Social Networks
Prof S. R. S. Iyengar
Department of Computer Science
Indian Institute of Technology, Ropar

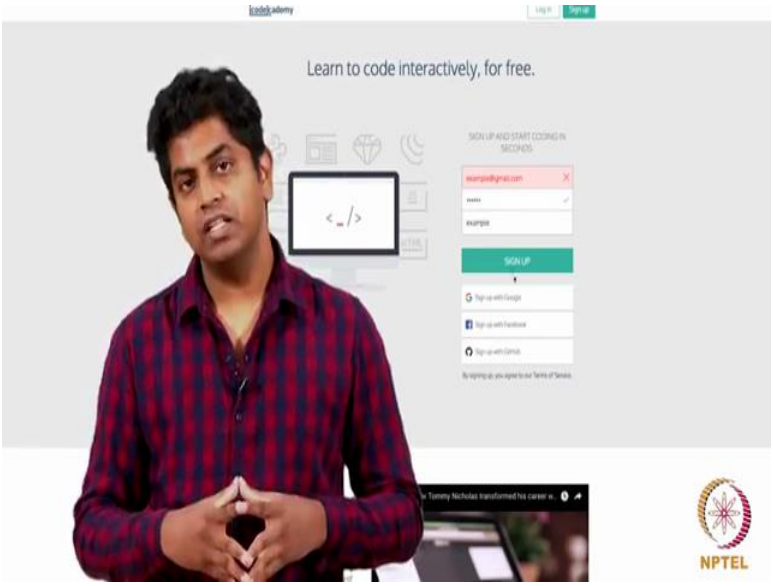
Lecture - 03
Introduction to Social Networks
Introduction to Phyton-1

Hi, in this module, we will be introducing you to python. Python is a very powerful programming language yet very simple to understand, very easy to use and it is an open source package. It is a scripting language which has a whole lot of APIs online and it is very easy for you to install these APIs and work with python this is one major reason why I thought you will use python for our network analysis related work social network analysis related work.

So, in this session I will be showing you some quick ways of learning python; quick things that you must know about python and you can get started. It is not required that you undergo a complete course in python and then start using python 3-4 things and you will and you are on you can start using python.

So, my personal recommendation, my personal favorite is this code academy dot com where you can go create a user name then start learning python, it has very nice exercise unless you solve the exercise, you cannot go further and these exercises are in increasing difficulty.

(Refer Slide Time: 01:00)



I strongly suggest that you finish a few lessons there and get yourself familiarized with python.

And or you can just look at what we do on the screen cast which I am going to show you right now and learn python parallely. Now a word on installing python, if you are using o one 2 the process is pretty straight forward on MAC again it is again very easy to install.

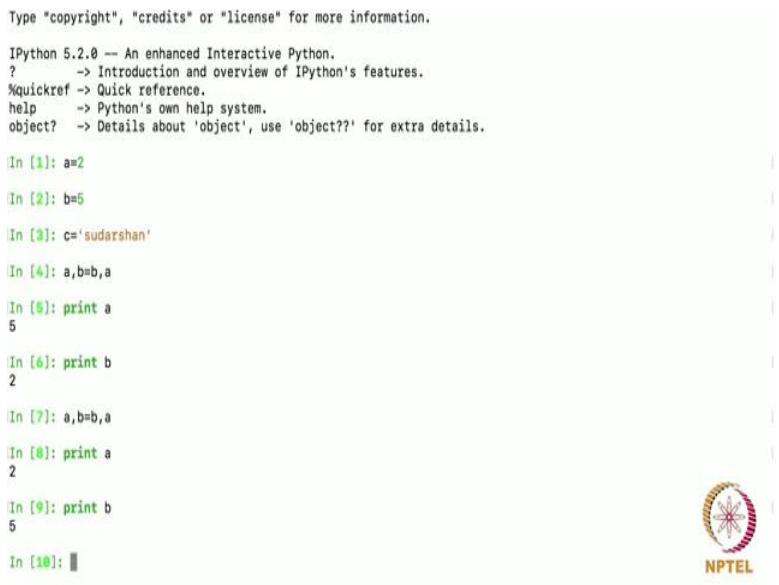
(Refer Slide Time: 01:41)

But if you are using windows, I would suggest that you download what is called anaconda which is a big package which includes all the scientific programming

requirements using python and once installed I think you are all done, but one o one 2 one MAC it is a very straight forward process.

For our demonstrations we are using a MAC machine and this is the terminal that I have opened on a MAC machine. So, what I do is I type ipython. So, this ipython is interactive python which opens a shell like this where I can easily type the code.

(Refer Slide Time: 02:15)



A note on IPython it is very easily installable on o 1 to 1 MAC, but on windows as I told you need anaconda in which ipython is a built-in package.

In ipython let me try showing you a few things a = 2, this is basically python with a interactive shell and then let say b = 5. So, unlike c you do not have any declarations that are required for variables you just say a equals to b equals 5 and it assigns a to 2 and b to 5 and if I say c equals let say Sudarshan. Now c will be string type variable automatically and look at the ease with which I can swap to variables a comma b is equal to b comma a and then if I print a and print b the values are swapped, let me swap them once again back to normal if I print a it will be 2 if I print b it will be 5.

We will now see how one can use for loop while loop and if loop in python. It is pretty straight forward I will still give you an illustration I would suggest that you try getting your hands on with these things so that you get familiarized with it, so a for loop works like this.

(Refer Slide Time: 03:41)

Say for i in range 10 I say print I, it is simply a prints i from 0 to 9, this is how a for loop works, a while loop also works in a similar way if I say k equals let us say 10 and I say while k is less than or equal to 20 do print k and then increment k just starts k with 10 and goes on till 20.

(Refer Slide Time: 04:13)

So, the value of k now is 21, what I will do is I will illustrate if loop now using this k value, if k is greater than 20 which by the way is the case k's value seems to be 21, I will say print the value of k is greater than 20.

So, this is true. So, it displays the value of k is greater than 20. So, if I say if k is greater than 25 then print something something I am illustrating the usage of else here I will say else, print the statement is incorrect is false let say is false let say the statement is false right this is how if else loop works. So, we now just now say how for loop works while loop works and if loop works; so just that you people play around with it so that you get familiarized with it.

A big advantage of using a scripting language like python is it comes with a whole lot of library functions I am now going to show you one such library function which will be using a whole lot throughout our case.

(Refer Slide Time: 05:41)

Let me use a library function called random this is how you input if you load the library function to your memory, you say import space random and then in this function you can call in this library you can call a particular function let say rand range one comma 10 what does this do this simply gives you a random number from 1 to 10, let me execute this command once again.

(Refer Slide Time: 06:39)

You press the up arrow in ipython shell and the entire things comes press enter, you will get another random number from the range 1 to 10 again you see I am getting random numbers from 1 to 10.

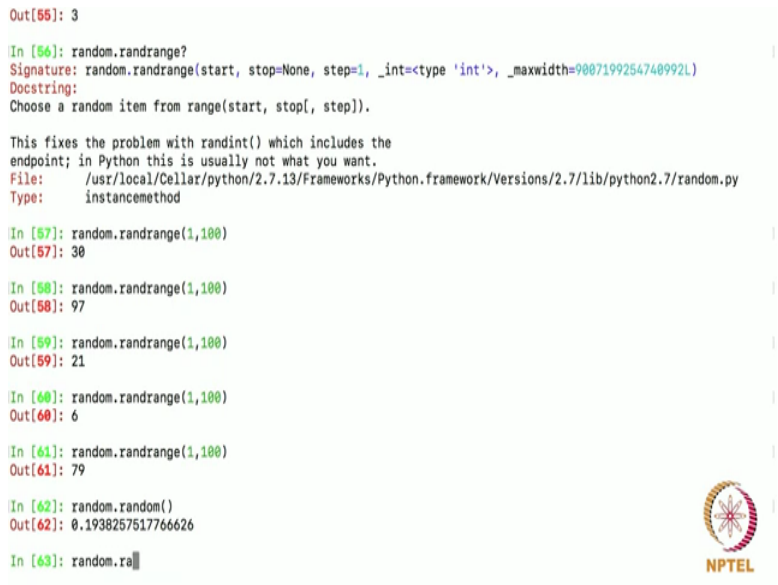
(Refer Slide Time: 06:44)

Let me now change this instead of 10, I will make it 4, this will give you random numbers from 1 to 4, 1, 2, 3 and so on.

(Refer Slide Time: 06:53)

Yeah, you will get 4 2 as you keep trying or maybe it is only till yeah it is only till. So, you will not get 5 you will only get the number just before. So, random dot rand; rand range I do not know what it does. So, what I do is random dot rand range question mark.

(Refer Slide Time: 07:11)



This will tell me everything of what it does as you can see choose a random item from the range start till stop the stop is mostly one before the numbers specified here.

So, if I say random rand range one comma hundred it will give me a random number from 1 to 100, in case I say random dot random once again this is another library

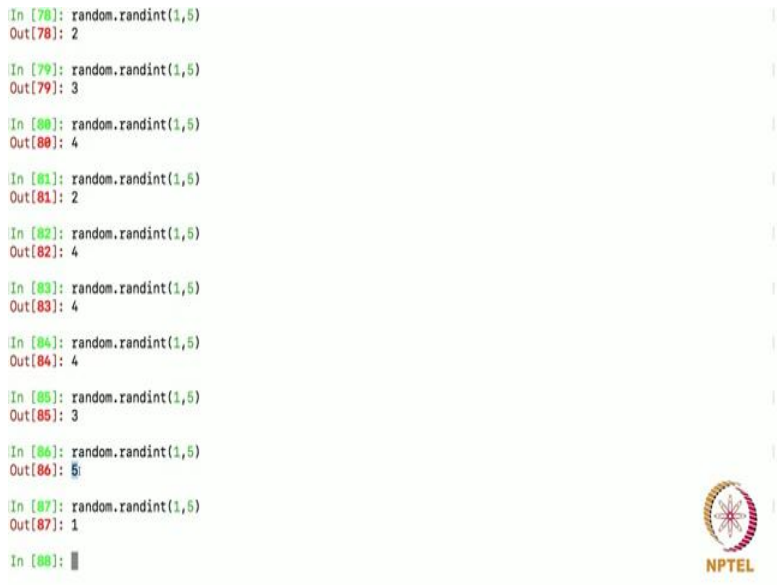
function which gives me a random number between 1 0 to 1 see it gave me a random number between 0 to one let me type that once again.

(Refer Slide Time: 07:49)

Another random number between 0 to 1, another random number between 0 to 1 so on and so forth. So, basically when I am doing this I am pressing my up arrow and pressing enter.

Up arrow just repeats the previous command all right if you want to see what all you can do with the random library function.

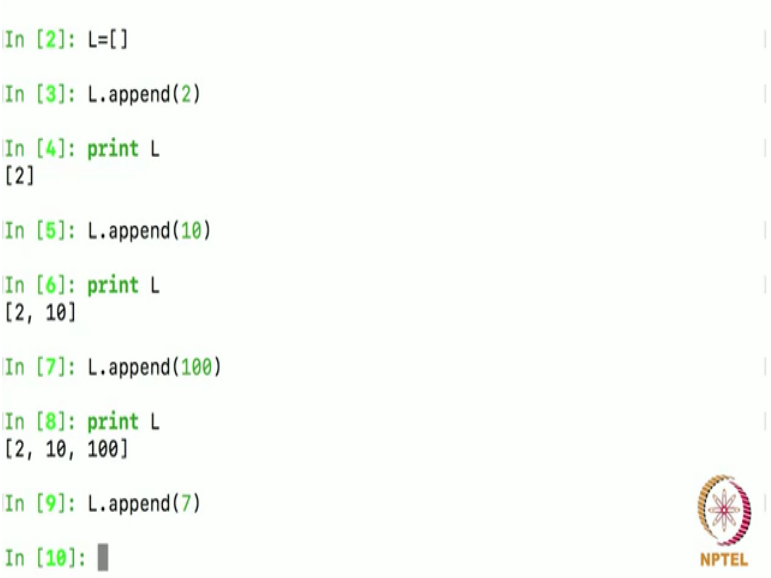
(Refer Slide Time: 08:02)



Just put a dot after random and press tab you will get all possibilities and you can use the arrow mark to go around and see what does what and there is a whole lot of functions written here.

Let us look at yet another function random rand int. So, what does it do put a question mark you will get to know that it returns a random integer in the range a comma b including both end points. So, let us try that random rand int 1 comma 5 gives you 2 so on and so forth. Somehow strangely 5 is not coming, yeah 5 came right now. So, it is between 1 to 5, but rand range is different as you saw before.

(Refer Slide Time: 08:58)



We will now look at what are lists in python, lists in python as I said are very similar to arrays and we will see how to use them. So, initially what I will do right now is I will declare lists called L, this is how you declare it L equals open bracket close bracket and then you say L append you put some number 2 print L, L will have that number 2, right, Now let me say L append 10, now to the existing list L, the number 10 gets appended if you say print L, we will get 2 comma 10 so on, I keep adding and then if I print you will see 2, 10, 100 and let us say I will append 7 print L, 7 gets appended so on and so forth.

(Refer Slide Time: 09:49)

```
In [10]: print L
[2, 10, 100, 7]

In [11]: L.append(77)

In [12]: L.append(14)

In [13]: print L
[2, 10, 100, 7, 77, 14]

In [14]: L.sort()

In [15]: print L
[2, 7, 10, 14, 77, 100]

In [16]: L.reverse()

In [17]: print L
[100, 77, 14, 10, 7, 2]

In [18]:
```

Best part is you can append whatever you want including character string; character string and whatever you want any data type now look at his let me add more numbers to this 77 L append 14 and then print L you have these numbers.

Sorting this is very easy simply say L dot sort and it will sort L as you see L is now sorted if you want to reverse L you simply say L dot reverse see what happens to this see it got reversed descending order.

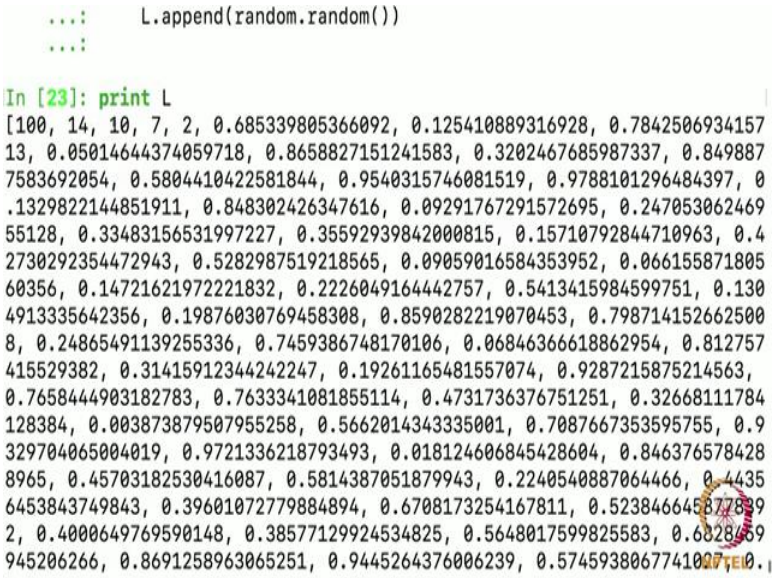
(Refer Slide Time: 10:37)

```
In [17]: print L
[100, 77, 14, 10, 7, 2]

In [18]: L.remove?
Docstring:
L.remove(value) -- remove first occurrence of value.
Raises ValueError if the value is not present.
Type:      builtin_function_or_method

In [19]: L.remove(77)

In [20]: print L
[100, 14, 10, 7, 2]

In [21]: import random

In [22]: for i in range(100):
    ...:     L.append(random.random())
    ...:

In [23]: print
```

What I can you do with this sort and reverse is what I told you what all can you do simply put a L dot and then press tab key and you will see all possibilities append count extend index insert pop remove. So, if you want to know what remove does type remove and put a question mark and it will tell you remove first occurrence of the value which means if I say L remove 77 it will remove 77 from the list as you can see 77 is here in this list let see if it removes or not I have removed 77 print L yes 77 is removed now.

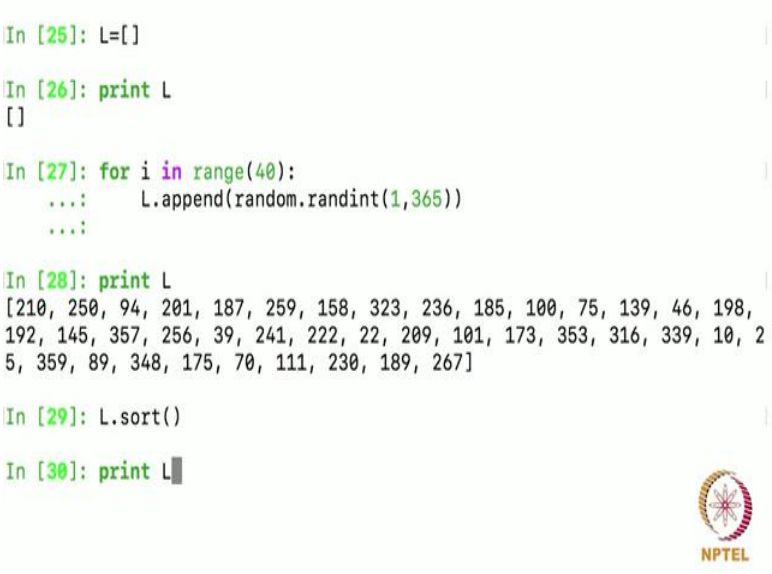
So, this is about list let me type something non trivial and show you what all you can do with the list we use random function rights. So, let us import random library function and then I say for I in range hundred L append random.

(Refer Slide Time: 11:39)



So, which means it appends 100 random numbers to L and L you see a lot of random numbers that apart from the initial 114 10 7 2 that we appended, we now have 100 random numbers appended.

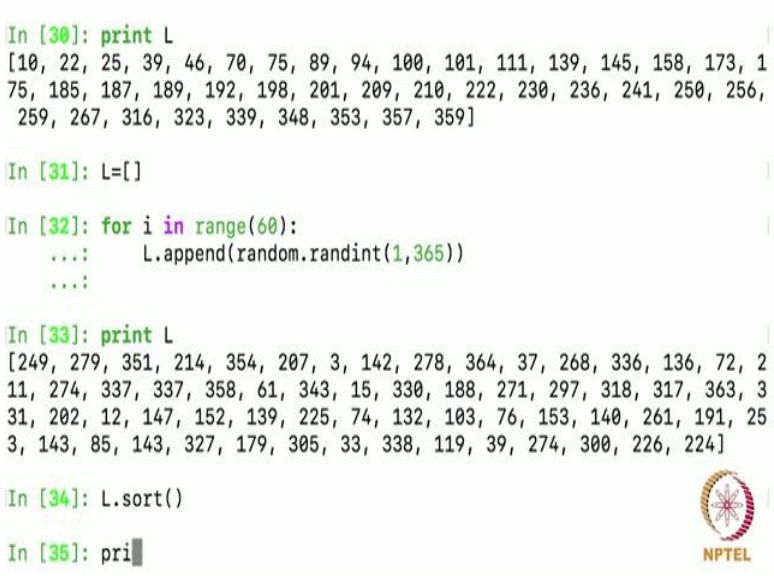
(Refer Slide Time: 11:58)



So let us do a small exercise let me clear the screen this is the command list for clearing the screen what does this do else I will reinitialize l. So, that whatever was present when L is lost, now I will do for I in range 40 L append random rand int 1 comma 365, what am I doing here? I am picking data of birth of a person let say if he was born on February first then you assign the number 32 to him if he was born on January fifth, you assign the number 5 to him. So, based on which day he was born you assign a number from 1 to 365 to that person pick forty people and assign this date of birth randomly to each person.

The point I am trying to make is I am I have now created a list L which comprises of 40 different birthdays dd, mm, day, month, only not the year, 40 different birthdays of people, let me see how it looks like print L has 40 different numbers from 1 to 365 let me sort this let me print this.

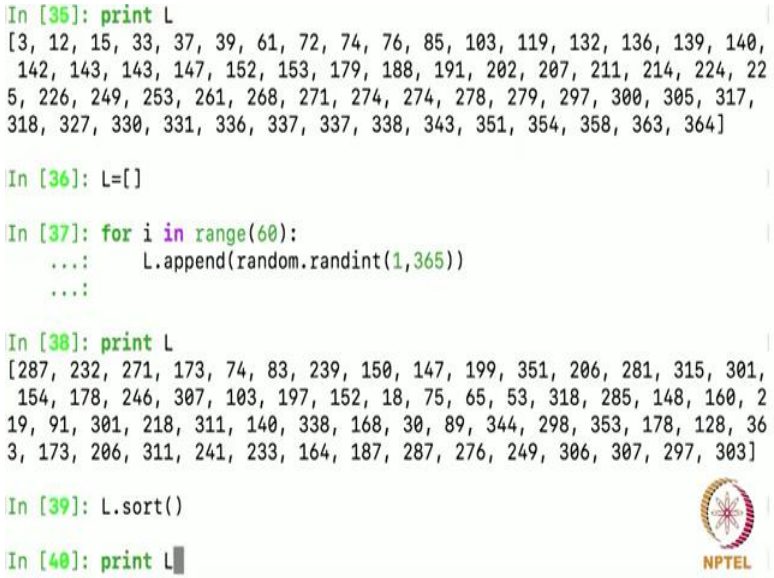
(Refer Slide Time: 13:16)



So, you see there are all these numbers and there are no repetitions I do not see any repetition here correct there are no repetitions perfect.

Now, what if I were to do the same thing like say I reinitialize for I in range 60 I say L append random rand int 1 comma 365 print l. So, I will again sort this.

(Refer Slide Time: 13:59)

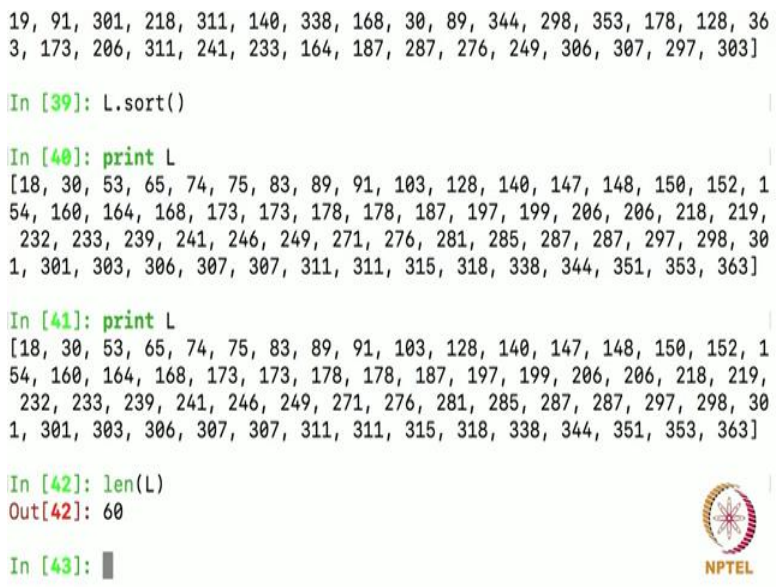


So, that I can check for repetitions if an I will see if there is anything repeated 3, 12, 15, 33, let us see if there is any repetition I do not see a repetition yeah I see a repetition look at this 143 is picked twice, 153 and so on is there any other repetition I do not see any

other repetition 274 is another repetition correct and then any other repetition that you can spot let me check 3; 3, 7 is another repetition there are 3 repetitions. So, fine there were 3 repetitions, but is it coincidental. So, what do I do?

Let me repeat this experiment L equals initialize once again for say for I in range 60 L append random rand int 1 comma 365 then you print L it is; obviously, not sorted. So, you sort it and then print L you will see for 40.

(Refer Slide Time: 15:00)



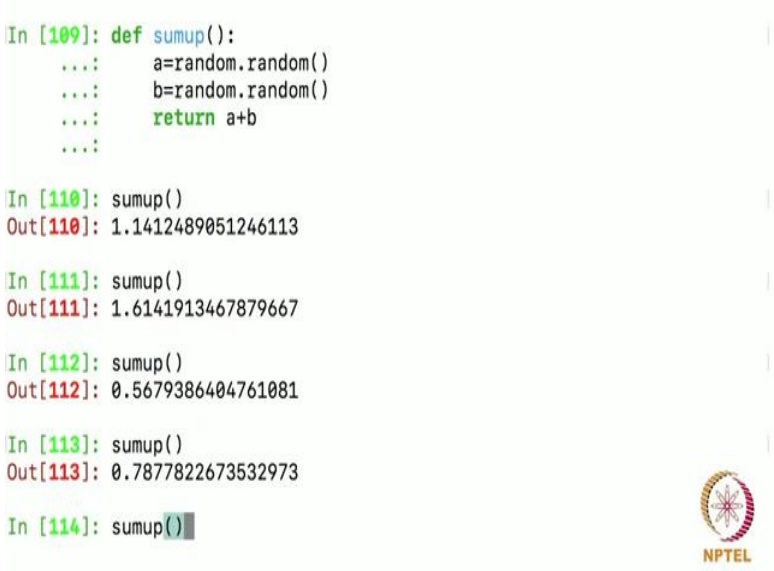
Picking there was no repetition for 60 there was I am repeating the experiment to see if it is true that you will always get repetitions there is a repetition once again once again 173, 173, 178, 178, second repetition right and then 206, 206, another repetition any other repetition let see 287, 287, third repetition, 301, 301, 4 repetitions in this case 301 and 301, excellent, 307, 307, 5 repetitions 6 repetitions, 311 and so on, correct.

So, what did we just observe we observed that by just taking list L with how many entries? Let us say the; I just printed L before let me again print it. So, there were len of L tells you the length of L number of elements in L. So, there are 60 elements because of a for loop it is 60 elements and if you pick randomly some 60 people you will have couple of them sharing their birthdays this is always true almost always true with a high probability when you pick 60 people from randomly and ask them for their birthdays you will have 2 people with the same birthday, what does that mean?

You go to a classroom randomly with some 60 students and ask them for their birthdays you will see 2 people with the same birthday in that class walk in to any classroom, this is almost always true, this is called the birthday paradox this is not a command by the way in python I am just writing it to illustrate it. So, a birthday paradox you can just google for it for more information a very interesting fact that if you pick some 40 to 60 people 40, it is rare, 60 definitely if you pick definitely you will observe 2 people with the same birthday we just took a code and observed it right now like.

That is with less as you would have seen the course homepage the prerequisites for this course is simply a first course in programming I suppose you all have studied programming in one form or the other one of the most powerful facility that you get in programming is your ability to write functions and python has a very easy way in which one can write functions we are going to illustrate that right now.

(Refer Slide Time: 17:41)



Now, demonstrate how we write functions in python it is a pretty straight forward process let me try writing a function on the shell itself first and then I will show how to write functions in a file. So, it is done the following way you say d e f define and then you will say sumup and then a colon open bracket close bracket and a colon and then you say let say a equals random which means assign a random number from 0 to one to a and then b equals same random assign a random number to b and then I return a plus b enter. So, what does it do when I invoke sumup it just shows me the answer please note that the

answer is more than one here that is because a and b are 2 random numbers between 0 to 1, there is some can beyond one let me invoke sumup once again what will happen picks 2 random numbers between 0 to 1 add them and then display it so on, perfect.

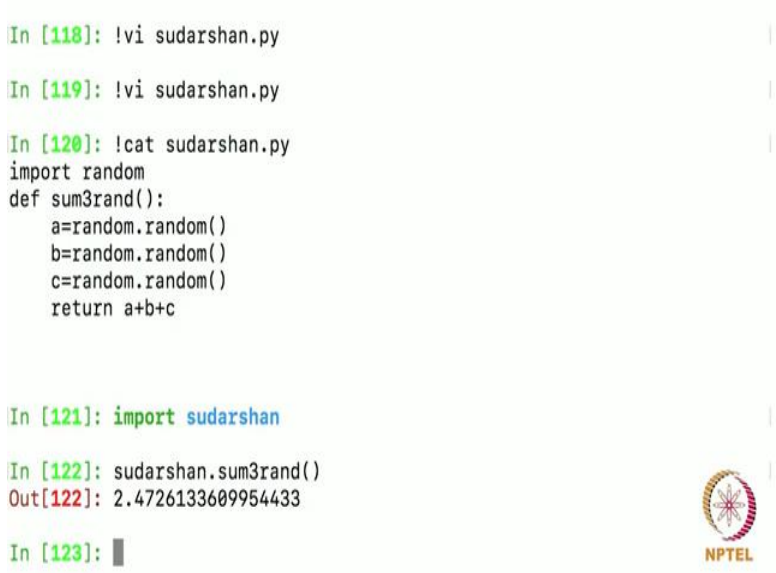
(Refer Slide Time: 18:58)

So, what you can do is whatever command whatever function definition you wrote right now you can actually open a file and then type it as well let me do that I am opening my favorite notepad I mean favorite editor which is vi editor. So, how do I invoke it I use exclamation vi space let say file name say sudarshan dot py.

(Refer Slide Time: 19:30)

This opens a new file you can use your own favorite text editor I use my I use vi editor note that exclamation stands for the here.

(Refer Slide Time: 19:41)



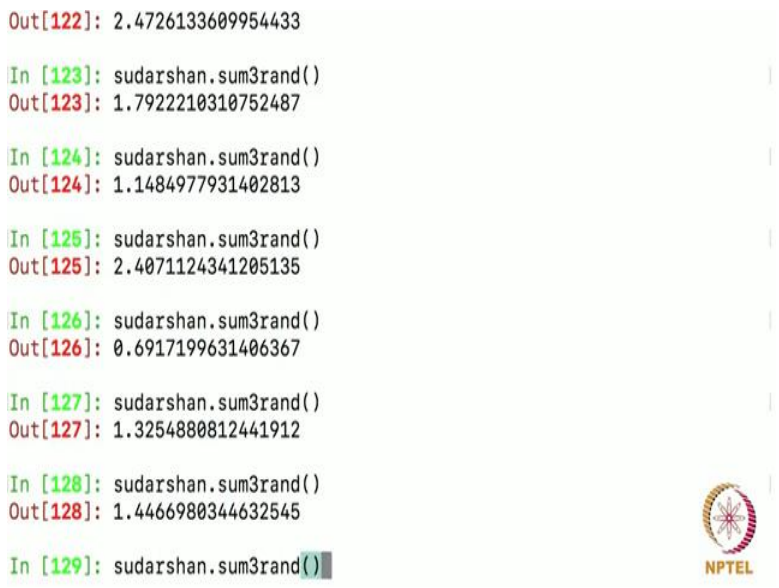
When you say exclamation it stands for outside command and then editor name and I type my file name you can even go outside this terminal and then open the text editor and then type it there also no problem.

So, let me define something here define sum 3 rand which is sum in 3 random numbers how do I do it I should first import random without that I cannot use the random function here in a file a equals random random b equals random random c equals random random and then I return a plus b plus c and then I come out its now saved I will display the contents of sudarshan dot py using the cat command of Linux it will just display the contents of the file you see that this much is there please note I have put import random here otherwise this will not work this was not the case with the defined thing that I wrote here I straight away used it that is because I had imported random already perfect.

Now, what do I do how do I invoke the sum 3 rand that is written inside a file by name sudarshan dot py that is pretty simple. So, what I do is I say I use the same import command when I say import sudarshan the sudarshan library whatever is contained in sudarshan gets loaded basically when I say import sudarshan import random happens and a function sum 3 rand gets defined with a following statements and now I can invoke it

see how I can invoke it I say simply say sudarshan dot sum 3 rand and that is it there you are random number between 0 to 1 added thrice.

(Refer Slide Time: 21:54)



Let me type it once again sudarshan dot sum 3 rand again gives you the same where sum 3 rand gives you some random number between 0 to 3 it is not really random because its picking a random number from 0 to 1 adding it thrice, this is actually not uniformly at random if you know probability you will know that, but that is not our concern here we are just observing that it is generating a random number thrice and adding it and then displaying it.

(Refer Slide Time: 22:18)

```
Out[129]: 1.768291238275506

In [130]: sudarshan.sum3rand()
Out[130]: 1.6856580370369079

In [131]: sudarshan.sum3rand()
Out[131]: 0.7293873918240366

In [132]: sudarshan.sum3rand()
Out[132]: 1.7983099756881937

In [133]: sudarshan.sum3rand()
Out[133]: 1.4387419167035427

In [134]: sudarshan.sum3rand()
Out[134]: 1.0149422968898556

In [135]: sudarshan.sum3rand()
Out[135]: 1.9272455647807818

In [136]: sudarshan.sum3rand(
```

The point is you see what is happening here sudarshan is the name of the file and then in that I am invoking sum 3 rand and I am displaying it.

(Refer Slide Time: 22:41)

```
 1 import random
 2 def sum3rand():
 3     a=random.random()
 4     b=random.random()
 5     c=random.random()
 6     return a+b+c
 7
 8 def sumkrand(k):
 9     ans=0
10     for i in range(k):
11         ans=ans+random.random()
12     return ans
13
14
15
~
~
~
~
~
:
```

Now, let me further edit sudarshan dot py, how do I do that vi sudarshan dot py and let me define another function here define sum k rand what does that mean add k random numbers now how do I do that I take k as a parameter here and then I say for i in range k you have had some programming experience that is the pre requisite for this course you know what I am doing? I will say answer is equals 0 and then I will add to answer plus

random, random that is it let me go inside I must return this, I am sorry, I came out without returning it. So, I say return ans.

So, what have I done here k is the parameter here as we can see and then I have initialized the answer to 0 and then I am running a for loop k number of times that is what this means for i in range k means assign value 0 one to k up to k minus one to I whatever is assigned to I, it does not matter to me here I; all I want is I want to execute this loop k times this particular thing k times. So, answer equals answer plus random dot random simply picks random number from 0 to one and adds into the existing answer and assigns the new value to answer and finally, I come out of the; for loop and the return answer. So, what does this do? So, let me see let me save this and then come out.

(Refer Slide Time: 24:09)

```
In [136]: sudarshan.sum3rand()
Out[136]: 1.1629215697772501

In [137]: !vi sudarshan.py

In [138]: !vi sudarshan.py

In [139]: import sudarshan

In [140]: !vi sudarshan.py

In [141]: reload(sudarshan)
Out[141]: <module 'sudarshan' from 'sudarshan.py'>

In [142]: sudarshan.sumkrand(10)
Out[142]: 4.575307276426816

In [143]: sudarshan.sumkrand(10)
Out[143]: 5.2256366698481544

In [144]: sudarshan.sumkrand(100
```

Now, please note import sudarshan will actually not work what you should do is whenever you have opened a file editor and you have added something new and then saved and come out you should not use import you should use reload sudarshan. So, sudarshan is reloaded in to the memory with the new updates a new function that I wrote just now and then you can open the new file what is that sum k rand and then I say ten. So, what should this output 10 random numbers sum if I say hundred instead of 10 you see hundred instead of 10 it will add hundred times random numbers from 0 to 1.

(Refer Slide Time: 24:57)

```
In [147]: sudarshan.sumkrand(100)
Out[147]: 52.737544589525804

In [148]: sudarshan.sumkrand(2)
Out[148]: 1.0681717586350272

In [149]: sudarshan.sumkrand(1)
Out[149]: 0.8563130656430317

In [150]: sudarshan.sumkrand(1)
Out[150]: 0.03881258471291482

In [151]: sudarshan.sumkrand(1)
Out[151]: 0.6739417080580347

In [152]: sudarshan.sumkrand(1)
Out[152]: 0.5814914963040022

In [153]: !vi sudarshan.py

In [154]:
```

And then output the answer yeah, perfect, if I just say 2 if I just say one it is as good as a random function correct. So, what did we just now learn you learnt the following?

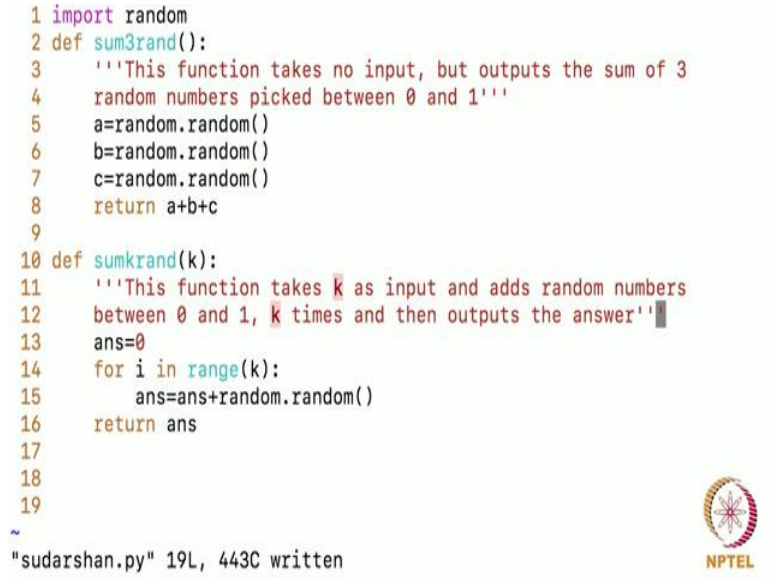
(Refer Slide Time: 25:22)

```
Out[154]: 2.572630972476484

In [155]: sudarshan.sumkrand(5)
Out[155]: 2.563732366891008

In [156]: !vi sudarshan.py

In [157]: !vi sudarshan.py

In [158]: reload(sudarshan)
Out[158]: <module 'sudarshan' from 'sudarshan.py'>

In [159]: sudarshan.sum3rand?
Signature: sudarshan.sum3rand()
Docstring:
This function takes no input, but outputs the sum of 3
random numbers picked between 0 and 1
File:      ~/nptel/sudarshan.py
Type:      function

In [160]:
```

You can create a file and then write down your functions there and then invoke it from outside how do you invoke just say sudarshan dot followed by the name of the file sum 2 sum 3 rand is what we declared without only a parameter it generated a random number by picking a random function thrice and we also saw we also wrote a function by name

by sum k rand with an input parameter let say 5 which took 5 random numbers added and then displayed.

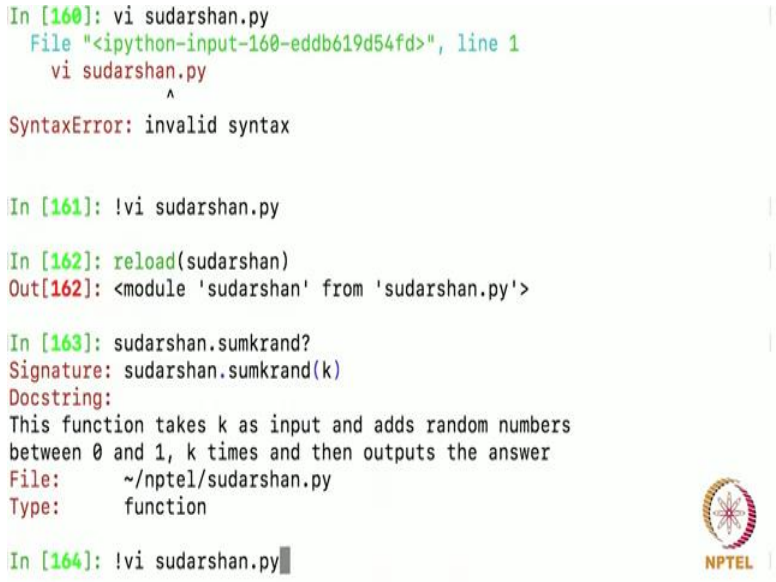
(Refer Slide Time: 25:56)



One important tip open the editor an important tip you can come here and then write your favorite help one liner you can say this function takes no input, but outputs the sum of 3 random numbers picked between 0 and one and then close it I will see what happens let me show you I am going out saved you just saw what I did I opened the editors sudarshan dot py and I have added this thing here what did I do 3 single quotes followed by some random text and then I close the 3 single quotes and I came out there is some change to the file. So, I should say reload sudarshan.

Now, when I when I want to execute some command in sudarshan I type sudarshan and then I put a dot and then tab is a very important key when you press a tab here it will show you all the functions that are available in sudarshan random is available because random is imported there right sum 3 rand sum k rand is available if you put sum 3 rand which by the way is a function that I just defined and then put a question mark guess what happens.

(Refer Slide Time: 27:20)



It shows you your help file that you just now wrote you see this; this function takes no input the output the sum of 3 random numbers picked between 0 and one who wrote this we wrote this just now correct.

Let me write as similar help file for the other command too. So, what do I do vi sudarshan dot py and then I am sorry I forgot to put an exclamation vi sudarshan dot py without exclamation the outside commands do not work vi is an outside command I come here what does this do let me just state that here this function takes k as input and adds random numbers between 0; between 0 random numbers between 0 and 1 k times and then outputs the answer towards the single code save you see what I did I just wrote a help function help one line for sum k rand function and then I go out and then I reload my sudarshan file now I say sudarshan sum k rand and then question mark it shows me the herewith whatever you have typed comes here.

Now, why is this even used this is used because when you write hundreds of functions. So, here we wrote only 2 functions what if I write hundred functions here in this file and I would not know what function is doing what.

(Refer Slide Time: 29:22)

So, it is important for us to write down what a function is doing in the form of help file that with the introduction to functions and you know how to define a function on the ipython terminal you also know how to define how to how to create a file edit the file and then type the functions there and then come out of it and then invoke it by saying. Firstly, import sudarshan and then second time when you edit something and then you want to reload it you may say reload sudarshan note reload is used when you edit the file and reload means update the contents of sudarshan, I have just now changed it and then what you do is you say sudarshan dot sum whatever is the name of the file name of the function and it executes that function; however big; however, complicated it executes the function and shows you the answer that is with functions.