 46 19

He's not working.

s8 Speaker 8 ▷ 1 46 21

No, let's see.

s1 Speaker 1 ▷ 1 46 22

We can check. We can check on that Denise in the break. Okay, so you can share your screen and then we can check that. Okay.

s1 Speaker 12 ▷ 1 46 31

Denise, I've, I've pasted the print statement that I put in. You can try that and see if that's working or we can.

s8 Speaker 8 ▷ 1 46 37

That's the one that I got, but I know why because I left out some quotation marks. My bad.

s1 Speaker 12 ▷ 1 46 43

Got it.

s1 Speaker 1 ▷ 1 46 44

Cool. Any questions guys, about this activity?

s3 Speaker 3 ▷ 1 46 48

You what? If the user enters months in decimals, will it take for example like 2.5 months and 7.5 months? So will it just round, round about it or how would it print it?

s1 Speaker 1 ▷ 1 47 04

That's the way error.

s8 Speaker 8 ▷ 1 47 17

Oh

s1 Speaker 1 ▷ 1 47 20

Yes. Basically I'm defining an integer. I have to define a double

s8 Speaker 8 ▷ 1 47 27

For me worked when I put, how many months have you been coding? I put 0.5, 0.5 months and it went back, it went to the next question, but I put 4.2 for the next for the neighbor. And it showed an error.

s1 Speaker 1 ▷ 1 47 41

Yes, cause



s8 Speaker 8 ▶ 1 47 42

But the first one worked.

s1 Speaker 1 ▶ 1 47 46

I mean no, in order for the whole code to work, you have to get all the printing right. And we, because you were using integers is only looking for integers. Okay. That's why when we put doubles, he's giving us this error with base 10. You see?

s8 Speaker 8 ▶ 1 48 04

So if we put doubles, how do we, how do you put doubles instead of integer

s1 Speaker 1 ▶ 1 48 10

Doubles instead of intes? She just put it like, lemme here

s8 Speaker 8 ▶ 1 48 16

Does the doubles also takes negative numbers.

s1 Speaker 1 ▶ 1 48 20

I don't think it's double. She's what's the name of this type of variable? It's isn't it called float? Float,

s3 Speaker 3 ▶ 1 48 26

Float,

s8 Speaker 8 ▶ 1 48 27

Float. Oh yeah, float says does flow stick also negative numbers?

s1 Speaker 1 ▶ 1 48 35

No, I don't think so.

s8 Speaker 8 ▶ 1 48 39

Is there any type of data that can take decimals negatives or, or positive? Because if we don't know what the input is going to be, how can we create a data type that can work with

s1 Speaker 1 ▶ 1 48 57

Your number? Well when you're creating an application, for example, when you decide to select, to gather data and then you decide to go only integer without decimals, you can put an error, you can handle that exception. So if someone puts a negative number, if someone's put a float, if someone's put something different there, then you put, you have to select an integer. So floats can be negative though in python

s8 Speaker 8 ▶ 1 49 27

Floats. Oh, awesome. And and so for the, how many months have you been coding? If I put seven months, would that also take, cuz it's a, it is a float. So that will mark an error, right? If the user put decide to put seven months or one year,

s1 Speaker 1 ▶ 1 49 47

Yes it's gonna be an error because it's waiting for just integers or floats in this case.

s8 Speaker 8 ▶ 1 49 53

And there's another data type that can do numbers in like variance, maybe variance You could,

s1 Speaker 19 ▶ 1 50 01

You could just save it as a string and it'll still print out an integer.

s1 Speaker 1 ▶ 1 50 05

Exactly.

s8 Speaker 8 ▶ 1 50 06

Okay, cool. Thank you.

s1 Speaker 1 ▶ 1 50 08

No problem. Any other questions here?

s1 Speaker 1 ▶ 1 50 18

Okay, perfect guys, so let's do the last activity. So we can go to break. See this one second to save it. So we're gonna talk about conditionals. Conditionals pretty much the same as we saw actually in bba. Okay, so here we have, we're gonna find two integrators X and y X is one, Y is 10.

So to check conditionals here we're gonna use it double equal. Okay? This is how you write it in Python. So if X equal equal one two points, then you're gonna print X is equal to one. If why it's not equal to one, you use this symbol, then you're gonna print this. Okay, here you're gonna check if one value is less than the other one with this symbol, okay? Less than, okay, you're gonna check if it's greater than with the crocodile symbol the other way then you're gonna check if it's less than or equal to another. You just put it like the crocodile plus the equal.

s1 Speaker 1 ▶ 1 51 47

You can use the operators with the end. Remember the end you have to meet both conditions. You can put two conditions with or as well. You, you can meet this one or this one, okay? And you can do next step if statements. So if X is less than 10 and then inside if Y is less than five, then you're gonna print this. X is less than 10 and Y is less than five L. If here you are not using L, if you're using lif y it's equal five, then you're gonna put X is less than 10 and it's equal to five. L X is and than 10 and Y is greater than five. Okay. Any questions about these guys?

s2 Speaker 20 ▶ 1 52 52

Hey Hugo, unlike vba, we don't have to end if, right? This is just

s1 Speaker 1 ▶ 1 53 00

Exactly if

s2 Speaker 20 ▶ 1 53 01

We can move on to the next if statement.

s1 Speaker 1 ▶ 1 53 03

Yes. And one important thing here guys, indentation matters in Python, indentation matter. If I put this here, it's gonna give me an error, okay? But in visual studio code, when you do it like this, it's gonna indent automatically for you. But indentation matters in Python, okay? You don't have to close the sentence, you don't have to close like in JavaScript like this. Alright, any other questions guys? No. Okay, so let's actually take a break of 50 minutes and then we can come back to the next exercise.

Okay? So let's take a break and I dunno if you have questions that you want to share your screen or something like that. I guess

s8 Speaker 8 ▶ 1 54 02

I do, yes ma'am. And if I can get a breakout room cuz I'm running every time that I open a new file for different exercise activity, I, I can't write type again.

s1 Speaker 1 ▶ 1 54 15

Okay. If you want to go to a breakout room, I think, let me check who's in breakout rooms right now? Nobody.

s8 Speaker 8 ▶ 1 54 22

Yeah, they're like

s1 Speaker 1 ▶ 1 54 24

Multiple people. I dunno. I mean guys, if you want to go to break and I can stay here with Denise and you can share your screen Denise.

s8 Speaker 8 ▶ 1 54 33

Sure. Thank you. All right, there you go. So for some reason, and I think I spoke this with Andrew, remember Andrew, that we were trying to check in our hello world and it was working fine cuz I was like I just wanna make sure the VS code works runs good and if I tried to type, see like it takes now I can type but it's, it's getting stock every time that I switch that I take an activity, let me see if I open this activity for example, it's, oh now it's working. Is there any way to debug this when this happened? Because I've been running with that issue since I started.

s9 Speaker 9 ▶ 1 55 36

I think it, I think it might be the extension that you have. Cause it the way it's like highlighting,

s8 Speaker 8 ▶ 1 55 43

I already actually un

s9 Speaker 9 ▶ 1 55 46

Uninstalled it,

s8 Speaker 8 ▶ 1 55 46

Installed it and ha I un unabled it and uninstalled that and closed vsco closed my terminal, quit via code and reopen it. And I still running on that. So I'm wondering if there is any remaining on that from that extension that might be triggering my type, my typing because every single activity, I guess like when I was writing more code, if I switch to a different activity it will, it will give me the same issue that I can't type for a while.

s9 Speaker 9 ▶ 1 56 27

Hmm.

s8 Speaker 8 ▶ 1 56 29

Is, is there any way to see if there is any file or remaining of that extension that I can remove?

s1 Speaker 1 ▶ 1 56 40

Well the, I think let me guys, if you think something different, the easiest, the easiest part or the easiest thing that you can do is to delete BS code and try to install it again from scratch and, and just in install the, the extension of Python and maybe that fix all the errors that you have.

s9 Speaker 9 ▶ 1 57 10

Yeah, I was trying to see if there was a way also to just like reset settings of BS code just to,

s8 Speaker 8 ▶ 1 57 19

Yeah, I was trying to find that and I found this that to refresh I need to go there from the window menu, excuse my dog and then click on refresh button in the toolbar, but I can't see that button here.

s1 Speaker 11 ▶ 1 57 39

Yeah, it looks like it's going into insert mode automatically, which there might be a way to changing that.

s8 Speaker 8 ▶ 1 57 46

This one?

s1 Speaker 11 ▶ 1 57 48

Oh no, in your VS code.

s9 Speaker 9 ▶ 1 57 50

Oh yeah, I

s1 Speaker 11 ▶ 1 57 51

Think you're right because that that little white box is what looks like when it's in insert mode versus the other mode. There's

s8 Speaker 8 ▶ 1 57 59

One. Yeah. Do

s9 Speaker 9 ▶ 1 57 59

You have a, do you have a lock on your keyboard with insert? Cause some keyboards will have an im.

s8 Speaker 8 ▶ 1 58 09

No. Yeah, no. So you mean that when that insert is there is because my keyboard is, is locked?

s1 Speaker 11 ▶ 1 58 21

Not necessarily, there's just diff two different modes that are, it's like insert versus non insert mode when you have the little white box. So it's not gonna let you type. And there are, I mean there's a few things that could cause that, that we could look at. Like the Vim extension can cause issues if you wanna remove the VA extinction,

s8 Speaker 8 ▶ 1 58 44

The van pending command kit or this command line.

s1 Speaker 11 ▶ 1 58 48

Yeah, so Vim can cause I issues when you, when you you're in, in here. I'm just trying to, I'll have, I'd have to look in into it myself.

s8 Speaker 8 ▶ 1 59 00

Okay.

s1 Speaker 11 ▷ 1 59 00

But that's one of the common things that causes it to keep going into your, into the different modes, like insert or overwrite mode.

s8 Speaker 8 ▷ 1 59 10

Oh yeah, I do have an insert. I found it in my, in my keyboard. It looks

s1 Speaker 11 ▷ 1 59 14

Like it's going into overwrite mode automatically. Like switch to another one, let's go switch to a new Python file and so it's normal mode and go try to type something. Oh, and that, that's, you know, you want it to go to No, it's, you can see once it starts working, the the mode will switch down at the bottom.

s8 Speaker 8 ▷ 1 59 46

Oh yeah, yeah. And I was typing, not deleting.

s1 Speaker 11 ▷ 1 59 50

Yeah, so you want it to go to the right mode automatically.

s8 Speaker 8 ▷ 1 59 58

I

s1 Speaker 11 ▷ 1 59 58

Found go to your extension, well go to your extensions. So yeah, there, do you have Vim installed in as an extension?

s8 Speaker 8 ▷ 2 00 12

Yeah, this one

s1 Speaker 11 ▷ 2 00 14

We don't really, I mean Vims not the friendliest thing to use. You might wanna disable that and see if it fixes the problem.

s8 Speaker 8 ▷ 2 00 21

Should I un unin

s1 Speaker 11 ▷ 2 00 23

Just disable it for now? I mean you can, but now, now that it's disabled, you can see if the, if this changes the behavior, go find something else and open it up or, yeah,

s8 Speaker 8 ▶ 2 00 37

Nope.

s1 Speaker 11 ▶ 2 00 38

Well you have to go wait, it's when you open the file.

s8 Speaker 8 ▶ 2 00 42

See, but I'm trying to type right now and it goes hit

s1 Speaker 11 ▶ 2 00 44

The insert button. The insert button,

s8 Speaker 8 ▶ 2 00 48

Yeah. Should I press it?

s1 Speaker 11 ▶ 2 00 52

Yeah, I guess you can't switch modes like that but Oh yeah, we want, we want we just Google it.

s1 Speaker 18 ▶ 2 01 02

If it's in insert mode it says to try escape key to go to go to normal mode.

s1 Speaker 11 ▶ 2 01 15

Okay,

s8 Speaker 8 ▶ 2 01 16

I escaped and it's not working,

s1 Speaker 11 ▶ 2 01 19

It's working there.

s8 Speaker 8 ▶ 2 01 20

No, but it's, it's not even

s1 Speaker 11 ▶ 2 01 22



That's in visual mode. Disabled this Vim is, yeah, VIM is still, yeah, disabled the Vim stuff. Okay cuz Vims just hard to work with

s8 Speaker 8 ▶ 2 01 44

But still I can't like I would just

s1 Speaker 11 ▶ 2 01 46

Try opening up a new file after you disabled it. The extension.

s8 Speaker 8 ▶ 2 01 50

I'm just gonna close all this and I wanna close vs. Code quick.

s1 Speaker 11 ▶ 2 01 58

I mean it could be something else, but that's what usually what causes it is, is Vims very not fun to work with. It's not very user-friendly and it's treating it like a Vim editor.

s8 Speaker 8 ▶ 2 02 08

Okay? So it works now.

s1 Speaker 11 ▶ 2 02 12

Okay and let's open up another one.

s8 Speaker 8 ▶ 2 02 14

I read a bunch of stuff, it's what happened is when I start to write more stuff and then, okay, so this is number, this is Rock proper scissors

s1 Speaker 11 ▶ 2 02 26

Here go go to one of the solved folders or on instructor list, anything you haven't opened up before

s8 Speaker 8 ▶ 2 02 35

Unsolved this conditional. Okay,

s1 Speaker 11 ▶ 2 02 38

There we go. So vi's not fun, vin's not friendly.

s8 Speaker 8 ▶ 2 02 43

So now it should show that that was because of vin, right?

S1 Speaker 11 ▶ 2 02 47

Yeah, it's treating it like a Vim editor and, and it's so not friendly. Not many people use it but that should fix the problem.

S8 Speaker 8 ▶ 2 02 54

Thank you guys so much. I was panicking for a second. Thank you.

S1 Speaker 1 ▶ 2 08 29

So perfect guys breaks up. Okay and let's, let's do the next activity of today's class. So what we are gonna do is actually you're gonna start doing the S T U conditional conundrum, okay? It's activity number eight, okay? And if you open it, you're gonna follow the instructions. It's quite straightforward what you have to do here. And in this activity we have actually, let me tell you how much we have. We have this activity, 10 minutes, actually 10 minutes and then we can check and then we can go to lists. Okay? So 10 minutes starting now. Please start working this activity again. Breakout rooms are open. If you have questions you can jump right in, intro breakout room or you can ask it here. Okay? Please start working on this and then we can check together.

S1 Speaker 1 ▶ 2 18 56

Perfect guys. Time's up. I'm gonna start sharing my screen with the solve activity. Can you see my screen? Thumbs up. Perfect, thank you very much for that. Okay, so for this, oh, need some work. So basically I define here my  $X$  equal five. Then if two times  $X$  is bigger than 10, then print question one. Works else print O needs some work. Okay, here is an arithmetic, multi multiplication that you're doing and you are just adding this condition. Question two, you're gonna use this part that it says length, the length part actually is gonna return the number of items in this container. Basically it's gonna count all the, the letters in doc and if the letters in doc, it's less than  $X$  in this case, yes, you're in a print. Question two works. Okay, got question three basically is the exponential basically is gonna be  $X$  exponential to three if it's E, if it's bigger or equal to Y.

S1 Speaker 1 ▶ 2 20 10

And here we are gonna use an operator. And so both of these needs to be met in order to print. Got question three, then the done is in a group of three. We're gonna find the variable name, it's equal Dan. Then we're gonna define what we're gonna see next. The list, this is how you define a list in Python is the same thing as the arrays in B, B, a. So in group one you put Greg, Tony, Susan, Gerald, Powell, Ryder, Carla, Dan, and Jefferson. So he is gonna look through these three list and he is gonna try to find Dan how with this condition, if name in group one, print this message if name in group two, print this message if name in group three, print this message or else does not have a group. And in the last part it's actually we can, they can write bumper cards.

S1 Speaker 1 ▶ 2 21 09

We give a height, we have, we give an H and we the adult permits permission and basically we're gonna work with the condition. If height is bigger than 70 and age bigger or equal to 18, then you can ride all the rollercoasters and then you go on in others in other section. The other inter interesting section that we have here is this one because we have double brackets because we have an order between two different nested conditions here. But here's just an example on how powerful python can be because you are doing one condition here, an operator, another condition here, and then after that you are using another operator that is gonna compare or it is not gonna compare, but it's gonna give you the option that if this met or this met, then you can write bumper. Right? Any questions about this activity guys?

S8 Speaker 8 ▶ 2 22 14

I do for the double, the double aspects on in question three. On three question is if X and those two little stars, what does that mean?

S1 Speaker 1 ▶ 2 22 26

Potent

S8 Speaker 8 ▶ 2 22 28

Potions?

S1 Speaker 1 ▶ 2 22 30

I mean is a are medical speaking the potent

S8 Speaker 8 ▶ 2 22 39

Oh the potent. So is like the potent Oh, oh I see potent. Why is that to the, to the potent in there.

S1 Speaker 1 ▶ 2 22 49

Sorry,

S8 Speaker 8 ▶ 2 22 50

Why? Why is that in the PO potent in there? What is,

S1 Speaker 1 ▶ 2 22 53

Oh it's just to represent the aari medical use of a potent here. So to the potent of three needs to be bigger or equal to Y.

S8 Speaker 8 ▶ 2 23 08

So, okay, so X is equals to two, so that'll be two times two, four times two eight. Right? And then what? And then three what I'm, I'm, I'm very lost in that. So sorry.

S1 Speaker 1 ▶ 2 23 26

So this is an operation. How much, what's the result on this one Denise?

S8 Speaker 8 ▶ 2 23 32

So this excess equals to two so we're gonna do two times two is four times two 16.

S1 Speaker 1 ▶ 2 23 42

Oh right. And that is bigger than Y.

S8 Speaker 8 ▶ 2 23 47

Yeah, but when, why is it three in there?

S1 Speaker 1 ▶ 2 23 51

Oh no, it is just part of the exercise.

S1 Speaker 17 ▶ 2 23 55

It's basically two to three, right? So it It's eight.

S1 Speaker 1 ▶ 2 23 59

Yes, it's

S8 Speaker 8 ▶ 2 24 00

Eight. Sorry, yes. Sorry, I got it confused. And then you, I multiply that eight times three.

S1 Speaker 1 ▶ 2 24 09

No, no this is just like the push the two two to the third power, right? Exactly. Two to the third power, that's it. Oh

S8 Speaker 8 ▶ 2 24 21

I see, I see. Okay, got it. Thank you.

S1 Speaker 1 ▶ 2 24 25

No problem. And this is five to the two Power. Power, yes. Right. Okay. Any other questions?

S6 Speaker 6 ▶ 2 24 43

Oh go. What does it means that if land dog, what's the mean?

S1 Speaker 1 ▶ 2 24 49

The length is gonna take the length of this container, of these letters. So it's gonna take, it is gonna count the letters on this string.

S6 Speaker 6 ▶ 2 25 02

Oh cool.

S1 Speaker 1 ▶ 2 25 03

Okay.

S8 Speaker 8 ▶ 2 25 05

So there's two in there, right? Because it starts from zero

S1 Speaker 1 ▶ 2 25 08

Exactly. 0 1, 2. Perfect. Any other questions? Okay, great. So let's go to the arrays or to the list in, in Python we see this as list and we're gonna have some functions that work with list very well. So in Python you can

create list with different data type, different data types inside how you create the list. You create a variable equal two and then you use the square brackets. And in the square brackets you use the fine, all the variables are all the type, all the data types that you want to put in here. Okay, here are they just printing the list and this is one of the elements that the list allow us to use and we are gonna use it with dot notation. Once that we have our list defined in a variable, then we can put my list dot append and I am going to append math. So basically when I'm doing this, math is going to be append to this list. Okay. To the end of the list. Is that clear?

S1 Speaker 1 ▶ 2 26 31

Yeah. Perfect. Okay, so the next part is actually I want to return the index of that first object. Basically I'm going to print my list index and math. What index I am going to print here guys. 4, 4, 0, 1, 2, 3, 4. Yes, exactly ma'am. Perfect. Because it is 0, 1, 2, 3, 4. So it's exactly the same thing as bba, but here we start with zero as well and we use square brackets. Okay. And because we are appending math to my list, so this leads is going to look like this now. Okay, print. And here I am actually getting the index of math, so it's gonna be 0, 1, 2, 3, 4. Yes. Good. Okay. The next part is gonna change and specify element within a list and the give an index. So I am going to access to my list with the index, my list and then square brackets. The index I want to add in this case is gonna be 0 1 2 3 80 and I'm gonna change it to 85.

S1 Speaker 1 ▶ 2 27 54

Okay, so this 80 is not going to appear in this print. In this print is going to appear an 85. Then I am going to return the length of my list with len my list and it's gonna give you, your list contains as many elements of it's contained. Okay? I can remove a specify objects from a list with the my list remove and I can remove Matt again. Okay. Another way of removing objects is with index specification, my list pop zero. In this case I'm gonna remove Jacob. And then my new index zero is gonna be 25. And if I do my list dot pop zero now I am going to remove 25 from here in this print list. Questions so far about list? All good thumbs up?

S8 Speaker 8 ▶ 2 29 00

I I do, sorry, I messed when the, you said the print L my list, is that gonna print how many items are in the, in the list?

S1 Speaker 1 ▶ 2 29 12

Yes, exactly. So my list starts with four items, then I add another item. So now I have in this my list five items. Then I get my index from math, it's gonna be my index four, then I change my index three. So I change 82, 85. Okay. So I am, I am just changing that, but I am keeping mat, then I get the length of my list. Okay, then I remove mat. So I go to my original list, but instead of 80, 85 then I pop. Pop is removing elements with indexes, not with names. So with pop I remove elements. Here I am going to remove zero. The first zero I have Jacob, then another index that is gonna be zero. Then for the new list is going to be 25. Okay, so if,

S8 Speaker 8 ▶ 2 30 19

If you call back, because what, so between pop and remove, what is the difference? If you get, if you pop Matt, you can still call your list and per se in a loop and we will include Matt or what's the difference?

S1 Speaker 1 ▶ 2 30 40

The difference here basically is remove, it's gonna remove an exact element on the list. Okay. You select the element like the string or the integer. Okay. Oh,

S8 Speaker 8 ▶ 2 30 52

So pop is by index?

S1 Speaker 1 ▶ 2 30 54

Exactly. Pop is by index. Got it. No problem. And the other part is that we have two, the two, the only difference between list and two is scripting is just curly brackets, the normal brackets let's say. And that this is immutable in Python. Okay. So basically pupils can't be changed or can't be manipulated like this. So in two you don't have removed, you don't have pops, you don't have a pens or anything like that because this is in multiple, okay, that's the main difference. And this

S8 Speaker 8 ▶ 2 31 36

Like an

s1 Speaker 1 ▶ 2 31 36

Absolute one of the questions that you get always in coding interviews.  
Like what's the difference between triples and lists? Okay,

s8 Speaker 8 ▶ 2 31 46

So topos is like, is is like established list, like you can't like, it's like

s1 Speaker 1 ▶ 2 31 53

Exactly like a static list.

s8 Speaker 8 ▶ 2 31 55

Okay, got it.

s1 Speaker 1 ▶ 2 31 58

No problem.

s1 Speaker 17 ▶ 2 31 59

One question you go, so line number 24 and 25, you used couple of  
same lines, right? So will it work this way? Like my list dot pop zero  
comma one, it won't work, right? I mean what if I want to remove the first  
two indexes at a single, in a single line? I mean this,

s1 Speaker 1 ▶ 2 32 20

Yes, in that case you can do it like this.

s1 Speaker 17 ▶ 2 32 24

Okay.

s1 Speaker 1 ▶ 2 32 28

Yes you can. You can work with the indexes. Yeah.

s1 Speaker 17 ▶ 2 32 33

Okay, sounds good. Thank you.

s1 Speaker 1 ▶ 2 32 35

No problem. Okay,

s2 Speaker 21 ▶ 2 32 40



Ugo? Yes. Is my tupu a variable name or a function?

s1 Speaker 1 ▷ 2 32 47

It's a variable name.

s2 Speaker 21 ▷ 2 32 49

Okay,

s1 Speaker 1 ▷ 2 32 50

Just

s2 Speaker 21 ▷ 2 32 52

So how do you know which is to to pull or list? I mean how do you, because it looks like you are using,

s1 Speaker 1 ▷ 2 33 02

This is different. The brackets.

s2 Speaker 21 ▷ 2 33 05

Oh, the brackets.

s1 Speaker 1 ▷ 2 33 06

The brackets. Look at the brackets. Square brackets.

s2 Speaker 21 ▷ 2 33 11

Oh okay, okay, I get that.

s1 Speaker 1 ▷ 2 33 13

No problem. Any other questions here guys?

s6 Speaker 6 ▷ 2 33 18

Go. I didn't understand this between the brackets what you just

s1 Speaker 1 ▷ 2 33 22

Said, if you use square brackets, it's a list and you can do all the modifications to the list. If you use these kind of brackets, it's a tool and you can't do any modifications to that to pull.

s6 Speaker 6 ▷ 2 33 36

Oh okay, got it. If we use parenthesis, we cannot make any modification.

S1 Speaker 1 ▶ 2 33 41

Exactly. Okay, so because we are running out of time, we are actually going to jump to this activity that I'm gonna help you do it. The rock paper, scissors. Let me open the RHYTHMIA here. So as you can see here, it says create a R P S game that takes a user input from the command line and plays against the computer using the terminal take and input off rock paper or scissor, which will stand for rock paper, scissors. Have the computer randomly pick one of these three choices and then compare the user's inputs to the computer choice to determine if the user won lost or tied. This problem guys, believe me, I got this problem in one coding interview and it was really, really useful to, to see it first before doing it because I needed to like coat in that moment so I remember it.

S1 Speaker 1 ▶ 2 34 46

So basically what I'm gonna do is first off you're gonna get introduction of the first incorporate the random library, okay? This is how you incorporate libraries to your Python program. We're gonna talk about this in next classes. Don't worry about it just for this one. You just have to know that this is a library that I am going to use with my Python program. Import random, okay, it's a library. Then print the title, print, let's play rock, paper and scissors. I am going to specify the three options in a list options and I select I, I create a list with r, p and s, rock, paper, scissor. Then this is what this library is doing here. Creating, I am going to create another variable that, it's called computer choice. I am going to use my library, random choice and what choice in this options, okay, choice from this array.

S1 Speaker 1 ▶ 2 35 52

Okay, this is what the computer is doing randomly. Then I am going to add my user selection. User choice is gonna be an input with a question. Make your choice and I am actually specifying the user. Select R for R. P for paper and S for csar. Okay, inside this I am going to run this conditional user choice is equal, equal R and computer choice is equal, equal P, you choose rock the computer, choose paper, so you lose. And

I'm going to do all the eh combinations that I can. Okay? Rock and cassar. Rock and rock. Paper and paper. Paper and scissor. Paper and rock, cassar and paper. Scissor and scissor and scissor and rock. Okay, else if they don't use this notation, for example, if they put a number, if they put something else, I don't understand that. Next time choose from R, P or S. Okay, let me run it so you can see how it works. I'm gonna close. Here it goes. Let's play rock paper, scissor. I'm gonna choose S, you choose ssar, the computer, choose scissor, a smashing tie. That's it. Any questions about this activity, guys? Okay, all good. Thumbs up.

S1 Speaker 12 ▶ 2 37 36

Hugo, how was random working here?

S1 Speaker 1 ▶ 2 37 40

Random, basically what is doing is selecting one of these three choices from the list and putting it in and saving this option in this variable that is called computer choice.

S1 Speaker 12 ▶ 2 37 55

Okay, so you are putting the conditional condition saying if the computer chooses P and the user's choices are, then you are printing the statements.

S1 Speaker 1 ▶ 2 38 04

Is that

S1 Speaker 12 ▶ 2 38 05

Okay? Got

S1 Speaker 1 ▶ 2 38 06

It. And I am doing it for every combination.

S1 Speaker 12 ▶ 2 38 09

Got it. So R, p, r, S, r, R and then another three con conditions with P and then three more conditions with S.

S1 Speaker 1 ▶ 2 38 17

Exactly. J

S1 Speaker 17 ▷ 2 38 19

Thank you Chico, can you scroll up please?

S1 Speaker 1 ▷ 2 38 22

Yes,

S1 Speaker 17 ▷ 2 38 23

Yes. So when you defined computer choice, you added random dot, why is there a choice?

S1 Speaker 1 ▷ 2 38 33

Oh, that's how you defined the random function here. Random choice.

S1 Speaker 17 ▷ 2 38 39

Okay, so it's, it's not random dot options or random dot something, it should be random dot choice.

S1 Speaker 1 ▷ 2 38 46

Yes. That's how you,

S1 Speaker 17 ▷ 2 38 48

Okay. Yeah, because it it, it got colored in blue, so that's where I was confused. Okay, got it. Thank you.

S1 Speaker 1 ▷ 2 38 55

No problem.

S1 Speaker 12 ▷ 2 38 57

Just to add to that is random the library and choice the function or like the module that you

S1 Speaker 1 ▷ 2 39 05

Basically random, it will be the function and the choice is going, is going to be the method that is gonna use an argument.

S1 Speaker 12 ▷ 2 39 11

Ah, got it, got it. Thank you.

S1 Speaker 1 ▷ 2 39 14

Good, no problem.

s2 Speaker 21 ▷ 2 39 17

Ugo, I don't know if you mentioned this, but why do we need the random library game?

s1 Speaker 1 ▷ 2 39 22

The random library? We needed it for the computer to do random choices on my options so you can play the game.

s2 Speaker 21 ▷ 2 39 32

Okay, so it it allows you pick any of the available options. That's,

s1 Speaker 1 ▷ 2 39 39

It allows, it allows the computer to select any of my options in my array. Correct.

s2 Speaker 21 ▷ 2 39 47

Okay. Okay.

s6 Speaker 6 ▷ 2 39 49

Hugo, why did you put it in square brackets? The option we change,

s1 Speaker 1 ▷ 2 39 56

Sorry,

s6 Speaker 6 ▷ 2 39 57

You put in a square, square bracket because we cannot make any alteration on that or

s1 Speaker 1 ▷ 2 40 03

No, because we, we can do alterations on this one. So it's, I'm defining a list.

s6 Speaker 6 ▷ 2 40 09

Oh, okay. Every time you put

s1 Speaker 1 ▷ 2 40 12

The tupu is a normal parenthesis.

s6 Speaker 6 ▶ 2 40 17

Oh yes, true.

s1 Speaker 1 ▶ 2 40 19

That's the tupu. Yeah,

s6 Speaker 6 ▶ 2 40 21

I, I wrote the, the right thing.

s1 Speaker 1 ▶ 2 40 23

I'm just asking the questions, sorry. Oh no, don't worry. Okay, any other questions guys? Perfect. That's great. Let's jump to the last topic of today's class and so we can actually work, you can work in the last activity of today's class because you will have time to work on the loops. Loops. Again guys, they are quite the same as B, b a, sorry that I mentioned this a lot, but they are quite the same. Just the scripting, the script is changing, but basically how you write a loop here in Python, it's easier for me for X in range five. When you say range five, as you can see here, you, you mean from zero through four, not including five. Okay, that's how Python works. So here, what is going to print?

s1 Speaker 1 ▶ 2 41 36

Anyone? Lucky guess zero through four. Perfect, zero through four. Thank you very much for that. So in the next part, this print is just to, I'm gonna print it and you're gonna see that it's just giving format to my print. Okay? Loop through the loop through a range of numbers. Two through six. Yes, six up two, but not including seven. So I'm gonna include two, but I'm not including seven here and I can define range two to seven. So here I am going to print from two to six. Okay? Then I'm gonna iterate through a word. Word. I'm, I'm, I'm putting piece, so for letters, this is my I, my X. Remember this is a variable. I, I don't care if you, sir, you use i x, Y, whatever you want to use. Okay? For letters in word print letters, okay here because I am using a string and I am looping through that string, I am going to print p e a c e. Okay then I am going to iterate through a list for animal ensue. Again, this is variable, okay, for animal ensue print animals. So I'm gonna print cow, dog bee and zebra

and we have another kind of loop, the loop while, okay the loop, while I am going to run equal y, just like that. So I am going to say while run it's equal to while, while this condition is met, print, high run equal to input to run again and third Y. So if I put something different than Y, my loop is gonna end.

s1 Speaker 1 ▶ 2 43 41

Let me run this so you can see it. Okay? The first one, as you can see here, is zero to four, the second one, two to six, the third one is the letters, p, e, a, c, e, the other one is gonna be the list. Cow dog, B and zebra. And the last one, it's going to be my hi run again. If I click yes, run again. Yes, run again. Why? Why if I click n, that's it. Questions about loop skies,

s6 Speaker 6 ▶ 2 44 25

I got confused.

s1 Speaker 1 ▶ 2 44 27

Yeah,

s6 Speaker 6 ▶ 2 44 29

So two through six,

s1 Speaker 1 ▶ 2 44 32

Two to six. Okay, so here when you were using range, we are including the first number, but we are not including the the last number. Okay? So I am going to run from two to one number before that number that I am specify. Okay? So for example, if I put here two to eight

s6 Speaker 6 ▶ 2 44 57

Would be two, seven,

s1 Speaker 1 ▶ 2 44 59

Sorry,

s6 Speaker 6 ▶ 2 45 00

It would be two to seven.

s1 Speaker 1 ▶ 2 45 01

Exactly.

S6 Speaker 6 ▶ 2 45 03

Okay,

S1 Speaker 1 ▶ 2 45 06

Any other questions? Okay, great guys. So in that case we can work in this activity. We have the rest of the class, like 13 minutes to work on this activity and then I can show you the, the answer. So basically this activity is chain up the numbers using a wild loop, ask the user how many numbers, then print out a chain of ascending numbers starting at zero. After the results have printed, ask the user if they would like to continue. If why restart the process and starting at zero again if and exit the change there is a bonus rather than just displaying numbers constantly starting at zero, how the numbers beginning at the end of the previous chain. Okay, so basically instead of rebutting, just start from where you left. Questions of the activity guys. Okay, so let's start working. We have about 12 minutes to work on this activity and if you have questions, rooms, breakout rooms are open, you can go in one to in one of the rooms and join there. Or you can stay here and ask me questions directly. Okay,

S8 Speaker 8 ▶ 2 48 21

Go. I have a question. Does the, when is to using like, like how many numbers using the wild loop? So I will have to specify the input, how many numbers in order to get the user's input in order to create the loop. Is that right?

S1 Speaker 1 ▶ 2 48 47

Mm. Can you repeat that again? Sorry.

S8 Speaker 8 ▶ 2 48 49

So this is the instruction says using a wild loop as the user, how many numbers the printout a chain of numbers in increasing order from zero to the user input number. So that means that I will have to first put the input from how many numbers asking the user and based on what the user says, I will put the loop.



s1 Speaker 1 ▶ 2 49 16

Exactly.

s8 Speaker 8 ▶ 2 49 17

So how did I know that before running the code?

s1 Speaker 1 ▶ 2 49 22

Actually the user is the one that is gonna indicate that for you. So if the user types five, the code needs to run from zero to four.

s8 Speaker 8 ▶ 2 49 32

Okay, so for my variable would be user's number and then parentheses, empty

s1 Speaker 1 ▶ 2 49 39

User user number is gonna be equal to in input how many numbers and then your four loop is gonna be four x, y, whatever in range. Remember the range is the one that is gonna be counting. So if the user print five, the range is gonna hold the user number. So basically the range is gonna have five.

s8 Speaker 8 ▶ 2 50 06

Got it. Okay, thank you so much.

s1 Speaker 1 ▶ 2 50 08

No problem guys, I'm actually gonna start showing my screen because I want to show the answer before we finish. Okay? And boom. So, and using my screen, thumbs up. Perfect. Okay, so what we have here, mm, basically we have initial variable to track a game play. This is my user play. Remember that We have to ask the user if he or she wants to keep playing. Well this is going to be so inside this for loop it's gonna be my while loop because remember in Python, indentation matters. So while user play it's equal, equal y, then do this. Okay? Ask the user how many numbers to loop through user number is gonna be equal to input. How many numbers then for X in range of that integer of numbers that the user give us. I'm gonna print X and then I'm gonna print the question. User play input continue, yes or no? Okay, I am going to run it. That's like

without the bonus. Okay? And as you can see here, it says the number. So if I print five, how many numbers it it's going to bring? Do you know? It's a question zero to four. Zero to four. Perfect. Yes, yes, yes, yes. Good. So print 5, 0, 1, 2, 3, 4. You want to keep playing? Yes. And it's gonna restart everything. Six zero to five. Any questions about this first part?

S1 Speaker 15 ▶ 2 59 07

Google, can you tell me about the, the indentation for while user play? Again, I can't, it's like my code is broken.

S1 Speaker 1 ▶ 2 59 15

Yes, there no problem. So basically I define this variable outside my loops, okay? Because this variable is the one that is gonna be helping me to keep playing or stop, okay? So my, while this is the while loop, it's gonna be like a loop with a conditional already embedded. So I'm gonna say while user play it's equal to this letter is a letter y then do this whole process. You see that this process is inside of my while, okay? Actually I can open it and close it, you see because in a visual studio you can, you can do that. So yeah,

S1 Speaker 15 ▶ 3 00 05

I think my issue is I can't get the i I can't get the arrow thing to pop up for right next to while.

S1 Speaker 1 ▶ 3 00 16

Okay we, we can check, we can check Terry, I mean after, I mean we're gonna start off these hours in just few minutes so we can check, I can check your screen, no problem. Okay. Too soon. And I'm going to open the bonus part. The bonus part is exactly the same, but I am going to start the number. Okay? I am going to start on last number. That is gonna be zero. So basically when I am looping in this range, I am going to start that number, okay? It's gonna be zero, the integer of the user number, okay? Because it's gonna be zero, it is gonna be five, six or whatever plus my starter number. So the first iteration is gonna be if my user number is five, my first iteration number is gonna be five. Okay? Because my start number is zero. But then in the next one is gonna be six in the

next one is gonna be seven, the next one is gonna be eight. Okay? And that's how you do it. So let's see, let's see the difference. So here I'm gonna put five, I'm gonna click yes, I'm gonna click six. And you see I am going to start from the number I left four five. Why? Because my range is going to start from the number that I leave plus the start number. Okay? Exactly.

s2 Speaker 20 ▶ 3 02 00

Hey Hugo, can you scroll down in the code? Yeah. So, okay, so there's another line of code where you're rein initializing the start, the start number.

s1 Speaker 1 ▶ 3 02 13

Cause I,

s2 Speaker 20 ▶ 3 02 14

Because I think before when you had that screen, it just looked like it was starting from zero all over again.

s1 Speaker 1 ▶ 3 02 20

Oh. Oh, okay. No, it's gonna start from a start number plus the integral user number. Yeah.

s2 Speaker 20 ▶ 3 02 29

And so the way the loop, this is the way the loop is going is it doesn't go beyond line eight, it it rein initializes to line 11 and goes from there.

s1 Speaker 1 ▶ 3 02 44

Just OK while line putting Y. Yes.

s2 Speaker 20 ▶ 3 02 49

Okay, think thanks. That makes sense.

s2 Speaker 21 ▶ 3 02 54

Ugo, can you explain the four four loop statements?

s1 Speaker 1 ▶ 3 03 00

Yes. I

s2 Speaker 21 ▷ 3 03 01

Mean how does it work with range stat number, user number plus stat number? Yes, I'm a little bit confused.

s1 Speaker 1 ▷ 3 03 11

No problem, I will do that. But first let me end the class because we have five minutes pass over the time and I then I can explain you that kicking over Ondo. So guys, this is it for today's class. If you don't have any questions and you want to leave now, that's per perfectly fine. Thank you very much for today's class and for your attention. And if you want to stay, the rooms are open and the TAs are going to be in one of the rooms. So you can go and ask questions to them. I dunno exactly how is set up. Bill maybe can help us with us with that. But if you don't have any questions, thank you very much guys, and see you on Thursday for the second class of Python. Thank you guys and have a good night. So, okay, so Kiki, basically you want me to explain the for loop, right?

s2 Speaker 21 ▷ 3 04 14

Yeah.

s1 Speaker 1 ▷ 3 04 15

So basically let me, let me tell you what is for Loop doing. I'm gonna close this. Okay, so this for Loop, what is doing is doing a range, okay? Range between two numbers here. So okay, to explain you this, I am going back into the basics, remember about this range?

s2 Speaker 21 ▷ 3 04 41

Yeah, yeah,

s1 Speaker 1 ▷ 3 04 43

Yeah. From two to seven. So I am going to go from two until six because I'm not including seven, right? Yeah,

s2 Speaker 21 ▷ 3 04 52

Yeah.

s1 Speaker 1 ▷ 3 04 53

So I'm doing exactly the same thing, but with variables instead of numbers. My variable start number is a zero,

s2 Speaker 21 ▷ 3 05 02

A zero

s1 Speaker 1 ▷ 3 05 04

And I am going to end here with my user number plus my start number. That is a zero. So if I put a five, my range here is going to be, it's going to look like this.

s2 Speaker 21 ▷ 3 05 20

Zero five. Okay. Yes,

s1 Speaker 1 ▷ 3 05 22

Exactly. Okay. Okay. And that's it. Basically when I, when my start number, when when this is finished, you see this start number is out of this loop. Okay? Yeah. But inside of my while loop, mm. So when these four loops ends and I click adjust, my start number is going to be the definition that is gonna be zero plus my integer. User number.

s2 Speaker 21 ▷ 3 05 55

User number, okay, I get it now, thank you.

s1 Speaker 1 ▷ 3 05 59

No problem.

s1 Speaker 12 ▷ 3 06 02

You have one question. Go ahead.

s2 Speaker 21 ▷ 3 06 03

Go ahead.

s1 Speaker 12 ▷ 3 06 03

Sorry. I'm sorry. Can who, should I ask or do you wanna go? I, no, go

s2 Speaker 21 ▷ 3 06 09

Ahead.

s1 Speaker 12 ▶ 3 06 10

Okay, so the first iteration, when you put five, it goes from zero to four, which is fine. The second iteration, if I say my user number is six, right? I input six there, then my start number becomes six plus zero, which was previously the start number. Is that correct?

s1 Speaker 1 ▶ 3 06 31

That's correct Jay.

s1 Speaker 12 ▶ 3 06 33

So in the first statement, the start number becomes six, the user number is six plus the start number again, which is six. Again, I don't, I, this is the part where I'm getting

s1 Speaker 1 ▶ 3 06 45

Yes, six again. So you're gonna start

s1 Speaker 12 ▶ 3 06 48

From six and end at 12.

s1 Speaker 1 ▶ 3 06 50

Exactly.

s1 Speaker 12 ▶ 3 06 52

But that's not what is getting printed, right? I'm, I see this, the print statement is printing five to 11, is that correct? No. Or five to 10. Why is this happening?

s1 Speaker 1 ▶ 3 07 04

Yes. Let me, let me explain you this part. Okay. Okay. So basically what's going on? Cause I have so many things here. Okay, lemme see. Just gonna run this one. How many numbers? Okay, I'm gonna print. You say four? No, five first.

s1 Speaker 12 ▶ 3 07 26

Yeah, five.

s1 Speaker 1 ▶ 3 07 29

Cool. Lemme just, yeah, I'm gonna print five here, so it's gonna be 0 1, 2, 3, 4.

S1 Speaker 12 ▷ 3 07 39

Got it. Yeah.

S1 Speaker 1 ▷ 3 07 40

Right. So that's the free run because my, my start number is, is zero  
Zero. Right. Okay. It's gonna be zero coma, five plus zero, so it's gonna  
be zero coma five. Right. That's, that one is understandable,

S1 Speaker 12 ▷ 3 07 58

Correct. Yeah.

S1 Speaker 1 ▷ 3 07 59

Printing that. But then my start number that is outside this loop is gonna  
be equal to start number. That is zero in this case. Plus my interior  
username. That is five.

S1 Speaker 12 ▷ 3 08 13

Right. So it becomes five there.

S1 Speaker 1 ▷ 3 08 15

Exactly. So when my user play input suggest it becomes five. Yes.

S1 Speaker 12 ▷ 3 08 22

Okay, got it. Okay.

S1 Speaker 1 ▷ 3 08 25

I print a J, it becomes a five

S1 Speaker 12 ▷ 3 08 29

And then now if you say 10, it starts to print from five to 10, which goes  
from five to nine is what is printed,

S1 Speaker 1 ▷ 3 08 36

Then you go to 10. Exactly.

S1 Speaker 12 ▷ 3 08 39

No. Okay. Five to 14, right, because you're printing 10 numbers. Yeah,

S1 Speaker 1 ▶ 3 08 45

Exactly. Well, but you are gonna start from five,

S1 Speaker 12 ▶ 3 08 48

Right? Makes sense. Okay, got it.

S1 Speaker 1 ▶ 3 08 50

You are not starting from zero again,

S1 Speaker 12 ▶ 3 08 52

Right?

S1 Speaker 1 ▶ 3 08 53

Yep.

S1 Speaker 12 ▶ 3 08 54

So now your start number after the hydration becomes five plus 10, which is 15, is that correct?

S1 Speaker 1 ▶ 3 09 04

That is correct. And you're gonna start with 15. Okay, so slightly, yes. Yeah, I put two.

S1 Speaker 12 ▶ 3 09 11

So 15 and 15. Okay, got it. Alright, thank you.

S1 Speaker 1 ▶ 3 09 14

No problem. Any other questions?

S2 Speaker 21 ▶ 3 09 21

Do we get those solution files? Excuse

S1 Speaker 1 ▶ 3 09 24

Me? Yes, yes, yes. They're gonna bring the solutions today. Tonight.

S2 Speaker 21 ▶ 3 09 29

Okay. Thank you.



S1 Speaker 1 ▶ 3 09 31

No problem. I

S6 Speaker 6 ▶ 3 09 32

Would, I would ask for this cuz I spent so much, so much time trying to figure it out my calendar. So basically I missed the, the whole class.

S1 Speaker 1 ▶ 3 09 41

Okay. I'm

S6 Speaker 6 ▶ 3 09 42

Confused with Python overall. I'll have to watch it again.

S1 Speaker 1 ▶ 3 09 45

Okay. Yes, no,

S6 Speaker 6 ▶ 3 09 47

If I have all the, all the, the activities I, so I can go over, it will be easier for me tomorrow.

S1 Speaker 1 ▶ 3 09 55

Yes, definitely Kelvin And you can watch the video and you can ask questions through Slack as well too, as if you are stuck somewhere.

Okay. Okay.

S6 Speaker 6 ▶ 3 10 03

Yeah. Cause I was trying to code him for some reason I wasn't seeing like the difference of colors and, and then I just realized that I have to change to Python on the bottom. I just could figure it out. This, it took me a while.

S1 Speaker 1 ▶ 3 10 17

Okay. No, no problem. Col.

S6 Speaker 6 ▶ 3 10 19

Thank you so much, Juul.

S1 Speaker 1 ▶ 3 10 21

No problem. Thank you guys. And I think, Terry, you have some questions, right?

S1 Speaker 15 ▶ 3 10 25

Yeah, can, can you show me, I kind of figured out why I was giving me, it wasn't doing the dropdown cause I didn't have anything underneath of it, so it kind of has nothing to link to. Can you show me the very bottom of this?

S1 Speaker 1 ▶ 3 10 39

Yes. Let me show you that. I'm just this one, right?

S1 Speaker 15 ▶ 3 10 45

Okay, awesome. One sec.

S1 Speaker 1 ▶ 3 10 53

Okay. And if you want to share your screen, go ahead and I can try to help you

S1 Speaker 15 ▶ 3 10 56

Out. Oh no, I, I think it was just the, the fact that I didn't have the,

S1 Speaker 1 ▶ 3 11 01

The last piece. Okay.

S1 Speaker 15 ▶ 3 11 02

Anything underneath it was just showing red lie dokey. Thank you so much Hugo. Appreciate it. No

S1 Speaker 1 ▶ 3 11 07

Problem. To, okay. And guys, you want me to go through Visual Studio Code while you are here? We can have some Yes, you go. Yes, yes please. Okay, let's do this. So Visual Studio Code is a text advanced text editor. I'm gonna tell it why, because you can code in so many language here and it, it's going to identify whatever language you are using. If you are using Python, if you're using sql, if you're using JavaScript, Java, c plus plus no Gs, whatever it's gonna, it's gonna identify that. Okay? So it's really helpful to understand that on Visual Studio Code. Okay. If I

want to go here and then I want to close everything, I'm just gonna put close folders and don't save and it's gonna close and it's gonna return to the origin as you can see. See it here. Okay. And I can open a new folder. I can open multiple visual studio codes so I, I can work with different windows. Okay. As you can see here, this is one and this is another one. Okay?

S1 Speaker 1 ▶ 3 12 29

Oh no, let me share my whole screen because if not, you're not wanting to see that. Okay. Can you see my screen? Yes. So I can open as many visual studios as I want. Okay, so this is the first part. As you can see here, we have here on, on the top corner, different, different ways of viewing my visual studio. I can change the total panel, I can change two secondary bar or customize the layout. Okay. As well. I can split this into two so I can work in different windows and I can have more actions here. Okay, so we have the traditional file, the selection view, go and run here we have different terminals. We can add terminals here in terminal new terminal. I can add GI Bash, I can add Ancon, I can add a shell shell script, or I can add whatever I want to add. In Mac, you don't really need to do this because you just have one terminal, but as well you can add a virtual environment here, for example. Okay.

S1 Speaker 1 ▶ 3 13 51

If you need help, specific things, like I have seen questions about how can I re, how can I just run a, a chunk of the code. You can actually go here and help and check on the documentation, show commands or whatever you want to do here. Okay? So the first thing that you're gonna do in Visual Studio Code is you're gonna try to open a folder because you always works, you always work in folders, okay? They save you this excellent thing that, well, for me, I think that you can open the, the past thing that you were working. For example, I'm gonna open this, it's gonna go directly in the activities that I was working on

S1 Speaker 12 ▶ 3 14 37

Yoga. One question is on my visual studio code, I don't see open folder, it directly says open. Is there a way for me to fix that?

S1 Speaker 1 ▶ 3 14 48

You don't see it here in file either,

S1 Speaker 12 ▶ 3 14 51

So if you, oh, I didn't go to the file option, but if I just look at the new window, right? If I open up a new visual code, it only says open. It doesn't specify a file, a folder in my case. So I just,

S1 Speaker 1 ▶ 3 15 05

Can you share your screen so I can see something there really quickly?  
Sure.

S1 Speaker 12 ▶ 3 15 13

It just says, oh, so this is automatically a folder, is that correct?

S1 Speaker 1 ▶ 3 15 17

Yes, that is correct.

S1 Speaker 12 ▶ 3 15 18

Okay.

S1 Speaker 1 ▶ 3 15 19

Yep.

S1 Speaker 12 ▶ 3 15 20

Got it. Sorry, my bad. I didn't see the image next to it.

S1 Speaker 1 ▶ 3 15 24

Oh,

S1 Speaker 12 ▶ 3 15 25

Where are you? Okay, yeah,

S1 Speaker 1 ▶ 3 15 28

There you go. Okay, so let me keep sharing my screen. So basically yes, I can open where I left. One of the great things about it is that if you close the window because you have to go, then you can just come here and then open visual studio code and it's gonna open exactly with your left.

So that's great because you don't have to save or anything like that. Like when you're working in a specific document, you just keep working on whatever you left or if the computer review reboots or anything like that, you will work there. Okay, so what's the menus that we have here on our left? The explorer is what you're seeing. You can actually make this smaller or bigger or however you feel more comfortable working. You can search for a specific things here, inside here, the code, you can remove your terminal here, click and close the panel.

s1 Speaker 1 ▶ 3 16 31

Okay, here, the source control actually goes to the repositories. We're gonna see how to work this with GitHub. So you can see branches, you can see changes, you can see a different repositories in this specific folder. And, and, and, well for example, yes, I'm gonna open this repository is the one that I'm working right now. And, and you see everything that has been done changes it. It gives you like, what changes it, it's done in git and you can see for example, that they change, they delete this from this document. They delete this line in GitHub for, I dunno why, but they did it. And you, you can find some comments here as well. There are variety of ways to quantify them all. So that's the work of this. You can go back to the explorer, you can close these windows here is to run and debug. Okay? Or you can run it from here. You can run it from here as well. Here is to add extensions. You can find any kind of extensions you want to get. Like for example, the Git history extension, Jupyter, VS Code Cloud extensions or whatever extensions you, you will want to work here is the database projects. Basically this helps you to work with databases and this is the SQL server as well. Helps you to work with SQL Server. You can manage accounts and you can go to settings.

s1 Speaker 1 ▶ 3 18 22

Mm, what else? You can view the common palette. For example, you can do something like this. You can select all this chunk of code, then right click it. And then you have an option here that it says common palette. The common palette is really useful because as you can see here, this is a common palette. And the common palette is, okay, you know what, I want to comment all this code. Okay? And so then you just put man, for

example, and then whoa, you, you couldn't find it. So then you just right click it and then again, let me right click it. Go to the comment palette and you see add line of comment. They give you the quick kick keys or you can just click it and that's it. You're gonna see that your code is commented. Again, you can do it, you can remove that by remove the comment line.

s1 Speaker 1 ▶ 3 19 42

So this pallet is really useful. So where are you getting that pallet, Hugo? I'm sorry. You right click okay. Here in this menu at the bottom. Okay, got it. Okay. And you can do as many things here. For example, you can select this and you can right click it and then you can go to the definition, go to the declaration, find reference rename symbol, change occurrence for mandate. So you can do many things. Run, run to a cursor, add to a watch. So basically as many things as possible. What else? In BS code. In Indias code, for example, you can run, you have goal, you have different views, you have selection, you have a date, and you have file. Right? When you run this, let's see the terminal here. In the terminal you have, you can add another terminal. As you can see, you can delete the terminals here. Okay? And you can see the problems. Specifically the output only used to check. Only the output. You can see it here.

s1 Speaker 1 ▶ 3 21 10

Well the output is not working. I have to check why. And we can see the Deach console that is here. But right now there is nothing to deach because everything is working fine. But when you have problems, you have to check the debunk console. And that's like the pretty basic steps. Of course you're gonna be, this is more than enough to keep the boat working, to create new files are here. You just go exactly wherever you want to create it. It, you can create folders here and you can refresh the explorer, you can collapse the folders, okay? To close everything, file close folders, and you, let's say you clean your visual studio code. Okay? So that's pretty much it. Visual studio guys. I mean we're gonna, we're gonna keep working on on that working. We're gonna see different, different options there and some tricks as well to work on that. Thanks Hugo. No problem. Thank you guys. Have a great night. Thanks Hugo.

Have a good night. Thank you, have a good one. Thank you Hugo. Bye.  
Bye.