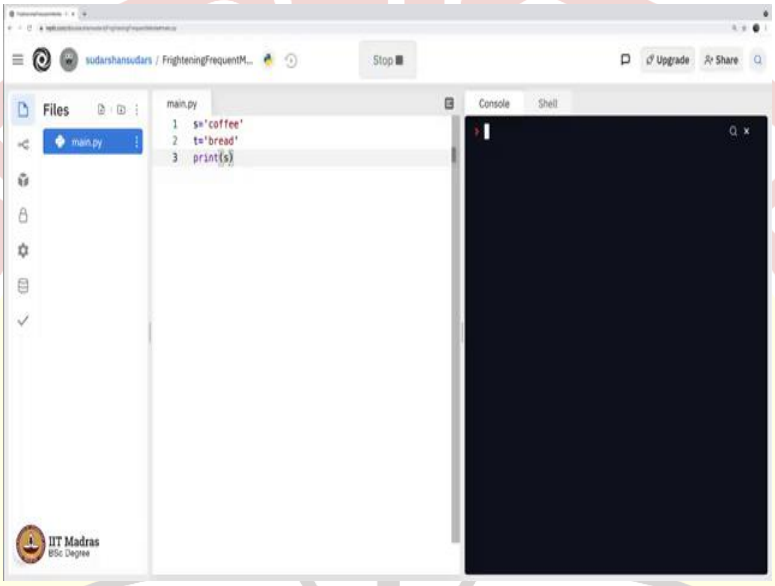
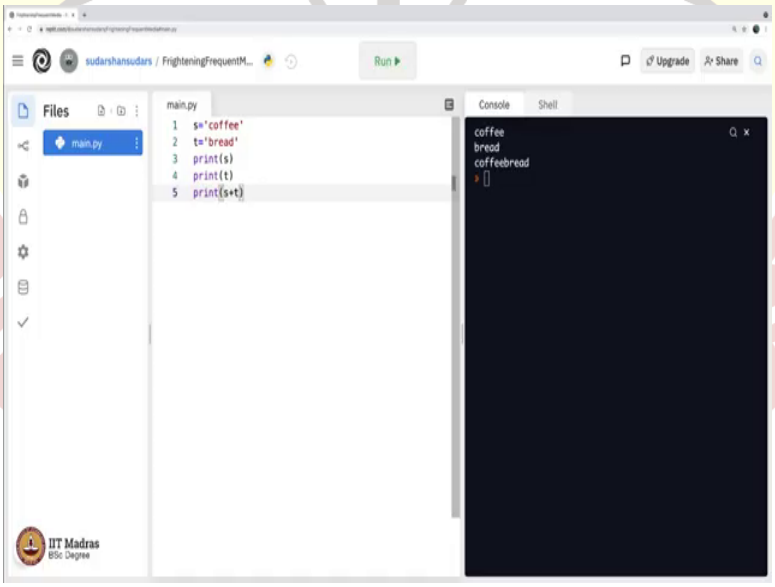
IIT Madras

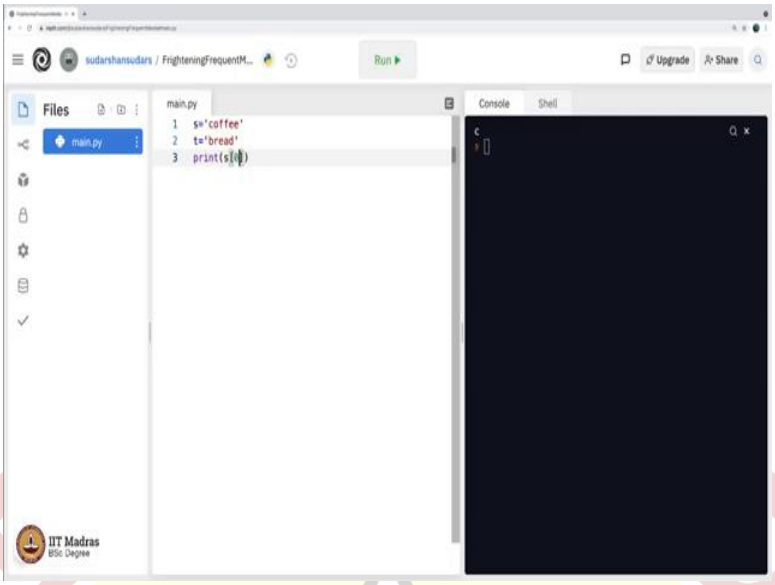
ONLINE DEGREE

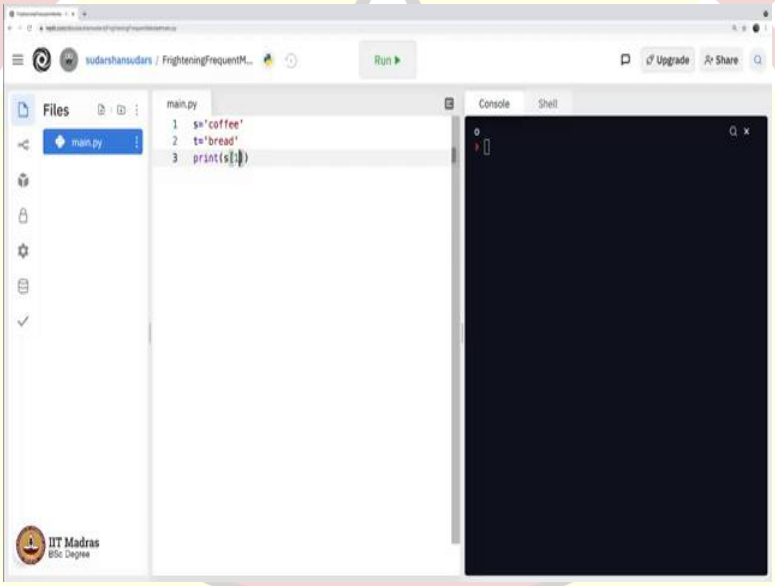
Programming in Python
Professor Sudarshan Iyengar
Department of Computer Science and Engineering
Indian Institute of Technology Ropar
Omkar Joshi
Course Instructor
Indian Institute of Technology Madras Online Degree Program
Introduction to Strings

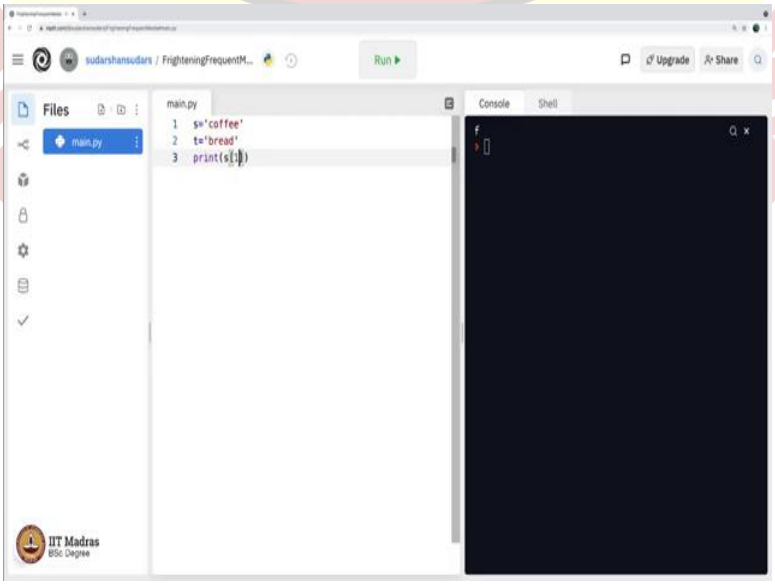
(Refer Slide Time: 0:16)

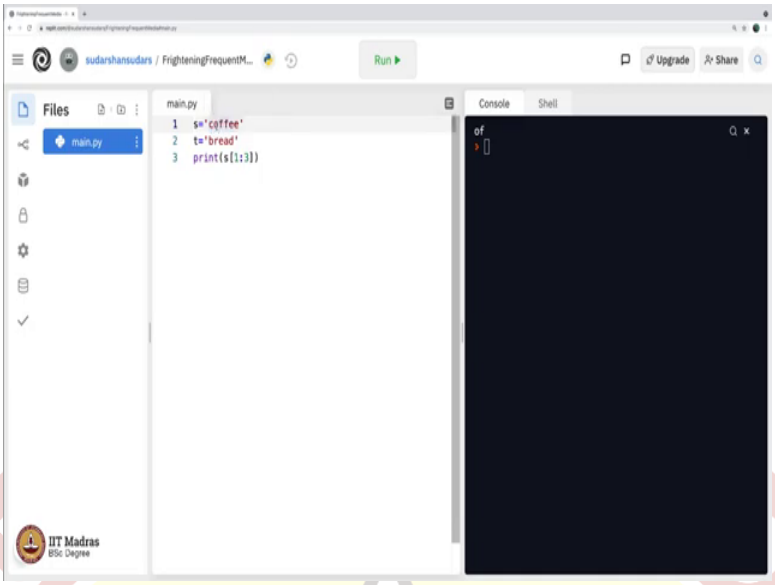


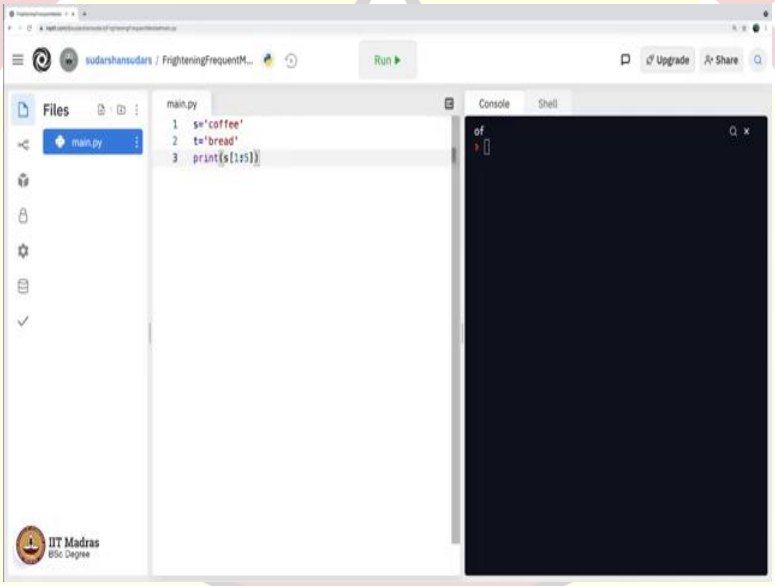


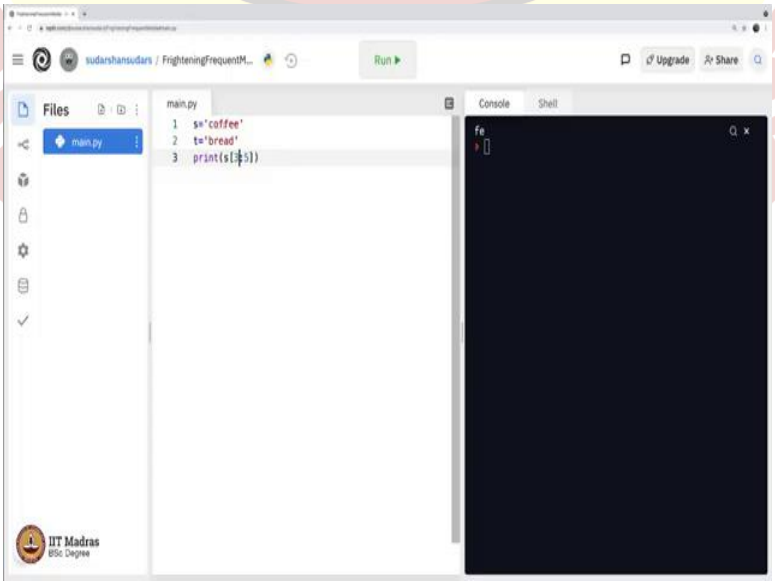

sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
1 s='coffee'
2 t='bread'
3 print(s[0])
Console
Shell
c
IIT Madras
BSc Degree

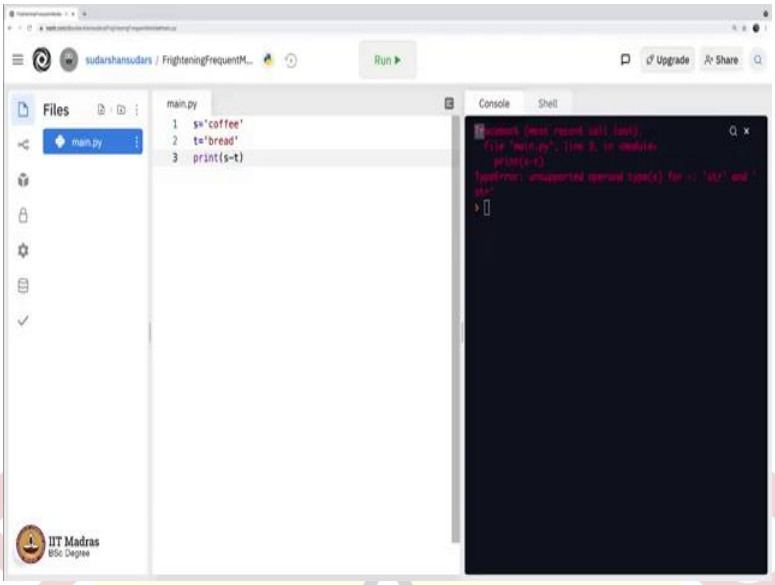

sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
1 s='coffee'
2 t='bread'
3 print(s[1])
Console
Shell
o
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
1 s='coffee'
2 t='bread'
3 print(s[1])
Console
Shell
f
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
main.py
1 s='coffee'
2 t='bread'
3 print(s[1:3])
Console
Shell
of
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
main.py
1 s='coffee'
2 t='bread'
3 print(s[1:5])
Console
Shell
of
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
main.py
1 s='coffee'
2 t='bread'
3 print(s[3:5])
Console
Shell
fe
IIT Madras
BSc Degree



We did see more on strings, but let us revisit strings now and start with declaring a variable s as let us say coffee and then t will be, what goes good with coffee, maybe cake or better even bread, I want something long word. So, you all know what happens when I print s, will be printing coffee, when I print t I will be printing bread, correct, coffee and bread and as we discussed when we print s plus t it will concatenate, that is probably a new word for you all.

It will concatenate, which means it will place one next to the other and display coffee bread. Right, fine now let me take you all to one level deep into strings, so the next level would be to understand what will happen if I were to simply say s of 0, even this we discussed, it will show you the first letter of s, s of 1 second letter of s, if I were to say s of 2, third letter of s and so on. But then what I did not tell you people is that if you specify s of 1 here and then 3 here it does something else.

Let us see what it does? It is showing ‘of ’. What exactly is it doing? It is taking, it is starting from the first number 1 s of 1, it is starting and going up to 2, it does not go up to 3, it goes one less, s of 0 is c, s of 1 is o right, s of 1 is o, and s of 2 is f, so it starts from s of 1, and displays this, what if I said 5 here, it will start from 1, s of 1, 2, 3, 4, it should display so much.

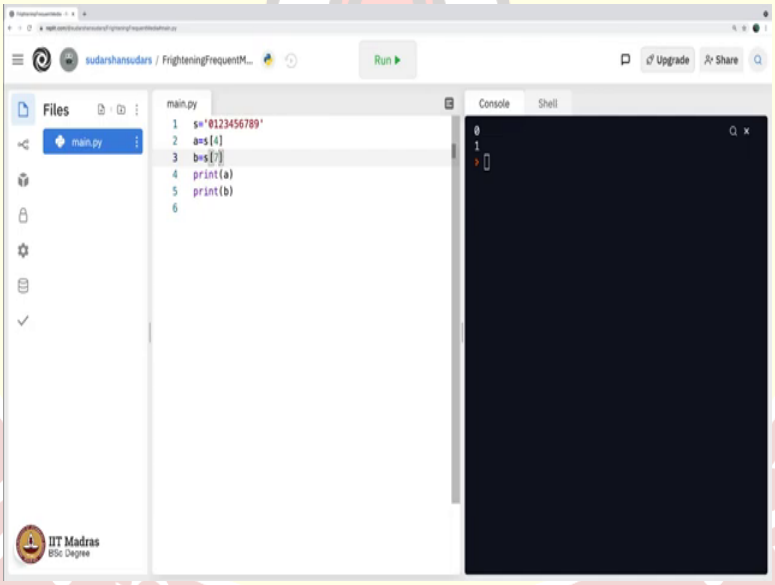
You may be wondering why does not it go up to 5 that only makes sense for it to go up to 5, for some reason Python is created this way. There will be many such questions in your mind. Why did Python do it this way? The answer invariably is always, it was their choice; there was probably some reason that we may not know today or maybe you can Google it up and you may know why exactly they have done it like this.

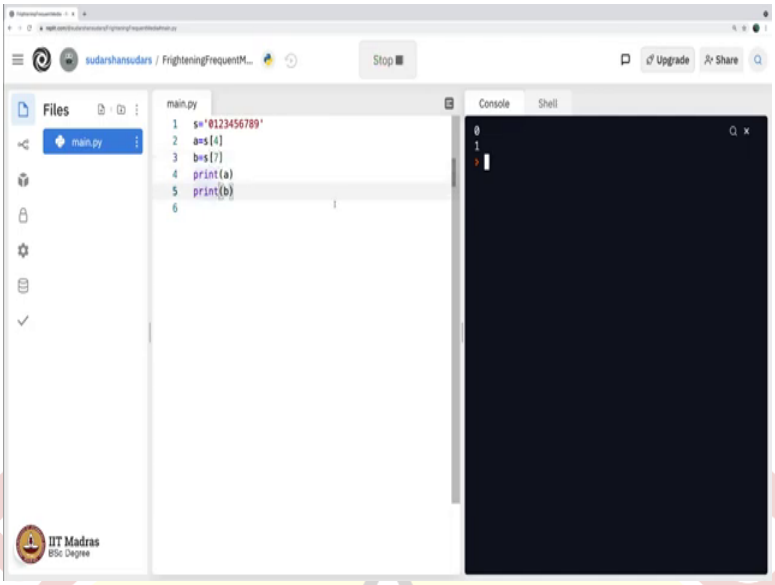
But by the way there are very nice reasons why Python starts with 0. So, does your C language. Anything before c with the best of my knowledge would always start with 1, the list would start with 1, the string would start with 1, you would refer it as s of 1 for c, but in Python you start with s of 0 for the first letter.

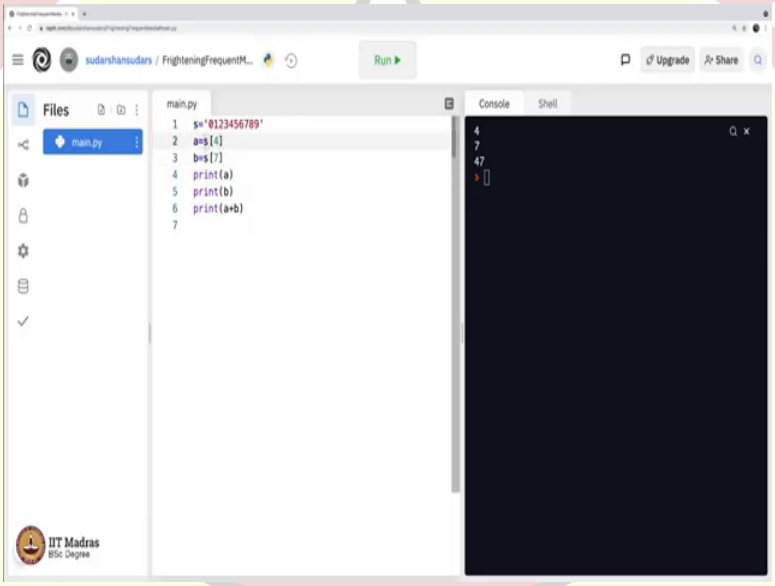
So, what will this output? This should output start from 1 go up to 5 minus 1, which is s of 4 this should be the output. Let us see if my, yes, my prediction is right, it goes up to 5, if you let us say start from 3 and go up to 5 what will happen, 0, 1, 2, 3, and go up to 5, 5 minus 1 which is 4, 'fe' should be displayed that is indeed displayed here.

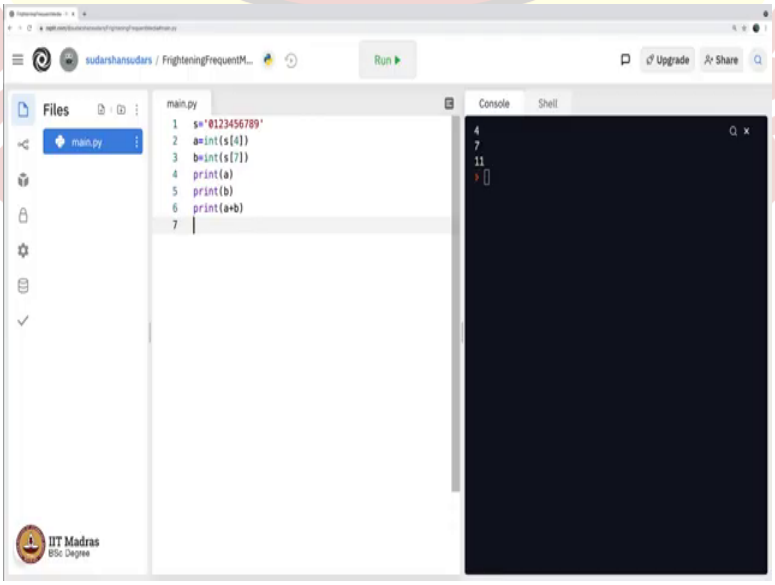
So, you sort of, you can slice the strings as you can see. This is called string slicing. So, let me go ahead and then play around a little more with the, with these two words, I said s plus t will give you concatenation as explained before, s minus t will throw up an error because it has no clue what to do with subtraction of two strings.

(Refer Slide Time: 4:05)




sudarshansudars / FrighteningFrequentM...
Stop
Upgrade
Share
Files
main.py
main.py
1 s='0123456789'
2 a=s[4]
3 b=s[7]
4 print(a)
5 print(b)
6
Console
Shell
0
1
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
main.py
1 s='0123456789'
2 a=s[4]
3 b=s[7]
4 print(a)
5 print(b)
6 print(a+b)
7
Console
Shell
4
7
47
IIT Madras
BSc Degree


sudarshansudars / FrighteningFrequentM...
Run
Upgrade
Share
Files
main.py
main.py
1 s='0123456789'
2 a=int(s[4])
3 b=int(s[7])
4 print(a)
5 print(b)
6 print(a+b)
7
Console
Shell
4
7
11
IIT Madras
BSc Degree

Now, to put everything in context I will type a nice piece of code and I will let you think on what exactly will this output. What is s? s is a string 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and what is a, a is s of 0. What am I trying to do here? You probably are confused, you always type something on the right side which is assigned to the left side right, you never put another variable here on the left side, this is indeed okay, you can put whatever you want on the right side, its value gets assigned to the left side.

It will take some time for you to get used to it, but commonsensically speaking a is now assigned s of 0 which you would have guessed is this 0, b will be assigned s of 1 which you would have guessed is 1. Now, when I say print a and then print b right, you will see the obvious happen here. It is taking some time, but what do you expect to see. You will see the value of, this is a usual problem with Replit.

I do not want to you know edit this part, but I want you all to take this warning message that it, if it so happens that it says fail to connect, it means there is some problem with your internet, you may have to wait for a second and then it gets connected, you see the output here, 0 and 1. Perfect. If I were to say s of let us say 4 and 7, it will say 4 and 7. Print a s of 4 is 0, 1, 2, 3, is indeed 4 itself, the number 4. So, it will say this.

Now can you all guess what will the output of this be, a plus b? Do you expect it to say 11? Let me see, I would expect as a layman, as a person who is new to programming, I would expect this to be 4 plus 7, 11, but if you have watched carefully from what we were discussing a minute before, you will say no it will not be 11, so let us see what the answer is, as I execute it I see it is 47, 4 plus 7 is 47.

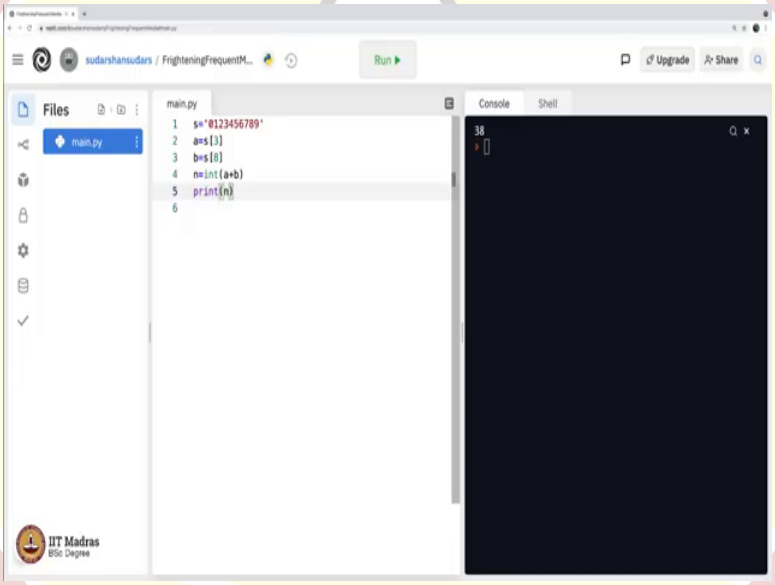
But why on earth would it say 47? That is because as you can see a here is actually a string, a one letter string, a character, b is another string, plus understands that it should take this string and this string and place them one next to the other, the concatenation as I have been talking to you people about, it will not see them as numbers, unfortunately and I have been telling you people computer does exactly what you instruct the computer to do.

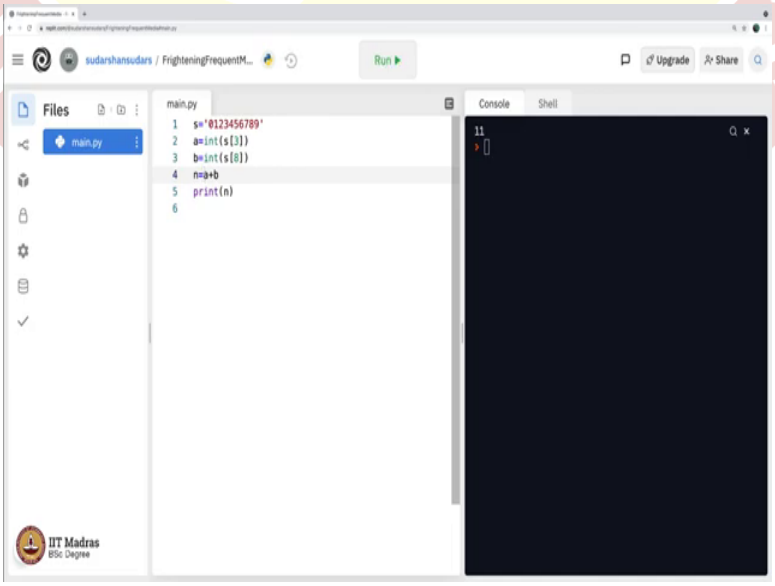
It does not have the intelligence to add things up if they are numbers. So, what do we do to make these make 4 plus 7 equals 11 here? That is going to be pretty simple, all you need to do is when you say a equals s of 4, you must say convert that to an integer, when you say b equals s of 7, you must say convert that to an integer, a very simple trick. And now finally when you say print, it will say 4 plus 7, simply because the plus here is acting on an integer a and an integer b, unlike before where both of them were strings.

If they are strings, plus this addition of strings means place them one next to the other, when you put plus between two numbers, it means do addition, the old school style, whatever you were doing in your school, 7 plus 4 was 11, but if there are strings, 7 plus 4 will be 7 4 placed next to each other. What is the moral of the story? The more the story is the way your computer sees things depends on how you store them and computer stores in its own way unless or otherwise specified.

You may have to specify in what format you want this to be stored just so that you can do the necessary operations on it. I hope things are clear, if not you may have to practice little bit on what I said just now. I will now try to give you a small program and I leave it to you people to figure out what is the answer for this code.

(Refer Slide Time: 8:31)





What if I said a equals s of 3 and b equals s of 8 and I will print, let us say, I will say n equals a plus b and then convert this to int, convert this to int, and then I will print n, what will be the output? The output is 38 as expected, correct? But if I were to put int here and an int here what is the output? I will remove the int here. What is the output? The output is 11.

Simply removing the int here will give you some output and retaining the int there and then the output will be something else. So, you can you can redo this code all by yourself and try to understand the motivation for data types.