# SNAKES AND LADDERS: NOT ON THE BOARD 03

Alright we just now saw favourite game of most of us in childhood snake and ladder being played by two persons, now let us play using a computer programme.

Hello all welcome to the programming screen cast of snake and ladder, in this programming screen cast i will take you down your memory lane where you had played one of the favourite games of i guess almost most of you in your childhood namely the snake and ladders, snake and ladder the board would generally consist of numbers from one to hundred arranged in a ten cross ten grid there would be snakes in some point and ladder in some points ladder would generally take you up the board that is from lower point to a higher point and snakes would generally takes you down the board from higher point to a lower point, this sort of symbolises theme of life that life is full of ups and downs so this would be favourite game for many people because if is somehow close to the real life scenario alright let us now see how we shall play this game in a very cool manner using our python code through our computers. Alright let's get started. Before getting started with the code i will show you the image of a snake and ladder board i had got an image from an internet this is here this is a snake and ladder board as i had said there are numbers from one to hundred in an ten cross ten fashion there are ladders somewhere as well as snakes somewhere i am sure you would have played this game in your childhood this is just journey down the memory lane alright, i would ask you guys to please pause this video at this moment make a note of snakes and ladders in this board various points from where to where is the transition of each ladder. For example this particular ladder starts at twenty one and takes you to eighty two and this particular snakes starts at sixty four and this takes you to thirty six like that you please make a note of this snakes and ladders by pausing the video, this would be of greater interest to us during the later part of our code, please pause the video at this point and make a note of the snakes and ladders alright i hope you would have paused and made a note of the snakes and ladders now let's get started with the coding part let me minimise this image let just started with the coding part when someone plays the game what are the two things you suppose to do, in the computer someone is going to play the game you are suppose to show him the board i call that as show board functionality the next is he is suppose to play the players are supposed to play and the end point of the snake and ladder we have to define, this snake and ladder this board has end point at hundred, hundred is the winning position ,when some player reaches hundred first he is considered as the winner, two players would play and the one who reaches hundred first is consider to have won the game till than the game will proceed in case the player wants to bought in the middle he can totally quit the game if the players continuously proceed hundred is considered as the winning position, this is how we are going to configure our game here so as i said first you are suppose to show the board to the players then you have to ask them to play this is the higher level abstraction details and this is considered to be good programming practice that has to, you have to abstract as much as possible like see a person who wants just an higher level overview he would read this he would understand the entire point is hundred and you are suppose to show the board and play the person is suppose to play this is what the game expects someone to do so this is the higher level overview so this is considered to be a good programming practice to give a higher level overview, your

code must be organised such that by reading these overview the person should be able to get the higher level idea of what your code will do and if they want to del deep into the code then you may allow them into del deep into it but for a person who just need high level overview he should not be made to go through the final details unnecessarily this is considered to be a good programming practice for this purpose we are using different functions and another use of functions is reusability if you want to repeatedly do something and if you want to reuse that the if you want to use the same functionality at various places then functions come to your rescue they greatly help in reusability so it is very good programming practice to use functions as much as possible and the higher level overview must be easily obtained for any one so the naming of functions also i recommend that you give such that it is easily understandable by someone it is also the syntactically it is corrective you can say f one f two f three function f, f one f two f three and so on but it doesn't give the overview of what is the function may contains some twenty thirty lines of code or may be more than that as well as an how the requirement is for the code may grow longer and longer that maybe there but to understand what that particular function say f one is doing you have to go through those hundreds of lines of codes and understand this is not good programming practice so to facilitate that a good practice is to name the function accordingly suppose the function is in calculating the matrix multiplication say just for example i am using it matrixes have got nothing to do with this game now i am just giving an example suppose the function is performing matrix multiplication you are suppose to name it as matrix multiply that is a good convention so that just by the name the person who needs a higher level overview can get it without getting deep into the code so this is the good practice please make a note of it and use functions as much as possible because it simplifies your work greatly alright so now let's gets started with it, we have to get deep into the functions, show board and play are the two functionalities and now i need to say what this show board has to do so let me close this image, now i don't think it is needed, let me close this because in show board i am suppose to open the image again, show board functionality this is the functionality, this is a image file so i have to import the image library from the package PIL input image library this is how i have imported the image library so let me now open the image img is the object that captures the image, image dot open here you are suppose to give the file path where the image file is present in my case it is present in the same directory as that of my code so i am just giving the file name in case you have stored your image in a different path please do specify the complete path so that the image shall be opened alright now once the image is opened it has to be shown so i will say image dot show this open show all these are pre defined functions in the image library and as you would see as i had said they had given the names such that you can understand what it does what it does, image dot show there will be a lot of transformations involved in showing from as you know computers deal every in terms of binaries zero and ones from that to a image that is understandable by humans to transform there are a lot of steps involved but all these are abstracted at what we know is, if you say show it will show the image so that is enough for us we need not del deep into the binary level information so in this way your functions must be named open show all these are self explanatory what this particular thing is going is clearly understandable just by this so htis is the show board functionality currently i am not going to code for play so let me command it but this will also be an important step so i have return it and command it and once show

board works will go with play, let us we had written the code for show board let us see if the board is being shown then we will go with the next functionality play i will save this file and i will run this see the image is shown, so this is the image that is that has been shown also the board will be shown so this functionality works fine no problem with this  let me close this now and ok the next thing is we have to go with this particular functionality play we will see what are the other functionalities required in play if it requires any other functionalities or any other libraries has to be imported as and when we del deep into we import and we will define new functions we will make the code more readable and if you just give this code probably to your maybe younger sibling maybe ten, twelve years old sibling who barely understands English that person just by seeing the names would be able to understand what are the checks that are being performed how the flow of programming is, everything would be very clear we will make it in such a way we will modularised this program this particular process is called modularisation like you split the you split the single units of tasks into chunks like show board this is one particular task i have made it once chunk and play is one chunk the play consist of some sub chunks we are making the functionalities into chunks so that anyone can get the higher level overview, even test it someone who just understand English with the names they would be able to understand what is the flow of the program, you may check that alright we have done with show board functionality let us go into play functionality