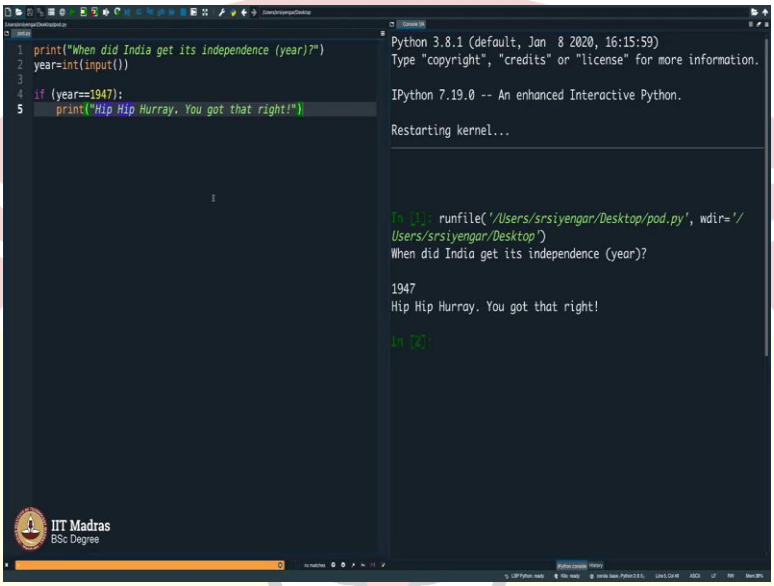
IIT Madras

ONLINE DEGREE

Programming in Python
Professor Sudarshan Iyengar
Department of Computer Science and Engineering
Indian Institute of Technology, Ropar
Introduction to While Loop

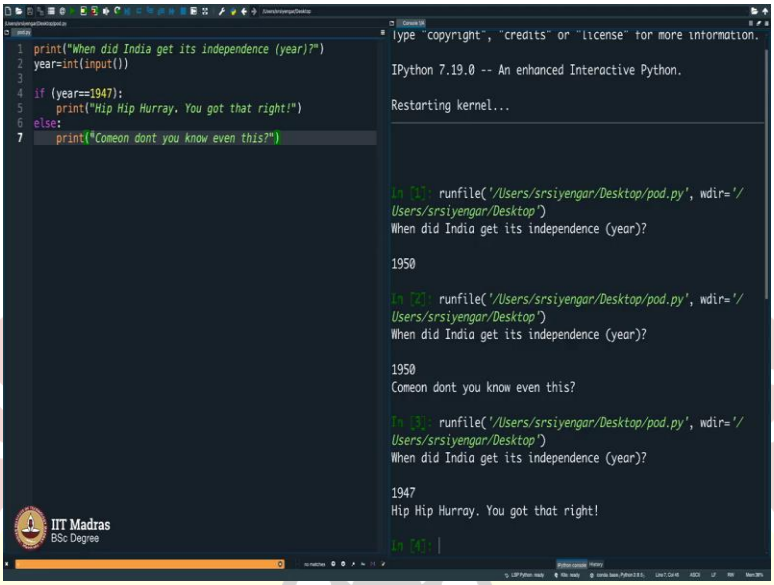
(Refer Slide Time: 00:15)



Let us try coding a quiz program, a piece of code that will simulate a typical quizzing like situation. It will ask a question and then check if the answer is right or incorrect. So how does it go? Let me use the simplest possible question which I am sure all of you would know the answer for, when did India get its independence? So, I am talking about the year here, hoping that you all know int input, you know, you all know what I am doing here.

I am just taking the user input and assigning whatever the user inputs to the variable year here and I convert that to integer. I hope you are understanding the previous two weeks is in place and you are able to see what we are doing here. And what should I do? I should simply say if year is equal to 1947, print Hip Hip Hurray, you are right, you got that right. So let us make it cinematic. So Hip Hip Hurray. You got that right as simple as that. I will execute this. What happens? When did India get its independence? 1947, it says Hip Hip Hurray. You got that right.

(Refer Slide Time: 01:55)

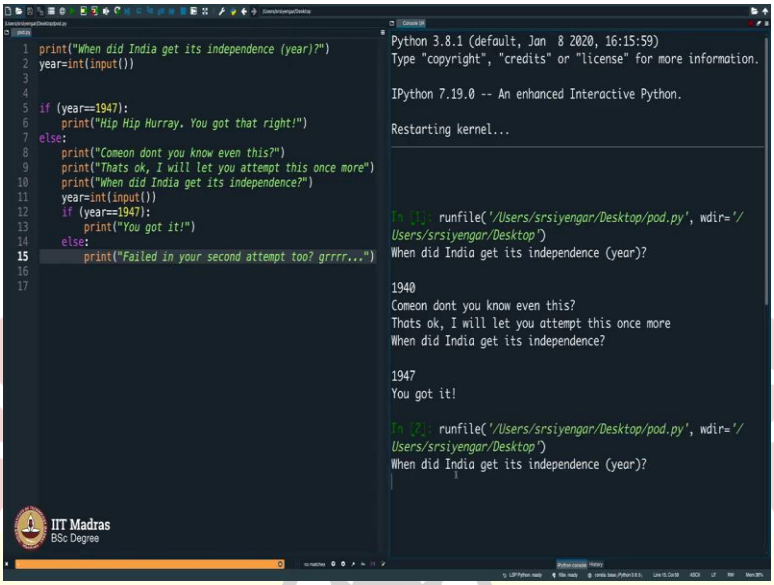


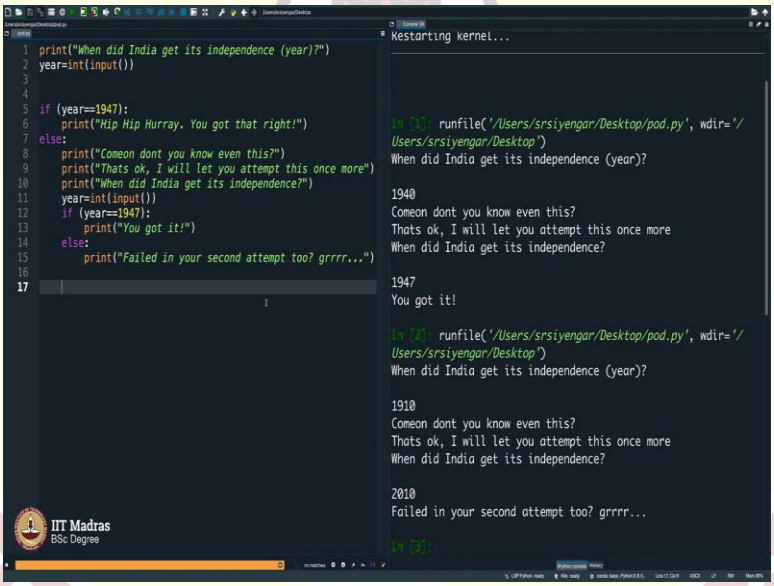
I will execute this once again, tell me what will happen in case I did enter incorrectly, assume I type 1950 in place of 1947. What would happen? 1950 would simply take me out of this if loop of course the if loop gets executed but whatever is inside does not get executed. It comes out and does nothing, it does not congratulate you the way it did the last time when you got it right.

So let us abuse the user, we are saying come on do not you know even this be being an Indian. Anyway, it is all for fun. No offense. So, let us see what happens here. When you say when did India get its independence? I say, 1950. And I get yelled at by the computer. Come on do not you know even this. And in case I get it right, the computer will say 1947, Hip Hip Hurray. You got that right. You probably are wondering, what is new in this week?

We are repeating the if statement that you learned the previous week, there is an important point that is coming up. I would like to ask this question; do not you think everybody requires a second attempt if something goes wrong. The moment if someone gets it wrong, they do not enter 1947. Of course, you can hurl and abuse and yell at the user. But then you must say, do you want to attempt once again? So how do we do that?

(Refer Slide Time: 03:35)



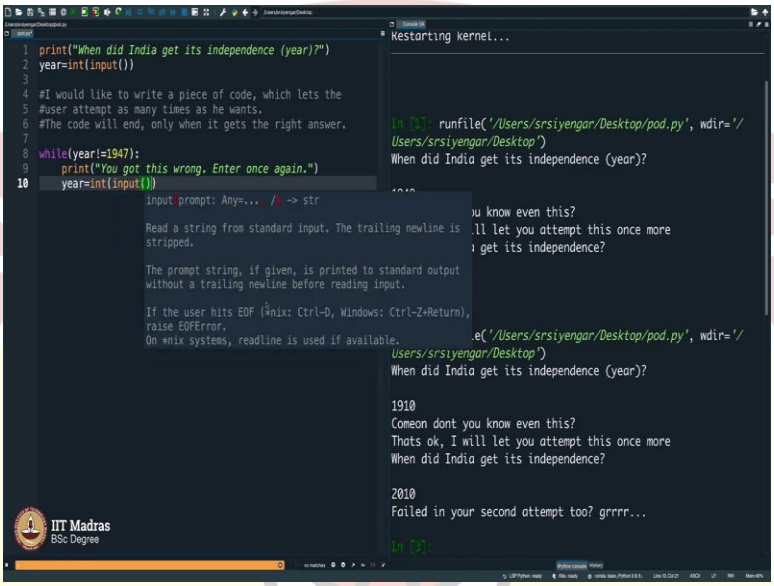


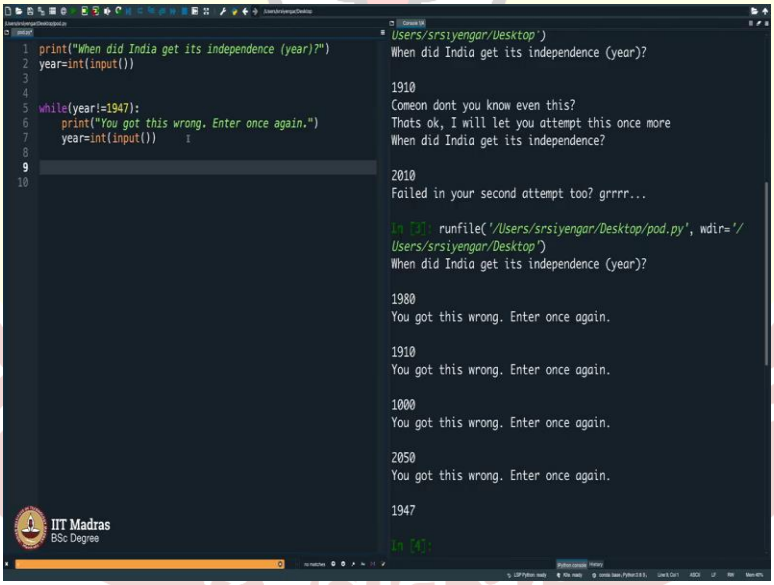
How do we do that? Let me write a code for that? Let me think how that is done? I will say do not even know this, I will say print that is okay. I will let you attempt this once more. And then again, I say, print when did India get its independence? And then once again I take the year as simple as that int input. And then again you see programming can get a little boring at times. If year is equal 1947, I will say print, you got it else failed in your second attempt too.

We'll see what it does. When did India get its independence? The year 1940 incorrect. Come on do not you know even this, that is okay, I will let you attempt this once more. When did India get its independence? I say 1947 and it congratulates, correct? Let us start up fresh, once again.

When did India get its independence 1910, no 2010, failed in your second attempt too? So now what do you do? Do you think we should write a piece of code to let that give you another attempt? So here is the deal. I would like to write a piece of code.

(Refer Slide Time: 05:30)





So let me remove this. I would like to write a piece of code that lets you attempt as many times as you want until you get it right. So if you write that down. I would like to write a piece of code which lets the user attempt as many times as he wants, as he wants the code will end only when it gets the right answer, wonderful. How do we do that? So, there is a difference between the word if and the word while.

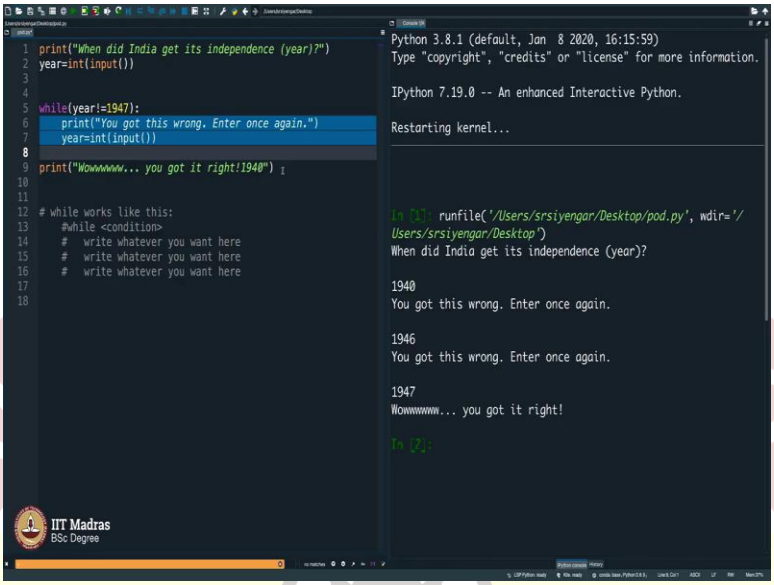
When do we use if? We use if to say, if that is true then if it does not rain then I will go and play. That is different from while it does not train, I will keep playing. So, you will, you will get this in a minute's time as I write a piece of code. I will not explain it. I will write and show you what this does and you rather should tell to convince yourself what exactly this is doing. Can say while year is not equal to 1947, I will say you got this wrong. Enter once again. What is happening here? Let us check this out. What is happening here?

Let me remove the comment just so that you see only the part of the code that is interesting. When did India get its independence? 1980, you got this wrong enter once again, 1910, you got this wrong enter once again, 1000, you got this wrong enter once again. We have not got an independence; we probably will get in 2050 you got this wrong enter once again. Let say in 1947 and then it exits, exits, but it did not congratulate me, did not say Hip Hip Hurray.

So where should I include this Hip Hip Hurray, so that once I get it right it says Hip Hip Hurray. Where should I do that? A little bit of thinking says before that. I believe you all understood what exactly happened here.

These two lines they keep getting executed until you get this right, until this is true. If this is true year is not 1947, then do this. Year is not 1947, then do this. After this do not come down and then end the code. After this go back and check if the year is not 1947. You make a mistake, I will make you do this again, you make a mistake I will make this do once again, make a mistake I will make this do once again.

(Refer Slide Time: 08:31)



You get this right, then I will not enter year, I will end here. When I end here, I will say print wow, you got it right. Let us see if this works. Clearing the screen. So, I will start executing the program. When did India get its independence? 1940, sorry, 1940, you got this wrong, 1946 you got this wrong, 1947, wow you got it right. So you see what this code is doing? Let us go over it once again I am sure most of you understood it already.

But then people who are seeing programming for the first time in your life and in this course, while is not so very obvious. We want to put this on your monitor and keep staring at it for like 15-20 minutes until you get it right. First line, when did India get its independence? Second line, you take the input and this the next line, the fifth line here, is while the year is incorrect you keep doing this, tell the user that it is wrong and ask him to input once again.

If he enters incorrectly, the computer will go back here. Now you may ask me why does it go back there? Sometimes it come goes ahead, sometimes it goes back, in if it goes ahead, it never goes back. But in while it goes back, why does it do it that way? That is how it is it is made to do, that is how programming languages are written. Whenever you put a while, whatever is inside while gets repeated until the point when the condition here is incorrect.

Let me just summarize that here or a comment while works like this. You say while and you put a condition here and then you have a tab here, this comment in this, you have a tab here write whatever you want here. Similarly write whatever you want here, write whatever you want here,

write whatever you want here, whatever you write here will get executed. This keeps getting executed as long as the condition is true.

So here what is the condition? The condition was that year should not be 1947. The fact that the year is not 1947 became true and that is why this got executed multiple times. A little complex there, there is a true and a false here, the condition is true that the year was incorrect. It gets executed again and again and again until the point that you get it right. So, this is like a very patient teacher you have programmed a very patient quiz master where he will wait until you get the right answer.

He does not get frustrated, he will have, he will not have a mood fluctuation, he will not go angry against you. That is the advantage of having a machine do the job of a quiz master or for that matter any other task, there is no emotion involved here. It will not go angry; it will not get frustrated. I hope you enjoyed this piece of video, while loop is very important. It is used most often than not in fact as I as I keep saying the for loop, while loop and if you will see them in every single code more or less.

And this is where you make your computer repeatedly do a piece of thing again and again and again. In fact, this will do it over a million times if you get it wrong, correct? So let us, in this week we will be seeing more of these looping structures, we will see while loop, we will see for loop. And we will write some complicated code using these two things and solve many problems that is of interest to you but then a word of warning you may find it a little difficult. But as assured always, I have been assuring you from day one it is all a matter of time. So let us go to the next video.