Python and AI Power-Up Program Online Classroom-20250812_113242-Meeting Recording

August 12, 2025, 6:02AM

1h 44m 39s

Margi Varmora started transcription

Tarun Jain 0:04

Cover the comments. So basically if you want to give single line comment right for any of your code then probably we can just start that with hash tag. For example this is a single line comment and one more important thing why I wanted to cover comment is when we start working with agents.

Defining functions is very important, so let me show you some examples.

So this is 1 open source library called open EGI. So here what I'll do is I'll just open the functions column.

Uh, actions tools. OK.

So I hope everyone knows what is agents, right? At least you might have heard the word agents until now. And one of the best example is this mana CI, which is actually very cool enough to automate most of this stuff.

Ajay Patel 0:51

Yes, we.

Hardip Patel 0:52

Yes.

0:52

Yes.

Hardip Patel 1:00

Hello.

Tarun Jain 1:01

Coming to agents, right? I will not dive much detail now because we have specific

session for it. What I wanted to show is in order to build an agent, the major thing is you need to have tools that you build.

Hardip Patel 1:02
Yeah.

Tarun Jain 1:16
You have to pass these tools to the agent right now. This agent is nothing but the LLA model that I'm referring to. Let's take one example. The time is 7:30. If not, let me open my slides.
OK, so here let's just take some example of why the doc string is very important. So you have a task. The task is very simple. It says what's the forecast for tomorrow? Should I take the umbrella or not? OK, it's very simple. You have one context, the task, what's the forecast tomorrow once you get the context.
From first question you need to tell should you take the umbrella or not. So there are two questions right now. Based on the first question, you have tons of tools available to you, right? You have YouTube, you have GitHub, you have Yahoo Finance, you have Wikipedia and you have other tools as well.
Now, which tool do you think is most relevant to this?

Ajay Patel 2:23
Google.

Hardip Patel 2:24
Please.

Mitesh Rathod 2:25
Google of course.

Tarun Jain 2:25
Google, right. So you'll use Google and based on the Google results, whatever you search, you will just go ask what's the weather forecast for for tomorrow and you'll get certain results right. Now based on this results, you will ask the second question that you had. So will you take the umbrella when there is 28 degrees Celsius?

Ajay Patel 2:44
Uh, no, normally, yeah.

Tarun Jain 2:44
So this basically I don't want it to cover now, but the major focus is this tools building part right? In our example the question was weather forecast, so it was very straightforward to pick Google.

Hardip Patel 2:44
No.

Tarun Jain 2:59
Now let's take second example, right? What if I ask something related to stock price? In that case, which will you pick? Will you pick Google or will you pick Yahoo Finance?

Hardip Patel 3:08
It.
Yeah, fine.

Tarun Jain 3:11
We'll pick finance. Why? Because that is most relevant here. But Google will also give you the results. It's not that Google can't give that result. So what we try to do is whenever we build any tool, we need to give a doc string, right? So what doc string does is we have simple function.
Let's suppose this is a simple function.
Inside this you just have to define what is the use of it. So if you notice this this is also multi line comment. Here we started with single line comment. For multi line comment also you will start with triple quotes, triple quotes and then you can write anything of your choice.
Anything meaningful?
Right, so this is multi line comment, but if you define this under a function it's called doc string. As of now we are just discussing what is comments, but I'm telling you what is the importance of multi line comment right? So the first use of multi line

comment is.
If it is used under a function, it is referred as doc string. Now this is very important because if you use tools right, let's suppose I will go for Google search. So if you see here a Google search is a tool used to scraping the Google search engine extract information from Google search results.
So this will go as a context to LLM. So whenever I ask a task right what is the weather forecast for tomorrow, now LLM needs to decide what tools to pick because it has hundreds of tools so it has to get certain context right? That context is fetched from doc string which is a multi line comment inside a function.
It can be inside function, it can be inside class. So you understood the importance of multi line comment and second you can also use multi line comment as a string. Let's suppose we are building LLM agents, we are building RAC pipeline, we are just using Gemini. We need to write prompt right? So let's imagine if you write.
Prompt.
In double quotes, how weird will it look? How big can you write right? So that's the reason why if you want to format it well, you can also write it inside a triple quotes this thing. So if you use like this and if you're defining it in a variable, that's considered a string.
Which is mainly used for multiline string to write prompts right? Whenever you have to define it in a line by line way and if you want to make your code readable that is where there is different purpose of multiline. So to summarize, if you use triple codes there are three different purpose of this. One is it will act as a.
Comment one more it will act as a doc string and one more it will act as a a prompt or a multi line string. So here I have an example. So this is a string. If you see I'm starting with double quotes, double quotes. Then this is also a string. You have single quotes, you have single quotes.
Then this is a multi line string. You have triple quotes, you are adding multi line prompt or multi line statements and then you are closing. So all these three things are string.
Is this clear?

Hardip Patel 6:23
Yes.

Tarun Jain 6:23

OK, so there is one operator in Python which is not available in any other programming language. That is this double asterisk. So what double asterisk will do is let's suppose you have two integers, right? You have 10, then you have two. What this will do is it will just do.

Mitesh Rathod 6:24
Yes.

Ajay Patel 6:24
Thanks.

Tarun Jain 6:40
10 to the power of I'll use cap to the power of 10. So what do you think is the answer?

Ajay Patel 6:49
10 to the power of 10 that is 011 billion, 10 billion 10.

Tarun Jain 6:56
100.

Hardip Patel 6:58
Oh, isn't it a square?

Ajay Patel 7:01
Multiplication.

Tarun Jain 7:01
Yeah, sorry, it should be 10 to the power of two.

Ajay Patel 7:03
Sorry again, 10. Oh, OK, yeah, yeah, you says, yeah, 10 to the power of 10. That means.

Tarun Jain 7:07

I don't answer.
Yeah, so it is just this square into the power of. So here this operator again, it's very useful whenever you are working with shapes. So initially when you come with shapes, right for neural networks.

RamKrishna Bhatt 7:12
100.

Ajay Patel 7:13
Mhm.

Tarun Jain 7:27
You will use this operator a lot. OK, so this is 2 things that we missed out yesterday, so I wanted to cover. Let's probably start with quiz. So I have two questions from the yesterday's session that we took.
And it's very similar to what we had. It's just a quick recap. So can anyone tell me what will be the answer for this?

Hardip Patel 7:51
What about this VADOVKRK?

Ajay Patel 7:52
Vadod Vadod VADOD.

Tarun Jain 7:56
Minus it is.

Mitesh Rathod 7:59
It will start from the reverse, right? Yeah.

Tarun Jain 8:01
No one already told the answer. Yeah, it's CRA.

RamKrishna Bhatt 8:02
Yeah, OK.

Mitesh Rathod 8:06

Yeah, yeah.

Tarun Jain 8:07

Uh, what about this?

Hardip Patel 8:07

Oh, OK. OK.

Ajay Patel 8:09

OK, we have to Google print phrase. It will give an error.

Tarun Jain 8:15

It will give an error.

What if I use this?

Mitesh Rathod 8:20

OK.

Ajay Patel 8:22

Uh, phrase, phrase, uh, learn, learn, learn three times.

Mitesh Rathod 8:26

Three times.

Tarun Jain 8:26

We learn, learn, learn without a space.

Ajay Patel 8:28

Hmm.

This will capitalize. First one will be capitalized.

Tarun Jain 8:34

There are two.

RamKrishna Bhatt 8:34
First later, OK.

Ajay Patel 8:37
Yeah, and the second one will be the true.

Hardip Patel 8:38
Persist cap.

Ajay Patel 8:41
No, no, no, not true. It's a poll. Oh, yes, sorry.

Tarun Jain 8:41
So the first answer is.

Mitesh Rathod 8:44
No, false. It will be false. It's false.

Tarun Jain 8:45
close.

Hardip Patel 8:45
Uh.

Tarun Jain 8:46
It will be false, so you this will print allies.
And then what this will do is it will convert the allies into.
YLICE and then you are doing dot starts with. Now what is the data type starts with will return.

Mitesh Rathod 8:57
Buckets.

Ajay Patel 8:58
Hmm.

Mitesh Rathod 9:01
That's it.

Ajay Patel 9:01
Yeah.
It's a case sensitive, so yeah, it's.

Mitesh Rathod 9:05
But yeah.

Tarun Jain 9:06
It will return bullion. So now it is checking with A and L So we have letters AL but here it is small let L so it will be false.

Ajay Patel 9:08
Yeah.
Hmm.
Mm.

Mitesh Rathod 9:16
Oops.

Tarun Jain 9:18
Uh, what about this? It's a single line, uh, conditional statement.

Ajay Patel 9:23
Bin.

Mitesh Rathod 9:24
Thank you.

Tarun Jain 9:26
And.

Ajay Patel 9:30
Count the count 5-5 greater than.

Mitesh Rathod 9:31
That's fine.

Hardip Patel 9:37
Yes.
4 oh sorry, sorry.

Ajay Patel 9:41
Just a second 5 greater than 05 equal.

Mitesh Rathod 9:44
OK.

RamKrishna Bhatt 9:46
It will be taking account.

Ajay Patel 9:48
File.

Hardip Patel 9:49
5432.

RamKrishna Bhatt 9:50
Three times.

Ajay Patel 9:50
54.

Hardip Patel 9:52

One I guess 54321.

Ajay Patel 9:52

5-4 only five and four.

Mitesh Rathod 9:52

It will be 4.

4.

Ajay Patel 9:56

54 only 5/4.

Mitesh Rathod 9:56

54 only.

Hardip Patel 10:00

OK, yeah.

Tarun Jain 10:00

5421.

Ajay Patel 10:04

It it will be skipping.

Hardip Patel 10:05

Refer to yeah, yeah.

Tarun Jain 10:06

Please continue.

Mitesh Rathod 10:06

OK, only for matching equal to three, OK.

Ajay Patel 10:06
Oh, OK, OK. It's equal to equal to two. I yeah.

Tarun Jain 10:10
So what is happening is you're trying to use a simple decrement this thing. You start with a higher value, you're going to 0, but zero will not be printed because you don't have equal here. Now whenever it is 3, you're decrementing the number as per the.

Hardip Patel 10:10
Sir.

Ajay Patel 10:17
Yeah.

Mitesh Rathod 10:26
When is the one?

Tarun Jain 10:27
Logic. So this logic is the right count -1. It's happening, but for three also it's happening. But after that you are asking continue. As soon as you see continue it will go back right? So it's not even printing 3. Then it is doing count -1. What if I remove this?

Hardip Patel 10:30
Oh.
Yes.
It easy.

Mitesh Rathod 10:48
1/2.
So sorry.

Hardip Patel 10:51

Read.
05541.

Ajay Patel 10:55
No 5.
54.
No, no 5/4.

Hardip Patel 11:00
25431.

RamKrishna Bhatt 11:03
541.

Ajay Patel 11:03
Yeah, 54354543.

Tarun Jain 11:04
Three will be printed. No three you see count double equals to three. We get inside.
It's doing increment of one.

Ajay Patel 11:10
5421.
5421.

Mitesh Rathod 11:14
521.

Tarun Jain 11:15
I.

Hardip Patel 11:15
Decreement of 153543.

Tarun Jain 11:18
Existing.


Ajay Patel 11:18
The same answer will be same now.


Mitesh Rathod 11:18
OK.

Tarun Jain 11:21
Cool. Oh yeah, this all questions I had. So now what we can do is we'll get to the recap thing that we had yesterday.


Hardip Patel 11:23
Oh.
OK.

Tarun Jain 11:35
Till here we have completed it till while. So what we can do is we can start with the list, tuples, dictionary and set and if we still have time what we will do is we will start with functions right? So this will be the main agenda for today.


Ajay Patel 11:38
Yeah.


Hardip Patel 11:39
Yeah, we.


Ajay Patel 11:47
Mm.

Tarun Jain 11:51
So what we can do is we can write code alongside. Now whatever you are discussing now, it is considered as collections, right? So I mentioned few key points like list is a

dynamic collection, then tuple is a fixed collection. Now why this is fixed? Why this is dynamic? Once I cover tuple, I'll show you the difference.
And then we also have dictionary which is mainly used for key value mapping. So if you have already covered JavaScript, this is nothing but an object, right? And then you also have set which is similar to dictionary but it is unique. You don't have key value pair, right? So let's get started with list here.
I'll just take a simple example cart and you can add any product of your choice. I'll just add Apple iPhone, then I'll just add a MacBook Pro.
And three then.
I've got this should be. So now in order to check the data type, what function do we use?

Hardip Patel 12:55
She.

Mitesh Rathod 12:58
Bye.

Tarun Jain 12:59
It is type.

Ajay Patel 13:01
Hi.

Tarun Jain 13:02
Type of at.

Ajay Patel 13:04
Start.

Tarun Jain 13:05
It's a list, right? So in list there are multiple functions that you can utilize, right? If you want to know what are those function, one we can directly use cart and then we can just add dot and we can wait for some time. If not, there is a new command called DIR. You can just add carts inside this.

Hardip Patel 13:07

Edit.

Tarun Jain 13:24

So once you do this right, it will show you all the commands that it has, right? So most of the time again, we might not have autocomplete in some cases, but if you just want to know what are the different functions that a variable processes, you can just do DIR which is directory.

Ajay Patel 13:43

Yeah, this is similar to what we use in PHP as a reflection class, so it gives us more metadata about those variable and or the object.

Tarun Jain 13:44

OK.

Similar to which command?

Ajay Patel 13:57

Uh, reflection class.

Tarun Jain 14:03

Cool. So now what we will do is we will try to check how this append works, what is this clear, then why do we need copy, then count, extend and some of the major functions that we might use in our.

Upcoming sessions as well, so we will start with indexing. So what is the logic for indexing?

You use a variable and then you just define an integer, right? So for indexing the value.

Hardip Patel 14:28

M.

Index.

Ajay Patel 14:31

Hmm.

Hardip Patel 14:32

Mhm.

Tarun Jain 14:37

Inside the square bracket.

Needs to be integer.

All right, so this will print Apple iPhone. Now what is the syntax for slicing?

Uh, just a second.

Yeah, uh.

Can you repeat?

Hardip Patel 15:22

0 column I guess if we want to add then three.

Tarun Jain 15:25

No problem.

Hardip Patel 15:28

Uh, can you?

Tarun Jain 15:28

It can be any value, right? So it starts with 0. So what is the output for this?

Ajay Patel 15:31

Mhm.

Hardip Patel 15:35

AB.

RamKrishna Bhatt 15:37

It.

Tarun Jain 15:38
So if I print at 02.

Ajay Patel 15:38
No.

RamKrishna Bhatt 15:39
No, I see not a beating.

Tarun Jain 15:42
It won't look at the string now, so typically what is the screen?

Hardip Patel 15:44
All right. Yeah. All right. All right.

Ajay Patel 15:44
Yeah.

RamKrishna Bhatt 15:45
It.

Ajay Patel 15:48
It's on our list now.

Tarun Jain 15:48
One rule of slicing is.
Sorry, I should have added text, not code.
OK, so one rule of slicing to keep in mind.
If you are using string, OK, let's suppose you are slicing string. The output data type output will also be string.
And if you are using list slicing, the output will be.

Ajay Patel 16:36
List of item.

Tarun Jain 16:39

Will be list.

And if you are using tuple.

The output will be tuple.

Now all the syntax is same. The syntax of slicing is same if you give single index. Let's suppose if you give single value, it's a indexing. If you give colon, you start with zero and you go till two. That means you're starting with 0 index.

You'll take first index, second index is ignored, right? So now if you look at the data type, it is a list, but the syntax is same as we saw in string. But what it will do in string? Let's suppose I have word develop.

This word. If I do zero to two, this is where you'll get DE right now. Once you get DE, what is the output data type of this? It's a string. So if you do string slicing, you get output as string. If you do list slicing, you'll get output as list.

Is this clear? Now one more thing is there. If you do colon and if you do this thing right, you ignore what is the end index. It will take all the values. Same goes to you remove zero, you add -1.

Ajay Patel 17:42

Yeah.

Mm.

Tarun Jain 17:57

It will take entire thing whatever is there, but it will exclude the -1 index. You got it. This is similar to what we saw in string.

Ajay Patel 18:04

Yeah.

Tarun Jain 18:09

OK, so we'll start with the first what you call action step, which is append.

So what append will do is it will add new item or usually we refer this to be element. Append new element inside the list. So what we can do is I'll write one example which is hard. You guys can take one more example and then you can just write at least three or four elements, right? Even if you just use a simple list, it is fine like a

equals to.
List of range zero to seven or zero to six. So how many values do I have now inside here?
So what will this year have?

Hardip Patel 19:04
Zero to 5.

Ronak Makwana 19:07
125.

Tarun Jain 19:08
We will fight.
Zero to five, right? So this is a simple list. So now what we can do is we can try to modify this list with the functions that it supports, right? So the first function is we append a new element, so I'll just do a cart.

Mitesh Rathod 19:10
Sit.

Tarun Jain 19:26
dot append and I will just add.
I'll just start charger.
Right now one more thing you want to have to notice is if you define any variable, let's suppose add new is a variable and now if you add card dot append.
Cell phone it will add a new item. So let me show you the results. I do card dot append and then if I print cart I have charger. OK now if I do card dot append cell phone, cell phone will be added so.
So when you tell append it will add new element at -1 index which is at last. So now if I print this and if I print cart so where it will add you will have Apple iPhone, then MacBook, then iPad, then charger and then you'll have cell phone.
But if you print add new, it will be empty because card dot append returns none. OK, you can't assign this to any variable, you just have to write like this. That's it.
Is this clear?

RamKrishna Bhatt 20:39
Hmm.

Tarun Jain 20:41
You can't aside.
Any variable.
While using append.
OK, so now most of you know what is stack and queues, right? If I use pop, let's suppose I do cart dot pop. Which element do you think will be removed?

Ajay Patel 21:10
Cellphone.

Mitesh Rathod 21:11
From the left last element.

Tarun Jain 21:11
Cell phone will be removed, whichever is at the -1 index, right? And if you remove this, do you think you can save that in a variable?

Mitesh Rathod 21:22
It should.

Ajay Patel 21:23
It's.

Tarun Jain 21:23
So you can just tell removed element.
And then do card dot pop. Now what card dot pop will do is it will return cell phone and then it will save inside a removed element.
Is this clear? We just defined a list which is a simple array. You add multiple elements of your choice, whether it is string, integer, doesn't matter. You can also have mixed list for example.
You can define I will, then you can define 3.5, then you can define 10, then you can

also define true. So now if you see you have mixed list with string, then you have float, you have integer and then you have boolean. Even this will work right? So if you check the type of it.
It's a list.
Is this clear? This is one difference between arrays and list. In arrays when you define square bracket you should have all same elements right? Or same data type. But in Python list you need to. You can have any data types whether it is string, whether it is float, boolean, integer doesn't matter, right? So this is.

Ajay Patel 22:21
Mm.

Tarun Jain 22:40
One difference and then you can just use mixed list.
And just add dot button, you'll see all the methods. As of now we have seen append, we have seen pop which is very straightforward. And now what we'll do is let's look at copy. Why do we need to use copy? So most of the time when you do data preprocessing right some of the elements you might.
Improvise. You might delete, but you have to save the original copy as well. What is the original copy of that particular list? So here what I'll do is I will just define.
Mixed copy.
And I'll just write mixed list dot copy. Now what will happen is let's suppose mixed list the value of first index. So what is the output of this?

Mitesh Rathod 23:32
2.5.

Tarun Jain 23:32
3.5 So is list mutable or immutable?

RamKrishna Bhatt 23:33
2.5.

Mitesh Rathod 23:38
OK, well.

Tarun Jain 23:39
Mutable. Now how do we find that out? What was the command or function we used?

Hardip Patel 23:39
Yeah.

Mitesh Rathod 23:46
Mhm.

Hardip Patel 23:46
Hidy.

Ayush Makwana 23:47
ID.

Tarun Jain 23:48
We use ID right? I use ID and uh, I'll give mixed list.
It's 848. Now what I'll do is I will just write mixed.
List. OK, I'll just bring this down.
I want to make this mixed list first index to be 4.5. OK if I print this it is 3.5. Now if I do this it's 4.5. So I change the value. Now if I print the ID again.

Mitesh Rathod 24:24
It was the same.

Tarun Jain 24:25
It will be same.
You understood the syntax of what ID does. Now what has happened is I changed my mixed list. So just imagine you're doing some preprocessing. Initially your data had some value. Now you built your logic. You have new kind of data. Now it shows 4.5.

Ajay Patel 24:30

Yep.

Mitesh Rathod 24:34

OK.

Tarun Jain 24:47

Now what will happen is there might be some scenarios where original data might be required just so you can compare. It can be used for evaluation. So during that time it's always a good practice whenever you do text pre processing.

Text preprocessing. Make sure to create the copy of your original list or original dictionary. So now if I click on OK, if I click on mixed copy, what do you think will be the response? It will be the same. Whatever you have 4.5 it will be the same.

But if I use mixed copy, sorry it's 3.5 mixed copy dot.

Append.

And I'll just add some value which is 25.

And if I print mixed copy you have a new value and I will print mixed list.

It is the original 1, so understood. It's always a good idea to have a copy of your original list.

So I'll just summarize what we did so far. We started with lists. So basically in list what you need to do is you need to add multiple elements. So this elements can be of same data type or it can be of different data type. So here we had everything of string.

For A we had everything in a integer. Now once you define a string, there are different operations that you can perform. The two basic operation are similar to what we did in strings which is indexing which will extract the values from particular index.

And then you have slicing. And the rule of thumb for slicing is if your input is of one data type, the output will be of the similar data type, right? And once you completed slicing, you had a new function called DIR. So DIR you any variable, it will tell you what function it has.

For example, for I added word. So what is word? Word is a string. So for string we saw yesterday we had capitalized, we had upper, lower and everything. So these are the functions that it supports. So this is very important.

And once we cover DIR, we appended. Whenever we append any new element to the list, we just have to make sure it is not assigned in a variable. Don't assign.
It in a variable and when I say append, in which index will it append new element?

Mitesh Rathod 27:26
-1.

Tarun Jain 27:26
What is the index? It is -1 which is at the end and same when we use pop it will pick from the final element and if you're defining A variable that's fine. It will save that particular value in that variable. If you don't want to assign that even that is fine if I just remove this.
Now if I do card pop, what will be the output?
Cell phone is already removed. Now if I do card dot pop it will remove charger.

Ajay Patel 27:52
OK. Oh, sorry.
Charger.

Tarun Jain 27:57
OK and once we completed that we saw copy. So copy what we can do is we have the original string or original list. We are just creating the copy of it. Now whenever you do text reprocessing we can use mixed copy instead of mixed list.
And again, as I said, list is mutable.
You can add or delete the.
Given elements.
With this same ID that it has. Is this clear? OK, so now what we will do is we will use a different operation or function. I'll do cart dot remove.

Ajay Patel 28:34
Yep.

Mitesh Rathod 28:35
Yeah.

Hardip Patel 28:36
Yes.

Tarun Jain 28:46
So what pop does is it removes from last value, right? But what if you want to remove from any index of your choice, right? So here what I'll do is I want to remove. MacBook Pro M3, right?
And you can just define that MacBook Pro M3. So now if you print cart.
You'll just have iPhone and iPad. So this is mainly used if you have any serial numbers, right? And if you're automating some things, just give me one more second.
Yeah, I haven't slept properly so I'll catch cold that's why. So here if you see card dot remove you just give some value and it will remove that particular value from the given card. So now where does this work? How is it actually used? Let's suppose you have data set.
Right, this data set has uh serial number.
Then product name of the customer and then you have what is the price of that particular serial number. Now once you have all these list items and if you want to remove anything, most of the time what you can do is you can just use.
Whatever that variable is card dot remove and it's always good to add serial numbers. So based on that serial number you can attach that value and it will be removed and the syntax is that it has to be same if I just write iPad.
What do you think will happen here?
It will print an error because there is no value. If you see iPad is ** not in the list right? So if I use the exact same keyword then it will be removed. So now how do we get that exact?

Mitesh Rathod 30:46
Um, I didn't, yes.
It will. It will be a case sensitive, right?

Tarun Jain 30:59
That logic needs to be written within our function code. Is this clear? OK, now that we

have removed a lot of items, our cart is now just single value, right? So how do I add multiple values? What if I want to add new values in it? What was the command?

Mitesh Rathod 31:06

Do.

Hardip Patel 31:20

Bench.

Tarun Jain 31:21

Append right? Append. We'll just add one element at a time.

What if I need to add multiple elements? So this is where we have a function called cut dot extend.

So what I'll do is I'll use cart dot extend. I will add monitor.

I just heard back.

I will add baggage.

I'll add TV. So now if you see you have four more elements that you have added, you just run it. Now if you print cut you have all the values. So what extend will do is if you are using extend for tuple, whatever syntax you write inside this should be a tuple.

But here what we are trying to do is we are trying to append a list. So what is cart? Cart is a list. So when you append, append will add, sorry extend.

We'll add.

Multiple elements followed that you give inside a list. OK, this is very important. You need to give inside a list. So is this clear? Append will just add one element at a time. Extend can add multiple elements.

At a time, right? So what we will do is let's take some quick questions so that everyone are on same page. I have few examples, I'll just copy paste.

RamKrishna Bhatt 33:00

Sorry to interrupt, can we say that extend will merge to list?

Tarun Jain 33:04

Uh, can you repeat?

RamKrishna Bhatt 33:07
Can we say that extend will merge 2 list?

Tarun Jain 33:11
Extend will merge to list. Yeah, you can say that because initially this is one list and then this is a new list and you're just merging it.

RamKrishna Bhatt 33:14
OK.

Tarun Jain 33:34
Let me quickly get one example.
So you have this thing continents. So if I print the length of continents, what do you think is the response?

Ajay Patel 33:49
But of count of elements.

Hardip Patel 33:53
7.

Tarun Jain 33:54
Seven right now if I do pop.

Ajay Patel 33:56
Mm.

Tarun Jain 34:01
But I'll do sort.
And now what I'll do is I'll do continents.
dot pop. So what do you think is the response?
Uh, which one?

Ajay Patel 34:20
Alta.

Mitesh Rathod 34:21
South America.

Tarun Jain 34:22
South America. Why? Because it is S So when I do sort, what it will do is it will automatically sort everything. So now if you run continents or pop, it is South America. So if you print continents it will be in.

Hardip Patel 34:24
Oh, yeah.

Tarun Jain 34:39
Sorted order. Probably what I'll do is I'll bring Europe here.
I'll take Isha.
Add it here. I'll rerun it.
So now if you see it is in sorted order initially when you print it.
You have OK, let me have this. So you have Europe in begin, you have Asia, then once you use sort.
It will be sorted in alphabetic order. Then you also have one more syntax called continents dot reverse.
So what do you think will be the response for this?

Mitesh Rathod 35:26
Opposite like South America will be.

Tarun Jain 35:28
Let me opposite of whatever you have as of now.
It is just in the reverse order.
Is this clear?

Ajay Patel 35:39
Yes.

Mitesh Rathod 35:39
Makes sense.

Tarun Jain 35:40
OK, so one thing what you have to understand here is sort. If you just hover on this you will see it will return none. So let me show this.

Hardip Patel 35:41
So.

Tarun Jain 35:50
So if you see, can you see this arrow mark?

Ajay Patel 35:53
M.

Tarun Jain 35:55
That means whatever function you have, it will return none. It is similar to append.
For example, if you add K equals to continents dot sort, it will sort the continents OK, but The thing is K is empty.
Similar to reverse as well. If you look at reverse, I'll just open this bracket. Can you see this? It will return none.
Now if I do continents print again, let's suppose I give it inside P equals to continents dot reverse. Now if I print P.
Is this clear?

Ajay Patel 36:33
Hmm.

Tarun Jain 36:35

OK, so let's just write down whatever command functions we use as of now. So we started with append, which is one element at a time at the final index.

Mitesh Rathod 36:47

Mhm.

Tarun Jain 36:51

And then we had.

Mitesh Rathod 36:53

Extreme so far.

Tarun Jain 36:55

Then extend which is multiple elements.

Then we had copy sort.

Mitesh Rathod 37:03

We not reverse.

Tarun Jain 37:06

Reverse. What else?

Poppy covered up and covered covered. Which one?

Mitesh Rathod 37:12

Remove, remove, remove, remove.

Remove.

Tarun Jain 37:18

Remove. So what is the syntax for remove? You have to give a value. The value needs to be same as per your list.

Mitesh Rathod 37:22

Let's value and with the name.

Correct.

Tarun Jain 37:28

OK, so we have one more command. This command it's not just for list, it will be applicable for every single variable you have in Python. You just have to use delete and if you do continents.

The entire continent is gone.

OK, so this doesn't matter whether this is a list or anything. You can use delete for any variable. For example, I had word right? So word is nothing but develop. I can also delete the word.

Now if I write word, word is not defined, so delete is one more keyword we have pop. Pop is from last index, remove is the exact value and delete is for the entire list. Is this clear?

Mitesh Rathod 38:18

Um.

Ajay Patel 38:22

Yeah.

Mitesh Rathod 38:23

After deleting after delete it will also remove the like a what says resource ID right?

Tarun Jain 38:23

What's your difference?

Yeah, it will also remove the resource ID. It will free up the memory space.

Mitesh Rathod 38:32

OK.

Clear the memory space, yeah.

Tarun Jain 38:37

So one more thing about delete is it can be used with GC.

GC Python. So yeah, so most of the time what happens is when you are working with Pytorch, most of the ML models they utilize too much of memory and once you load the model you might not need any variables. During that time you use delete

keyword.
Along with delete which will remove the memory and get you some space. You can also use GC to collect the garbage this time. So whenever you use delete, make sure you use GC along with that. It's very simple. We just have to import GC and then GC dot collect.
That's it. So this will free up some space.
Is this clear? In most of the Python frameworks this is by default. You don't have to install this up.


Ajay Patel 39:27
Yep.

TJ **Tarun Jain** 39:34
And we had pop which removes from final index. We had remove, we had delete and we have something called as clear. So what is difference between clear and delete?
So let me bring back continents.
So I have continents dot clear. What it will do is it will clear the entire list and it will make it empty. What delete will do is if I use delete.
Continents.
It will remove the entire continents. You can't even print that particular variable. Now continents clear will just remove the elements, whereas delete will remove the entire variable itself.
We got the difference.


Ajay Patel 40:35
Mm.

TJ **Tarun Jain** 40:37
So I'll write it here.
We have delete which will delete.
The entire variable itself and then we have clear which clears the elements and return empty or list.
So what we can do is we can try out one or two examples and see some of the changes that you observe and once you're OK with it, we'll proceed with the tuples and you can also just note this down.

Ajay Patel 41:13
OK.

Tarun Jain 41:54
Do we have any questions in list?

Ajay Patel 41:57
No.

Hardip Patel 42:02
Yeah.

Tarun Jain 42:07
There is one more hack that many people use whenever they're using list. So whatever we saw earlier, right for conditional statements, you used if condition within the same line. Now what you can do with list is let's suppose you're creating a new.
Element and you're adding one element at a time in the list. So what you can do is you can use list comprehension, but I'll give you one better example for this. So let's suppose I have certain file parts I have.
A folder. In that folder I have data.
Invoice dot PDF.
I have data inside that I have sales dot CSV.
And I have data.
I have a API dot TXT file.
And inside the data I also have some document. So this is nothing but PRD docs. OK, so I'll just print file path. So just imagine you have a folder called data. Inside the data you have all these files which is invoice dot PDF sales dot CS.
SV API dot TXT file and PRD dot docs. So now what you're supposed to do is you need to extract only the extension of it, right? So how do you use for loop with?
So the adding is.
How to use loops with list or tuple right?
So here I'll just define for element because everything is an element in file path.
And I will just print yearly. So what this will do is it will first print invoice, PDF, sales,

API, text and PRD. Is this clear? So can anyone tell me how do you usually use follow from yesterday's example for?

Ajay Patel 44:15
Hmm.

Tarun Jain 44:20
I then what was the syntax?

Mitesh Rathod 44:23
Uh, in and then edit.

RamKrishna Bhatt 44:24
In range.

Tarun Jain 44:25
M.
Range a value and then you can do print I right? So what if you want to use range in this particular?

Mitesh Rathod 44:28
Range.

RamKrishna Bhatt 44:29
Thank you.

Mitesh Rathod 44:32
Good.

Tarun Jain 44:39
We got logic as well. So instead of file path what you can do is you can write range of length of file path. Now what this will do it will print 012.

Mitesh Rathod 44:49
I want OK.

Tarun Jain 44:55
012 because it should go till three. It will go 0123. Now what you can do is.


RamKrishna Bhatt 44:59
Hmm.

Tarun Jain 45:05
But why did it print 301? OK, it's 4444 length is a.


Mitesh Rathod 45:08
Uh.


RamKrishna Bhatt 45:09
Because we have 4/4.


Ajay Patel 45:11
Play this phone.


Mitesh Rathod 45:11
Yeah.

Tarun Jain 45:14
Length is 4. So now what we will do is we need the actual values. So what will you change here? Can anyone tell me?


Mitesh Rathod 45:24
Uh, next. Uh, we are in that index.


Ishan Chavda 45:25
The length high part of element.


Ajay Patel 45:27
No, no, no.

Tarun Jain 45:30
Oh, can anyone repeat?


Ishan Chavda 45:32
Of five parts of element in.

Tarun Jain 45:35
Correct. You just have to give file part of ELD.


Ajay Patel 45:38
And but.
Early.

Tarun Jain 45:41
So everyone understood it. When you use range it is giving you the index values. So the length is 4. It will go zero to three. Now what you're trying to do is you're using file path and then you're giving that element. Now it will print the actual values.


Ajay Patel 45:43
Yeah.

Tarun Jain 45:57
So this is how you use follow right now. There is one more way you can use it.
Let's suppose coming back to our same problem statement. The problem statement was you need to extract this given particular value. So how do we do it? Before that I'll show you one command of string.
You have something called as.
File part.
dot split.
Sorry, sorry, sorry, so.
File part.
Of zeroth index. So what will be this output?

Mitesh Rathod 46:46
The first data like data slash not PDF.

RamKrishna Bhatt 46:46
Post.

Ishan Chavda 46:47
Alright.

Tarun Jain 46:48
So I'll again repeat what we're trying to do. Our goal is to look through entire document and just get the final extension of this, right? So when I look, what is the data type of this line?

RamKrishna Bhatt 46:56
Mm.

Ajay Patel 47:00
OK.
String.

Mitesh Rathod 47:04
String.

Tarun Jain 47:04
It's a string right? So you're getting string data invoice dot PDF. Now if I do file path of 0, I'm doing split. This function we didn't look yesterday, right? Split. So you use split.

Ajay Patel 47:05
Yeah.
Hmm.
M.

Mitesh Rathod 47:12
Or sitting.

Ajay Patel 47:12
Of 0 dot.

Mitesh Rathod 47:14
And speed.

Ajay Patel 47:17
Hmm.

Mitesh Rathod 47:18
Yeah.

Tarun Jain 47:20
And now if you give back slash it will give you the file path.

Ajay Patel 47:25
Hmm.

Tarun Jain 47:26
You have you have file path. Now if you give dot then you have data invoice and then you have PDF. Is this clear?

Mitesh Rathod 47:27
Yeah.

Ajay Patel 47:35
Hmm.
Yes.

Mitesh Rathod 47:36
Got it.

Ishan Chavda 47:37
Yes.

Tarun Jain 47:37
So now what we will do is we will run a loop for element. OK, it's showing autocomplete but I removed so for element in file path.

Ajay Patel 47:43
In.

Tarun Jain 47:51
Rint element of split.

Ajay Patel 47:56
And O.

Tarun Jain 47:57
But still, if you see this will give you a list, how will I get the extension? What should I change here?

Ajay Patel 48:00
Yeah, pop pop. Just add a pop after dot split.

Tarun Jain 48:04
OK.
No, Bob will remove PDF. But yeah, what you can do is you can save that in a variable.

Ajay Patel 48:14
Yeah.

Tarun Jain 48:14
And then you can do uh.
You have.

Mitesh Rathod 48:25
Extension like EXT.

Tarun Jain 48:26
While logic and you have EXT equals to element of var of pop.
And then you can just rint extension.

Mitesh Rathod 48:39
Yeah.

Tarun Jain 48:40
But what can be the most simplest logic? This is 1 logic which is valid. I'll just print this. You have PDF, you have CSV, you have text, you have docs. This works right? So let's look at another alternative.

Ajay Patel 48:44
Hmm.

RamKrishna Bhatt 48:44
We can use ends with.

Tarun Jain 48:55
I'll remove this. I will just print -1.
And then I have the extension here, same output. So y -, 1. If you notice it is PDF. If I do one, it is CSV. It's a final index. It's same as what we did in pop. Pop will also get us the last value itself and it will save it in.

Ajay Patel 49:10
Hmm.

Mitesh Rathod 49:13
Yes, we.
Thank you.

Tarun Jain 49:20
Extension file. So here we can define pop as well. Both are same, but pop again it's running a function -1. It's direct indexing. That's it. Is this clear?

Ajay Patel 49:23
Hmm.
Hmm.
OK.

Mitesh Rathod 49:33
Um.

Ajay Patel 49:35
Yep.

Tarun Jain 49:36
Now the logic what I wanted to show is it's called.
List comprehension.
So how this works is whatever output you have right? Currently it is a individual string. What if you need to save all the output inside a list? So you understood it.
One way what you can do is you can have extension.

Mitesh Rathod 50:01
It's to declare a list.

Tarun Jain 50:03
Empty bracket. OK, now what you will do is here you will just use EXT. What is the function?

Mitesh Rathod 50:12
XD dot open open.

Tarun Jain 50:15
Opened.

videos dot append.
And we will just run this follow. Now if you print TXT you will have all the extension right? So what you can do with list comprehension is all this logic, whatever these three lines of code is there right? It will be added in one single line.
So I'll just write EXT equals to 1st is the same syntax. Then you can start with for again it's giving auto complete which I don't want.
So you start with for loop, same what we did with conditional stated statement as well. What was the syntax? You start with condition.

Mitesh Rathod 50:58
OK.
If the word stream.

Tarun Jain 51:04
If this is true, where do you add the elements?

Mitesh Rathod 51:08
Uh, is that true? It is a left hand side.

Ajay Patel 51:11
Website.

Tarun Jain 51:12
Yeah. So I'll just say in your person.

Mitesh Rathod 51:13
And falls on the right.

Tarun Jain 51:17
Is this clear? This was the syntax here. Also same you start for element in file path. Now how do you want this file path to be used that you have to write here? So I'll just tell.

Ajay Patel 51:19
Yep.

Tarun Jain 51:32
File path is there. Split that based on dot and give me the -1 index. Is this clear? You write the for loop 1st and how do you want this to be used? So it should be ELD.

Ajay Patel 51:43
Hmm.

Mitesh Rathod 51:49
Yeah, yes, yeah.

Tarun Jain 51:51
Right so you use ELE dot split dot -1. How do you want this every single individual value to be used? Now if I print extension it will be PDF CSV text docs.
This is very important. Not many people uses it, but you can still use it. So the syntax is here. You start with start with brackets, then write loop logic or I in range.

Ajay Patel 52:11
Mm-hmm.

Tarun Jain 52:21
I'll just tell it 10. Now how this I needs to be used, I'll just tell I double asterisk 2 and I will save it inside a variable called K.
So can anyone tell me the output of this?

Mitesh Rathod 52:39
It will be 222222.

Ajay Patel 52:41
Range. No, no, no. It's a just a second now for I in range one.

RamKrishna Bhatt 52:42
Table of two.
Table of two and three.

Ajay Patel 52:50
Hmm.
Yeah.

Ishan Chavda 52:53
2468.

Ajay Patel 52:57
No, no, it's a 114916.

Ishan Chavda 53:05
Yeah, it is the power.

Ajay Patel 53:07
Power starting from zero.

Tarun Jain 53:10
You start with 0, then you have one, you have four, you have 91625. Same you got it. So this is just single line for looping, but it is called as list comprehension.

Ajay Patel 53:11
M.
Mm.

Margi Varmora 53:17
Yes.

Ishan Chavda 53:18
Yeah.

Ajay Patel 53:25
OK, so for this, you know, shorter term of for for syntax, we have to write it in a square bracket, right?

Tarun Jain 53:35
Huh. Square bracket. So what we are trying to do is whenever.


Ajay Patel 53:37
Uh, OK.

Tarun Jain 53:42
Hello.


Ajay Patel 53:44
Oh.


Mitesh Rathod 53:44
Uh, yes.

Tarun Jain 53:46
I'm audible.


Mitesh Rathod 53:46
Yes, audible.


Ajay Patel 53:47
Yeah.


RamKrishna Bhatt 53:47
Yes, yes.

Tarun Jain 53:52
M.
Hello.


Ajay Patel 53:56
You are audible to us, Tarun.

Mitesh Rathod 53:57
Hello.

RamKrishna Bhatt 53:57
We are able to view you.

Hardip Patel 53:58
Hello.

Tarun Jain 53:58
OK, I guess my monitor was disconnected. Cool. Yeah, Ajay, you were saying something. OK, so here basically what we are trying to do is whenever you're starting with empty list, let's suppose you're starting with empty list and now if you want to append anything, right, you write a loop for it whenever you have any elements.
So list comprehension what it is trying to do is it will save all your details inside a list.
So adding bracket is very important.
Hello.

Hardip Patel 54:32
Yeah, yeah, you're good.

Tarun Jain 54:37
OK. Uh, does that answer your question, Ajay?

Ajay Patel 54:40
Yes, yes, it answers my questions.

Tarun Jain 54:42
Good.
Uh, till here anyone has any doubts?

Ajay Patel 54:47
No.

Tarun Jain 54:48
We can take one more example. Let's suppose you have cart and everything is in small letters like you have apple.

Hardip Patel 54:56
I forgot to ask, uh, are we recording us?

RamKrishna Bhatt 55:01
Yes.

Mitesh Rathod 55:02
Yes, we are.

Hardip Patel 55:04
Sorry.

Tarun Jain 55:06
OK, so you have apple, you have orange and then I'll just add banana. So now what you had supposed to do is you need to convert this entire cart inside capital letters. Everything should be capital and you need to create a new list. So can anyone tell me the logic for that?

Mitesh Rathod 55:28
Uh, caps.
First we needed like a four, let's say a in or ELE in.

Tarun Jain 55:35
Phone.
Yeah.

Mitesh Rathod 55:40
In card.

RamKrishna Bhatt 55:41
Start.

Tarun Jain 55:42
Alright.

RamKrishna Bhatt 55:43
Left hand side.

Mitesh Rathod 55:43
Then go to left side.

Ajay Patel 55:43
OK, left side a dot capitalize.

RamKrishna Bhatt 55:48
You know.

Mitesh Rathod 55:48
And uh dot capitalize.

RamKrishna Bhatt 55:50
Of what? OK.

Tarun Jain 55:51
Up.

Mitesh Rathod 55:52
Oh, I put it, yeah.

Ajay Patel 55:52
Upper, sorry.

Tarun Jain 55:54

So now if I use caps, everything will be in capital. So I hope everyone understood what this thing is. So one more last example I'll take. What if you need to use if condition? OK, same syntax.

Mitesh Rathod 55:57
Hmm.

Tarun Jain 56:09
You only need to save those files which is PDF. OK, so I'll take file path.

Mitesh Rathod 56:16
M.

Tarun Jain 56:20
OK, so I'll extend and I will add more PDF files data dot BRD dot PDF then data dot.
What PDF folders we usually use?
I'll just write dot PDF.

Mitesh Rathod 56:41
Assignment, assignment, assignment.

Tarun Jain 56:45
Then data.
What assignment dot PDF? Cool. So now if I print file path, I have multiple files. Now the problem statement is you only have to save the entire path, entire path, not just the extension which has PDF. OK.
Which has PDF. So can anyone tell me the logic?

Mitesh Rathod 57:12
Uh, early PDF, first for loop like a for.

RamKrishna Bhatt 57:15
4.

Tarun Jain 57:18
What?

RamKrishna Bhatt 57:20
For be in.

Tarun Jain 57:21
Yearly in file path.

Mitesh Rathod 57:21
Yeah, we mhm.

RamKrishna Bhatt 57:23
I'll but.

Mitesh Rathod 57:25
And then go to the left side.

RamKrishna Bhatt 57:28
Left side. Can we use that? Uh, from from the window? Yeah.

Ajay Patel 57:33
No, there is OK.

Mitesh Rathod 57:33
No ELE. I guess first we like takes the check that if the string has a PDF extension, right?

Tarun Jain 57:42
Right, right. So the logic will be once you loop entire thing, the next thing is you need to have a logic to check whether that particular file has PDF or not, right? So here what I will do is I will just use if ELE.

Ajay Patel 57:43
Yeah.

Mitesh Rathod 57:52
Hmm.

Ajay Patel 57:53
Thank you.

Tarun Jain 57:58
Uh.

Mitesh Rathod 57:58
Uh, dot. We can use that here.

Tarun Jain 58:01
dot ends with PDF. If it ends with PDF then what you can do is you can save that yearly.

Mitesh Rathod 58:04
Dog videos.

Tarun Jain 58:09
You got the point, so the syntax for logical.

Mitesh Rathod 58:10
Perfect.

Tarun Jain 58:16
Logical or you can say conditional. I'll just write conditional better.

Mitesh Rathod 58:18
OK.

Tarun Jain 58:24

Conditional list comprehension is.
You start with brackets.
Then you are for logic. Then you have if logic.
Then the axial element.
You understood, whereas the normal list comprehension is you just add the element.
And your follow logic.
Now if I print only PDF.
You will have data invoice PDF, BRD PDF, book PDF, assignment PDF.
ohh Is this clear?

Mitesh Rathod 59:08

Yep.

Tarun Jain 59:09

Cool. So now I'll proceed with tuples. Tuples is very straightforward. We will not cover indexing and slicing because as I said, indexing and slicing is same for list string and tuple as well. Now if you want to define tuple.
You just have to enclose that with you now.
Parenthesis, right? You start with the value equals to 10/20/30 right? And there is another way you you can do this as well. I'll just write.

Mitesh Rathod 59:29

No, the term places, OK.

Tarun Jain 59:43

One piece come up.
So now if you this is a simple string. I'm not even using a what you call a tuple or what parenthesis, but this is also tuple. Why? Because if you separate multiple.
Strings with comma.
Not just strings. You can also have integer or float. If you're separating them with comma then that will be considered as tuple.
So now what is the function to verify whether this particular variable is tuple or not?
Data type type of value.

Mitesh Rathod 1:00:30
Right.
Super.
Yeah.

Tarun Jain 1:00:42
Clear. Now if you look at type sorry value dot you just have two this thing, one is count and one more is index. Whereas for list we had multiple operation, we had append, we had extend, we had pop, remove and entire thing but tuple.
As only count and index. Why? Because it is immutable. For example if I do value of 0 and if I make it 11 this should throw an error.
Why? Because tuple.
Is immutable.
You got this point and if you do slicing, let's suppose value of zero to two, what should be the type of this?

Hardip Patel 1:01:26
Yes.

Ajay Patel 1:01:35
Purple.

Mitesh Rathod 1:01:36
I've got my site.

Tarun Jain 1:01:37
It should be tough.
OK, we remove this if I just keep one.
You will see 10 comma.

Ajay Patel 1:01:49
Hmm.

Tarun Jain 1:01:50

Is this clear? OK, so now if I do value dot index, I don't know why they added this function in.

Ajay Patel 1:01:51
Yep.

Tarun Jain 1:02:01
Tuple or even list because it doesn't make sense. If you give 0 index it will give you 10. Sorry.
OK, it should be 10. So if you give 10 it will give you 0. If you give 20 it should give you one. If you give 30 it will give you 2. So whenever you give index you have to give value of that particular value of that particular list element that you have and then it will print you the index.
So the input is value and once you give the value it will print the index of that element or value and then you also have.
Value dot count and if you give 30 or 20 the value is just repeated once. Let me give more values here I'll add.
1040 and 50.
Now if I do value dot count 10, it is true.
So tuple only supports 2 variable, one is index and one more is comp and whatever other logics were there indexing, slicing.

Ajay Patel 1:03:20
2.
Just want to understand like why does this? Where is this helpful index and count like?

Tarun Jain 1:03:28
Yeah, you mean indexing this value or?

Ajay Patel 1:03:32
Name count for for for tuple for tuple. I'm talking about tuple.

Tarun Jain 1:03:37

OK, tuple usually where it is used this. Let's suppose you have image pixels. OK, so image it's basically 512 comma 512 comma 3, right? So.

Ajay Patel 1:03:43
Mhm.

Tarun Jain 1:03:53
Most of the image pixels that you play with will be in tuple. So mainly what you can do is tuples are mainly used for shapes which is of fixed dimension. So if you notice the adding, tuples are mainly for fixed collection whereas.
List was for dynamic collections. Where is it?

Mitesh Rathod 1:04:14
So can we consider the enum we are using in the PHP?

Ajay Patel 1:04:18
No.

Tarun Jain 1:04:19
Oh, what?

Mitesh Rathod 1:04:21
Because it's a fixed type.

Ajay Patel 1:04:24
I don't think so.

RamKrishna Bhatt 1:04:26
I think in JavaScript late and can match this concept. So in we will not able to change the resume in late.

Tarun Jain 1:04:26
Oh.

Hardip Patel 1:04:33
Yeah.

Tarun Jain 1:04:37
Sorry, I can't hear the voice. Can you repeat again?

RamKrishna Bhatt 1:04:40
Sure. I think in JavaScript we have concept of late and constant. So we are defining variable as a late and constant. In constant we will not able to change the value and in late we will be able to change the values.

Tarun Jain 1:04:55
But that is again similar to string and tuples. But typically tuples where we are using is most of the time whenever you have shapes. If you're dealing with shapes, let's suppose you are dealing with neural network shapes or image pixel shapes, it will usually be in tuples right if you just use simple.
Let's suppose I'm using open and once I use open I'm reading certain image files. Let me just show one simple example open and you have some value called image dot PNG and now whenever you print this particular image.
I'm commenting this.
If you print image dot shape you will get output as some image resolution which is 1980.
Then 10 cross 80 and then you'll have three. So typically what this first index is, it's a width. What is the second index? It is height and what is the third index? It is channel. So whenever you deal with shapes, that is where you'll be encountering tuple. Right now what is the importance of indexing in tuple again?
What you can do is let's suppose you have image itself, right? Let me add this as image.
Image dot shape.
Is this thing. Now what you are trying to do is what is the importance of indexing. You can define width equals to image of shape 0 index, then height you can define as image of shape one.
Then channel you can define as image shape 2.
And now if you print width, you know you have fight well. Now where this is used?

Again, most of the time when you build any models you will not use the entire pixel values, you will resize it right? So if you want to resize width, you just have to change this value.
So you'll do with dot resize and you will make it as just tell you made it as 256 and you're changing that particular value where you're resizing it. But The thing is your original shape is a tuple, then you index based on whatever that logic is.
In some cases height will be first and then width will be first, second and the last value will always be a channel. So this channel is nothing but RGB which is only three and if you see four probably it will be video, right? So what happens in video is.

Ajay Patel 1:07:27
M.

TJ

Tarun Jain 1:07:33
You have RGB and then you have frame. So for video you will have four, but for images you will have three. Is this clear where tuples will be used mainly for fixed dimension shapes?

Ajay Patel 1:07:46
OK.

TJ

Tarun Jain 1:07:47
So this is 1 logic that you can do. The second logic is you can define everything in a single line, just like I did here with comma, weight, height, comma, channel. Now if I run this, let me make this 111.

Mitesh Rathod 1:07:54
OK.

TJ

Tarun Jain 1:08:03
And then you can print it.
512.
Now, this is not something that we are defining. This is already defined in this frameworks, right? So when we deal with images, right, let's suppose you open any image and when you do shape you will get this particular value. I can show one

example.
Probably this uh from PIR import image.
Copy path paste.
And now I'll just print shape.
Image dot.
And.
OK, what was the syntax?
OK, this size, not shape. Sorry.
So you saw this. What is the shape of this now? I mean data type.

Ajay Patel 1:09:39
M.

Mitesh Rathod 1:09:40
It's a it's a tuple, yeah.

Ajay Patel 1:09:41
It's a tupple.

Tarun Jain 1:09:42
It's a trouble or it is size, not shape.

Hardip Patel 1:09:42
OK.

Ajay Patel 1:09:43
Mhm.

Tarun Jain 1:09:46
The.
But you will deal with this whenever we have images. Most of the time, whenever it is fixed dimension, you'll have tuple. Now this can be width and this can be height. Is this clear?

Ajay Patel 1:10:01
Yeah.

Tarun Jain 1:10:02
But in most of the cases, yeah.

Hardip Patel 1:10:04
So is it the? Well, sorry, I just had a question like is it the right use to use tuple to return multiple data?
Back in central.

Tarun Jain 1:10:21
Oh, sorry, the voice was not audible. Oh, can you repeat here are they?

Hardip Patel 1:10:22
Hello.
OK, am I audible? Am I audible now?

Tarun Jain 1:10:30
There is bit blurriness, but you can tell I am able to hear you.

Ajay Patel 1:10:30
OK.

Hardip Patel 1:10:31
Hello.

Tarun Jain 1:10:47
Oh, can you repeat the final part?

Hardip Patel 1:10:50
I mean like, let me change the mic.

Tarun Jain 1:10:51
Can you use?

Ajay Patel 1:10:57
So one thing about I guess it's similar to list, but it's immutable, right? It's immutable list. Can we call it as a mutable list? OK.

Tarun Jain 1:11:02
Oh.

Mitesh Rathod 1:11:07
Or is the constant we can say?

Ajay Patel 1:11:09
No, it's. We cannot say it's a constant because in constant is just a key value, no.

Tarun Jain 1:11:16
So in constant in the sense you're referring to JavaScript one, right?

Ajay Patel 1:11:21
Yeah, yeah.

Tarun Jain 1:11:23
Yeah, so basically in JavaScript, if you have already defined with one, you can't define this again that variable. Here you can write in Python. The only thing is the indexing. You can't overhead. Let's suppose you have with one, right? Let's suppose this is just image.

Mitesh Rathod 1:11:28
Yes, good.

Ajay Patel 1:11:29
OK.
Mhm, mhm.

Mitesh Rathod 1:11:36
Yeah.

Tarun Jain 1:11:41
Right. And if you want to do image the index, if you want to change that, you can't do.

Mitesh Rathod 1:11:46
Yeah.

Tarun Jain 1:11:47
I want to make this 256 directly. You can't change.

Mitesh Rathod 1:11:52
Makes sense.

Tarun Jain 1:11:53
This is similar to what you have developed.
Here if you want to change the zeroth index to capital D, you can't do it because the ID is getting changed.

Mitesh Rathod 1:12:05
Do it is correct.

Ajay Patel 1:12:11
Hmm.

Tarun Jain 1:12:15
Right, so this is the only thing. List can be used whenever you have something where you have to delete, where you have to create something. That is where the list is.

Mitesh Rathod 1:12:24
Do some, uh, performance. I mean operations, you can say.

Tarun Jain 1:12:27
Yeah, right. But when it comes to tuple, you don't even have to use this separately. Tuple is mainly used in some of the frameworks or some of the packages to return the fixed shape, right? Only then the tuples is used, but in most of the cases you will never encounter tuple.
As long as it's not related to shape or size.
So images is 1, videos is 1, audio is one more. So in audio also you have different signals. What is that keyword? It's called spectrogram if I'm not wrong, right? So whenever you have to check the shapes of it or the waves of it, during that time you will get tuples.

Mitesh Rathod 1:13:01
Yeah.

Ajay Patel 1:13:01
OK.

Tarun Jain 1:13:07
This is something which is inbuilt in some of the libraries, for example size, right? That's the reason why it's usually added as speech dimension, but in terms of code you will never use tuple. You will always use lists.
Oh, there was one message from Arvind. Is it right? In this case to use multiple values by using functions in general? Yeah, you can, but usually list is preferred in this case instead of tuple.

Mitesh Rathod 1:13:25
Yeah, good question.

Tarun Jain 1:13:39
Good. Uh, so yeah.

Ajay Patel 1:13:40
And one more thing like can we have a separate values over here like let's say 200 slash 200 slash than string value in a tuple?

Tarun Jain 1:13:51
Yeah, yeah, you can have mixed. So here I just tell.

Ajay Patel 1:13:54
Can we create it 1?

Tarun Jain 1:13:57
Mixed then height.

Ajay Patel 1:14:01
M.

Tarun Jain 1:14:02
And if I print image, that's fine. So mixed data type is fine. Mixed data type is similar to what you have in this as well.

Ajay Patel 1:14:08
OK.

Tarun Jain 1:14:10
So I can do free. I will get the height.

Mitesh Rathod 1:14:19
Also, uh, one more question from my end. Can we add a list inside the tuple?

Tarun Jain 1:14:25
Yeah, yeah, new one that is possible. Image. Sorry, I'll take one better example. Cart equals to you have tuple and my first cart has happened and it costs around 75,000 and then I have total 3 products.

Mitesh Rathod 1:14:34
Hmm.
OK.
OK.

Tarun Jain 1:14:42
Right. And now I'll have.

Mitesh Rathod 1:14:45
Banana.

Tarun Jain 1:14:47
My book.

Mitesh Rathod 1:14:48
Mango.

Tarun Jain 1:14:49
Then I'll just add 1,20,000. Then I have two products so I can just print cart. Now if I do cart of 0 it will print the entire list.

Mitesh Rathod 1:14:58
Mm.
Yes, correct.

Tarun Jain 1:15:01
OK, now here I can do what is my product. So you just imagine your first index is a product, so you do 0.

Mitesh Rathod 1:15:09
There you go.

Tarun Jain 1:15:11
You will get apple. Now your second list was price. Then your last quantity you will get quantity. Same was for the second product.

Mitesh Rathod 1:15:14
Uh.
3rd is the quantity.

Also, can we do as a operation here like as we do in the list like a pop or slice anything?

Tarun Jain 1:15:33
Yeah, so you can do part of 0 the list. So here what you can do is you can do append. What can I add? I'll just add. No, probably this is tricky. I've not done this before.

Mitesh Rathod 1:15:36
Little, yeah.
Hmm.
Mhm.
SQU.
Yeah.

Tarun Jain 1:15:50
So I'll just tell.
Chroma is the providers, right?

Mitesh Rathod 1:15:55
Yeah, when we can say yeah.

Tarun Jain 1:15:56
My print card.
OK, it has added. So got one dot append.

Mitesh Rathod 1:16:00
Oh, can do that.

Tarun Jain 1:16:06
I'll add, uh, reliance.

Mitesh Rathod 1:16:08
Reliance.

Tarun Jain 1:16:12
You have the lens, so if you check the type of cart it will be tuple.

Mitesh Rathod 1:16:17
Yeah.
Tuple, but kind of 0 it will be list.

Tarun Jain 1:16:21
Your 0th index is list which is mutable.

Mitesh Rathod 1:16:25
Make sense you do it.

Tarun Jain 1:16:29
Yeah, this I've not read before, but yeah.

Mitesh Rathod 1:16:31
OK and what is like one more thing because because tuple is immutable right? So the resource ID will be can be changed if we are changing the list.

Tarun Jain 1:16:33
Yeah.
Yeah.
Change in the sense.

Mitesh Rathod 1:16:47
Like a as we just add append right? Because right now before we have apple price and the quantity but later end we are also appending as a it's a reliance or Chrome. So what is the resource ID of the card? It will be a same or it will be updated?

Tarun Jain 1:17:10
I guess it should be same. I'll do ID of.

Mitesh Rathod 1:17:15
Don't.

Tarun Jain 1:17:16
Alright.
And.
I'll put here ID of cart again, so I'll print cart. Cart is the same original to list. I'll do the ID of cart. It is 016.

Mitesh Rathod 1:17:25
Mm.
OK.
But it's not here.

Tarun Jain 1:17:37
I'm doing card one append reliance. Now if I just print card, I'll have new reliance. This should be same because in general if something is immutable, the ID of their original variable will be same just like what you saw in.

Mitesh Rathod 1:17:37
06.
OK, OK.
OK.

Tarun Jain 1:17:56
So this won't change, but the ID of your zeroth index will change.

Mitesh Rathod 1:18:00
Makes sense, yeah.

Tarun Jain 1:18:01
Oh, the least one.
So is this clear? Just to summarize again, tuples is only used in these open source frameworks where there is fixed dimension like it can be image, it can be audio, it

can be video. You'll have different sizes of it. It can be width, height, channels.
So if you encounter 4, that means you're playing around with videos. If you encounter 3, that means you have RGB channel. And if you just get two, that means it is a simple image. And if you get just one value, that means you it is probably black and white, right? So these are some of the.
Shapes and then there will also be times when you'll get 200 cross 200, then two. So I guess this is for grayscale grayscale images.
In but in most of the cases it will be just two shapes or it will be 3 or it will be 4. For audio I can't predict because for audio these spectrograms waves will be different. Based on that these values will also be different and audio is again not my domain so I don't know much about these spectrograms but videos and images.
It's three and four.

Ajay Patel 1:19:18
OK.

TJ

Tarun Jain 1:19:20
Oh, is this clear?

Ajay Patel 1:19:22
Yep.

TJ

Tarun Jain 1:19:23
OK, do we have time? Yeah, so we'll cover dictionary, then probably we'll wind up. So this is again very easy for those who works with Jason and who works with JavaScript. Here you can just define any cart again.

Ajay Patel 1:19:31
M.

TJ

Tarun Jain 1:19:39
So as I said, I'll define.
Apple.
You have 75,000, you have MacBook. OK, whatever the auto complete is there, I'll just use that.

And then you have charger which is 5000 right? So here if you notice you are using dictionary which starts with curly brackets and then you have key value pairs and this entire thing is considered as item.
So I'll just run this and what I'll do is I'll print type of instead of cart, I'll make it product price.
Now if I print type of product price it is DIC which is dict.
Is this clear? Now if I do product of price, if I do dot keys, So what are the keys? It is the first value right? So if I print keys I'll get apple.

Ajay Patel 1:20:35
Yeah.

Hardip Patel 1:20:37
Yes.

Tarun Jain 1:20:49
I'll get MacBook, I'll get charger.
Now what is the type of this?

RamKrishna Bhatt 1:20:57
And just.

Ajay Patel 1:20:58
It should be least no.

Mitesh Rathod 1:21:00
Dish Net.

Tarun Jain 1:21:02
But usually you have to do list and then you have to print it.

Ajay Patel 1:21:06
Otherwise.

Tarun Jain 1:21:06
This will be correct.

Ajay Patel 1:21:08
Dict keys. So it's a special data type Dict keys.

Mitesh Rathod 1:21:09
Yes.

Tarun Jain 1:21:11
Yeah, this is an object which is from the dictionary itself. It's not a separate data type, it's just an object. If you want the list, you can just add it inside the list.

Ajay Patel 1:21:16
OK.
OK, just one do one thing. Can you please remove change from list to DIR?
DIL is spell bracket. Sorry, no.

Tarun Jain 1:21:35
No DI for variables.

Ajay Patel 1:21:38
Yeah.

Tarun Jain 1:21:39
You'll not have any functions here.

Ajay Patel 1:21:40
OK.
OK.

Tarun Jain 1:21:45
So all these things what you see are just, uh, the meta classes.

So now after values, so after keys you have values. For values you'll have whatever is there 75,000 then whatever the prices are.

Mitesh Rathod 1:22:05
Makes it.

Tarun Jain 1:22:05
So now all the prizes are in one list, right? So you had keys.
You had values and then you will have items. So items again if I print items.
This will group them inside a tuple. If I print this if you see you have a list but what it is trying to do is it is printing Apple and then 75,000 then MacBook then 1,25,000 then you have charger and then that particular price. Now if you see this is fixed. Right, so you have the value and this is fixed. What if I need to add a new item in this? It's again similar to what you do in object. You can do product price.
I'll add color equals to red.
Now if I print product price, you have a new item, Apple, MacBook, charger and color.

Mitesh Rathod 1:22:56
OK.

Tarun Jain 1:23:04
What if you want to run a for loop for dictionary? So the for loop for list is same list and tuple you have. I have to turn off this auto complete.
For I in what was the now I wanted to show couple image shape.
So this logic of whatever you saw in list is similar to tuple, but whenever it comes to dictionary, what you're supposed to do is you have to write key, then you need to have value and then in product price.
dot items.
So now what do you have in items? You have this entire list. So now this similar to what you did in list, but the only thing is every single element in list is a tuple. So how do you remove the tuples?

Mitesh Rathod 1:23:54
Uh, double, double list.

Tarun Jain 1:24:03
By defining 2 variables.
It's this similar thing. Let's suppose you have image. How did we split this image? We split it into width, then we split into height and then channel.

Mitesh Rathod 1:24:14
Yeah.
And back we can see, yeah.

Tarun Jain 1:24:19
Yeah, unfair.
So similar thing if you look at product price if I look at this zeroth index.
Zero index.

Mitesh Rathod 1:24:31
It will be no.

Tarun Jain 1:24:33
Sorry, it should be list.
And now if I do zeroth index.
You have Apple and you have 75,000. So what is Apple? It is key. What is 75,000? It is value. So you just have to.

Mitesh Rathod 1:24:47
Mm.
OK.

Tarun Jain 1:24:54
E is E.
And then print.
Value, value and then I'll just add this design.

Mitesh Rathod 1:25:11
That's true.

Tarun Jain 1:25:16
So you have key, then you have this separator, then again key key value, key value. Is this clear how to use loops in dictionaries?

Mitesh Rathod 1:25:22
Yeah.

Tarun Jain 1:25:29
All right, So what are the different functions that a dictionary supports? I'll do DIR, then product price.

Mitesh Rathod 1:25:37
Thanks.

Tarun Jain 1:25:40
You have copy, you have clear. So again clear. It's very straightforward product price clear. So what does clear do in list?

Mitesh Rathod 1:25:51
It will return not, but it will be I can say reset.

Tarun Jain 1:25:55
Reset. That means it will return an empty list in list. If you use product price of dictionary, it will return an empty.

Mitesh Rathod 1:26:03
Dictionary.

Tarun Jain 1:26:04
Dictionary and what is an empty dictionary? Just a curly brackets.

Mitesh Rathod 1:26:09
OK.

Tarun Jain 1:26:10
So if I do product price.
It is an empty dictionary. What if I need to add new products again? You can just do product price.
Apple then 75,000.

Mitesh Rathod 1:26:27
Welcome.

Tarun Jain 1:26:29
And now if I write this again, you'll have product price.

Mitesh Rathod 1:26:34
Yeah, it's a similar kind of a Jason object, right?

Tarun Jain 1:26:38
Yeah.
Uh, similar of.

Mitesh Rathod 1:26:41
Jason object.

Tarun Jain 1:26:43
Yeah, Jason object. Similar to Jason object you have something called as get right get function. In get function what do you do? You just add some value and then you will get that particular value price. So this get command is also similar to what you have in Jason object.

Mitesh Rathod 1:26:55
Mhm.

Tarun Jain 1:26:59
Right. So what if you do product price of orange?

Do you have orange in your dictionary? No right? It will print none. But what if you want to give some meaningful response to the user? You'll say no or empty.

Mitesh Rathod 1:27:07
Which is.
No.
The different value.

Tarun Jain 1:27:21
Empty product right now if I do product price get orange and then empty product it will print empty product. So what get comment will do is it will check for the product that you need. If it is not there it will look at empty product.

Mitesh Rathod 1:27:21
Go to any Hmm.

Tarun Jain 1:27:37
So if I do apple, what it will print now?

Mitesh Rathod 1:27:40
Apple.

Tarun Jain 1:27:42
It will print the value of it, which is 75,000.

Mitesh Rathod 1:27:45
75,000.

Tarun Jain 1:27:47
So if I print product price it is Apple 75,000. Now if I do product price dot get then if I write something else which is outside.

Mitesh Rathod 1:27:49
Yeah.

Tarun Jain 1:27:58
Then it will put empty product whenever you don't find that element right? So either you can do this way or you can do product price apple.


Mitesh Rathod 1:27:59
Alright.
But.

Tarun Jain 1:28:10
But the most convenient method is to use get function because get has alternative if I use orange here.


Mitesh Rathod 1:28:18
Do you?

Tarun Jain 1:28:19
Yeah, it will throw an error because key error. There is no key, there is no right. But if you want to tackle this, if you don't want user to see the error and have a meaningful feedback, then you can use get.


Mitesh Rathod 1:28:23
He exists.
OK.

Tarun Jain 1:28:39
Fallback strategy.
Or just invalid selection also cancel.
Is this clear?


Mitesh Rathod 1:28:54
Yeah.

Tarun Jain 1:28:55
OK, let's just see other functions that we have. Product price dot copies are copies

similar to what we saw earlier. We saw get, we saw clear. Let us make the list.
What will clear do? Clear will just empty.
The dictionary.
Can we use delete?
We can use delete. If you see if I write delete of product price, the product price is 1.

Mitesh Rathod 1:29:29
This.

Tarun Jain 1:29:32
OK, so you have clear, you have delete.
Removes the element.

Mitesh Rathod 1:29:41
Thank you, ma'am.

Tarun Jain 1:29:42
Variable. What are we supposed to use after delete?
Can anyone recall what do we do after delete?

RamKrishna Bhatt 1:29:53
You see?

Tarun Jain 1:29:54
Someone ordered.

Mitesh Rathod 1:29:55
OK.

RamKrishna Bhatt 1:29:56
PC.

Tarun Jain 1:29:57
We use garbage collector. OK, something you have to keep in mind whenever you use delete in small programs, it's fine, you don't have to do it. But once you have

higher memory utilization like you you you're using vector databases, you're using embedding model.

Mitesh Rathod 1:29:59
Oh, that's correct.

Ajay Patel 1:29:59
You see?

Tarun Jain 1:30:13
And if you're hitting the limits of your RAM or usage that you have during that time, delete the unused variables and clear the garbage collector. So this both goes hand to hand, right? If you use delete, just use GC.

Mitesh Rathod 1:30:16
Yes.

Tarun Jain 1:30:29
All right, so we saw keys. So what will keys do? It will print dict keys, which is a list.

Ajay Patel 1:30:30
Mm.

Tarun Jain 1:30:40
Then we saw values.

Mitesh Rathod 1:30:42
But.

Tarun Jain 1:30:43
Which is the list of items list of key element.

Mitesh Rathod 1:30:45
Let's.
Values.

Tarun Jain 1:30:51
And then we saw items.

Mitesh Rathod 1:30:53
Weapons.
OK.

Tarun Jain 1:30:58
Tuple which contains key comma value.

Mitesh Rathod 1:30:58
OK.
M.

Tarun Jain 1:31:05
So this is mainly used for looping, used for looping dictionary.
And what is? What else did we see? We saw get command. So get command is to get the value for the given key if key not formed.
You can have a message.
So I hope you know this right. This is argument one. This is argument two. Argument one is basically the key. Argument two is basically a message that you want to show to the user if the keys if the key is not visible. So get is done items keys till it clear.

Mitesh Rathod 1:31:45
He.

Tarun Jain 1:31:56
I'll again define product price.
Update. So update. Basically what you can do is you can define Apple and you can update that particular value. This is to modify the given data.

Mitesh Rathod 1:32:10
OK.

Tarun Jain 1:32:25

So now if I use product price, you'll have the changed value of it.
And similar to what we saw in our list, you have something called as copy as well.
When are you supposed to use copy? As soon as you define a dictionary, the second line itself should be copied. Let's suppose I define product price.
I'll just define copy of product price equals to product price dot copy. So this command is there in list as well. It's also there in dictionary as well. This is similar.

Mitesh Rathod 1:32:56

OK.

Tarun Jain 1:33:05

Anyone has any doubts in clear delete keys values? This is similar to what you have in objects.
Modify the values of the existing piece.
Any questions? We'll also have a recap tomorrow once we start because we covered multiple functions. But one thing you have to keep in mind in most of the cases if you want to start with fresh values or if you want to add fresh elements which can be.

Ajay Patel 1:33:38

Mhm.

Tarun Jain 1:33:48

Added or deleted in future purpose, go with list and if you're using any frameworks which is returning any fixed shape, only during that time you'll be using tuples. For example, if you saw items is not something we defined, it's already pre-built. In most of the cases, tuple you will only encounter when it is.
Is prebuilt. You will not define it from your end and then we have dictionary dictionary. It is for key and value mapping which is similar to what you see in Jason.
So list and dictionary is something that you will use more frequently compared to tuple and set set.
It's very straightforward whatever we saw earlier right in our what we call in school syllabus. What is the union between two set? What is the intersection which we will cover tomorrow, but in most of the cases dictionary functions that you have like

update.
Then copy, clear, delete, remove. So these are some of the functions that is supported in both the dictionary and as well as list. So just to summarize, list is mutable.
Topal is.

Mitesh Rathod 1:35:00
Beautiful.

Tarun Jain 1:35:00
Immutable.
Uh, string is.

Mitesh Rathod 1:35:05
Bring it's in table.

Tarun Jain 1:35:10
Dictionary is.

Mitesh Rathod 1:35:12
Um, we have table.

Tarun Jain 1:35:12
Mutable. So a mutable in the sense you can modify or add and here also you can modify or add.
We'll cover sets and functions tomorrow. I thought we'll cover this today, but yeah, any questions we have?

Mitesh Rathod 1:35:34
No from my end.

Hardip Patel 1:35:37
So is there any way uh to uh to be more the uh a particular index norm list?
Rather than I didn't.

Tarun Jain 1:35:49
Remove the particular index from dictionary or.

Hardip Patel 1:35:52
Like right now we use value to remove any value from the list, right?

Tarun Jain 1:36:02
Yeah.

Hardip Patel 1:36:02
But what if we want to do it?
Like, uh, removing next two.

Tarun Jain 1:36:11
For list or for dictionary?

Hardip Patel 1:36:15
Police.

Tarun Jain 1:36:17
Your voice was not.

Mitesh Rathod 1:36:18
So, uh, so uh, Hardeep's question is how?

Hardip Patel 1:36:20
I mean power list.

Mitesh Rathod 1:36:23
Yeah, for at least what if we need to delete particular item via indexing index value, not from the original value?
Let's say in the list we have like a apple orange.

Tarun Jain 1:36:35

That's a.
Let me define a variable.

Mitesh Rathod 1:36:40
Yeah.

Tarun Jain 1:36:49
No, it's not giving auto complete. Got it.

Mitesh Rathod 1:36:49
Good. Uh, there is, there is, there is, there is.

Hardip Patel 1:36:51
Good.

Tarun Jain 1:36:55
Yeah, what do we need to?

Mitesh Rathod 1:36:56
OK, now now what if I need to let's say on the one index number one there is a Africa. So I want to remove like I will give a continent bracket one.
Or some index to delete that Africa from the list.

Hardip Patel 1:37:16
What if we want to remove index one?

Tarun Jain 1:37:16
OK, before mentioning.

Mitesh Rathod 1:37:19
Yeah.

Tarun Jain 1:37:21
Late continents of first index. Now if I print continents, it's gone Africa.

Hardip Patel 1:37:29
Oh, OK.

Tarun Jain 1:37:30
So can you?

Mitesh Rathod 1:37:32
OK, or the user delete deal OK.

Hardip Patel 1:37:33
OK.

Tarun Jain 1:37:37
So when it as I said right whenever it comes to mutable, if you define dictionary, sorry if you define any index which has the element, you can either remove it via delete. If delete is not working you can use remove. So again when it comes to remove right?

Mitesh Rathod 1:37:40
Mhm.

Tarun Jain 1:37:54
What usually happened? I'll use continents.
dot remove.
What am I supposed to remove? I need to remove Antarctica, but you don't know the value of it, right? So you want to delete based on the index, right? So let's define. You want to define the.

Mitesh Rathod 1:38:07
Yes, correct.

Tarun Jain 1:38:13
Want to remove the index first one right? So I'll use continents.

Mitesh Rathod 1:38:14
Index.

Tarun Jain 1:38:27
Continents dot.
Even this will works. You can write pop and you can remove that index.

Mitesh Rathod 1:38:32
Yeah.

Hardip Patel 1:38:33
Yeah.

Tarun Jain 1:38:37
.

Mitesh Rathod 1:38:38
Get the pop. We can also accept the argument of the index.

Hardip Patel 1:38:39
Oh, perfect.

Tarun Jain 1:38:41
If you if you keep it empty, then it is -1.

Mitesh Rathod 1:38:45
-1 Yeah, yeah, that was my.

Tarun Jain 1:38:47
If you give any value, it will remove that particular index. So what I was about to try was continents.

Mitesh Rathod 1:38:56
Length -1.

Tarun Jain 1:38:57
That index.
OK, this index will you have to give. Never mind when you pop.

Mitesh Rathod 1:39:03
Oh.

Hardip Patel 1:39:04
So like you are trying to fetch the value and then adding it to remove something like that.

Tarun Jain 1:39:09
Correct. So I was trying to get the index, then based on that index I was extracting that value and then removing it. So you can either do this way value equals to continents.
Then you can add this value.
So now if I print value, I have Australia. The thing is Antarctica was removed, so now you have Australia there if I print continents.

Mitesh Rathod 1:39:31
It'll be a.

Tarun Jain 1:39:41
You have Australia second, so now you have your index that you want to remove.
You extracted the value, then you will use remove.

Mitesh Rathod 1:39:50
Mhm.

Hardip Patel 1:39:53
Yeah, I think I got it.

Tarun Jain 1:39:56

And now if I again print this continents, the Australia is gone.
You got it, so I'll just remove.

Mitesh Rathod 1:40:03
Yes.

Hardip Patel 1:40:04
Yeah, yeah. Thank you.

Tarun Jain 1:40:11
One is we have delete then list the particular index.
And then we had.
List dot remove.
Here what you can do is you can write list of that particular index.

Mitesh Rathod 1:40:32
Because is a list dot pop.

Tarun Jain 1:40:34
So what this particular index will do is it will get the value and once you get the value you can remove the particular element. Then you have dot pop.

Mitesh Rathod 1:40:43
Element list and index number.

Tarun Jain 1:40:47
That particular.
Oh, anything else?

Hardip Patel 1:40:57
No, I couldn't get the check.

Tarun Jain 1:41:00
So The thing is tomorrow we can have session from 11:00 to 2:00 because I have to travel to sorry 11:00 to 12:00 because I have to travel to Hyderabad. So we can have

just one hour session tomorrow. So Ajay, can you just coordinate with Nida if I'm not wrong?
Uh, I guess that.

Ajay Patel 1:41:16
Yeah, sure. I will. I'll let her know.

Tarun Jain 1:41:19
Yeah, only on 13th and 14th one one hour 15th we can again have one and 1132 one.

Ajay Patel 1:41:22
OK. OK. OK. OK. OK. No worries.

Tarun Jain 1:41:28
So 13 and 14 it's 11 to 12, just one hour.

Ajay Patel 1:41:31
Yeah, I guess everyone. Yeah, yeah.

Tarun Jain 1:41:32
OK, so just some practice of this. We'll again have some quiz tomorrow just so we covered some of the important functions. List and dictionary is very important. Tuple if you know indexing, if you know slicing, tuple is very easy, but list and dictionary sometimes can get tricky.

Ajay Patel 1:41:38
OK.
OK.
OK.

Tarun Jain 1:41:51
Good. Anything else before we wind up?

Ajay Patel 1:41:52

OK.
No, I guess that is fine. We can bring to close, but.

Mitesh Rathod 1:41:56
Enter.

Tarun Jain 1:41:58
Oh, yeah. Again, thank you everyone.

Hardip Patel 1:41:58
I have completely unrelated question. Sorry, sorry, I just have completely unrelated question about something about tick token. Is it OK to ask?

Tarun Jain 1:42:03
Oh, what happened?
Oh, can you repeat again?

Hardip Patel 1:42:14
So so I have the question about the token. I started using GPT 5. Is it OK to ask this or I don't know?

Tarun Jain 1:42:25
For some reason I'm not sure. I'm not able to hear you clearly.

Ajay Patel 1:42:28
No, no, even we are also facing challenges. Hardip, can you write it down in the chat?

Hardip Patel 1:42:31
OK, can can you? I I I think I I will DM you or something or I I will ask next time tomorrow.

Ajay Patel 1:42:38
Yeah.

Tarun Jain 1:42:38

No, yes, no.
Yeah, OK, you can also DM. I'll reply to that. Oh, is it possible to that? Like how do we create chat of the existing people if in case I want to share some?

Hardip Patel 1:42:46
OK. Thank you.
Do you have teams? Can you tell me to do teams?

Ajay Patel 1:42:53
Yeah, I mean.
Just go into teams.

Tarun Jain 1:42:58
Uh, one second. Yeah, I'm sharing my screen.

Ajay Patel 1:43:00
Hello. Yeah.

Tarun Jain 1:43:09
Where should I go?

Ajay Patel 1:43:11
OK, so in a meeting chat we teams.microsoft.com.
OK.

RamKrishna Bhatt 1:43:20
I think on right hand side it is already there.
Did a work call me?

Tarun Jain 1:43:29
The URL is correct, right?

Ajay Patel 1:43:31
Yeah, you are is correct.
OK, sign into your hmm.

Tarun Jain 1:43:38
Here I've signed in on different account.


Ajay Patel 1:43:41
OK.

Tarun Jain 1:43:42
So yeah, OK, I got it. What do you want? Yeah, I'll do that.


Ajay Patel 1:43:44
You have to use OK and over here you can see the lot of I mean current conversations, current meeting chat over here you can see in the chat but.

Tarun Jain 1:43:54
So should I use my same Gmail that I'm using for calendar? OK.


Ajay Patel 1:43:58
Yeah, yeah, yeah, yeah, please, please use the same Gmail Tarun Jain, Tarun Jain one.

Tarun Jain 1:44:03
OK.
It.
Yeah, this I'll do. I got it now.


Ajay Patel 1:44:08
Yeah. OK. OK. Thank you, Tarun. Thank you for the class.

Tarun Jain 1:44:13
Yeah, just log in from tomorrow. It will be just one hour. We can instead of 11:30 we can keep it 1111 to 12.


Ajay Patel 1:44:15
Yeah, yeah.
OK, OK, OK, 11 to 12 tomorrow.

Tarun Jain 1:44:22
Yeah.
11 to 12 on 13th and just two days because of I have other calls just after the meeting, that's why.

Ajay Patel 1:44:24
OK.
OK, OK.
OK. OK. Bye, Taran. Bye.

Tarun Jain 1:44:32
Yeah. Thanks.

Mitesh Rathod 1:44:33
Yeah. Thank you. Bye. Thank you.

RamKrishna Bhatt 1:44:34
Thanks.

Tarun Jain 1:44:34
Bye.

Mitesh Rathod 1:44:36
Mhm.

Hardip Patel 1:44:37
Thank you. Bye bye.

Margi Varmora stopped transcription