FLAMES 03

Alright guys, so in the previous video you had seen what is this game of flames. So it is basically you have two names and you have to cross out the common letters across the two names that is within the name if there is letter repeats you cannot cross it out, if it repeats that is if you have one letter at name one and other letter at name two you can cross out the two letters like that you keep crossing the common letters and at the end how many letters are left out you have a count of it and you go in steps of that many number of counts along the word flames from the left and if it exceeds the word length you again come back to the first letter in the anticlockwise direction and you proceed it, this is how you play flames right so we had run through an example we had taken two names ajay and priya and we had run through and we found that at the end of our game the letter that remains is F which denotes friends so ajay and priya are friends is what we got as a result. So something like that we have to play the game so for that will do the programming part now, before going into the actual programming part i would like to give you some introduction to some pre requisites that is needed to code this joy flames so let us go ahead with the pre requisites. So the string is nothing but the list of characters as you could see string is what the technical term we give for words, name is nothing but words right so name consists of group of characters so like that we have a group of characters, group of letters, letters is technically called as characters and a group of characters is what we call as string technically so when dealing with strings we need to know of some functionality of strings and which is available in the python package namely string there is a package name string so let me import that first import so let me say let me start so the first thing we have to do is import this package named string let me do that import string this is the name of the package, yes we have imported this package string, if you feel the name is lengthier you can use as and give us smaller name s g something like that, that's your wish from this string ok i am doing this, so let me say those string i have s is some string strings the notation is we initialising with double quotes so let us say the name of the person is raj, raj is the name of the person so if this is a name of the person i have a mix of capital letter that is the upper case character we call them as the capital letters and the small letters we call them as lower case characters we have a mix of upper case and lower case. If i want to convert it to all in upper case or in all in lower case i have a functionality with a corresponding name let us see that, so first let me see for lower case that is all small letters is what we call as lower case, lower case is nothing but small letters, i want to convert into completely small letters so i would do it as print i want to print it in that format i will print it i will print it see s dot lower is the functionality we have this functionality s dot lower, if i print see i am getting the name in all small letters same if i want it for all caps what i will do is print s dot as you would have guessed its upper cast the capital letters are technically called as upper case characters so we will use the functionality upper, upper ok this is the functionality so let me press enter see RAJ its printed in all caps the upper case letters so this is how you get the complete lower case or complete upper case of a mixed string or even if you give in all small letters if you still apply s dot lower there is no problem it will return the same string there is what this functionality does is it will check if the given input string has all small letters if yes it will return the same string otherwise it will convert all letters to smaller letters and return that converted string similar is that for upper function, it will check

if all the input characters that is the input string has some characters if all those characters are in upper case that is the capital letters or not if everything is in capital letters already it will return the same input otherwise it will convert the letters into the corresponding capital letters that is the upper case letters and return the converted string so this is how these two functionalities work. So upper and lower these functionalities for it to work properly to import this package string and the next thing is we were actually looking at individual character level right so we were checking this particular letter ajay A the first letter whether is it present in the next name priya it was present so we cancelled the next letter j we will compare, it was not present next letter A we will compare it is not present, so something like that we will be doing, right so we will compare letters by letters so when we have this has a whole string we cannot do that comparison so we need to convert it into list of characters list of individual letters that is what we need to do it, that will be required in our program so we will see how can this be done so let me say string list is nothing but list of the string s the variable is s so i am passing it ok so it was converted into a list so let me see what is present so let me say print string list ok so it contains the individual letters R A and J so it is the a single string is converted into list of three characters right so this list function precisely does that it will take the individual character make it an element and convert it convert a single string into list of individual characters that is what this functionality does. Another thing you can do with this strings is you can find and replace i hope you would have heard about this terminology this is most popular if you are a movie watcher as well if you watch in movies some movies have comedy scene based on that i hope you remembering three idiots is one of them it has been remade in into some other languages as well so in that they will find a particular word and replace it with some other word and they will make the hilarious situation out of it something like that we can do replace operation here so we will find a character and replace it with something else so let us say i don't want to print J i want to print some other letter in place of it so how should i do that, so i would say print i want to print the result S dot replace is the name of the functionality replace of as i said i don't want J so i will write J here J with let me say i want to print it with hash so i will say hash so if i say it is printed as R A hash in place of J hash is substituted so this is replace function precisely does that it will search for this particular string here and it will substitute this string in place of it, this is a source string this is a target, this is the source this is the target it will search for the source and it will replace it with the target its not necessary that you need to give single letter you can give sub strings as well so let me say let me modify it i will say AJ i can give a substring a part of the string is called what you call as substring in substring what is mandatory is the letters must be in same order if i say AJ and if i say if i press enter see R and the part AJ gets converted into this hash and here this is case sensitive that means if it is in a small letter this has to be in a small letter otherwise it won't agree if you say if i say RA and say enter see hash J is the output so this particular substring RA is searched and in place of that you are substituting hash it happens now i will check with smaller ra small ra i will check see you are not getting something like this you are getting raj completely because it is case sensitive that is whatever is the it is a smaller letter or a capital letter that it is upper case or lower case letter depends on that it will find and substitute so this replace functionality is case sensitive we call it as case sensitive by case sensitive we mean if it is a capital letter you must say capital letter if it is small letter you must say small letter it won't match ok it's the same letter

just that, that is in caps and this is in small it doesn't perform that kind of a match that is what we call as case sensitive and most of the passwords are case sensitive if you have been using it in your mail or with your net banking anywhere the passwords are case sensitive so in that case if you a capital A it is different from a small a this is what we call as case sensitive, alright this is how this replace functionality works and even you can give something so let me say i will i will give this capital RA and i can give star hash i can give something like that as well so even here you can give a substring in place of this substring you substitute this substring that can also be given it is telling us star hash and J or maybe i can do this way as well i can say just A see what happens let us see in case of A star hash has been substituted so this is nothing but you find this source whatever is the source in place of that substring you substitute this substring that is what this replace functionality does, i hope you are clear with this why would you use replace functionality in the code will see. alright the next thing that is needed is slicing of a list, i hope you have used it already in a some of your previous weeks but still let us have a recap because we will be using ti extensively here the slicing place a major part here so we will see a recap of it so let us see we had a list right sl this is a list you had R A J in case i want to extract only the specific part of the list, this is hte smaller list so let me take the larger list let me say L equal to i will take letter because we are going to deal with strings so will take letters only a b c d e this has to be a comma f ok so let us consider this list containing six elements this is my list L containing six elements a b c d e f so if i want to extract some portion of this list only and display that is what we call as slice in your taking the slice out of the entire list is what you call as slicing so how would you do that L within the indices brackets square brackets you should give the start index from the let me say index one up till index four let me say let me see what happens ok as you could see i had given index one to index four so what is the output, it starts from index one as you know the list indexing starts from zero so A is at index zero, B is at index one, C is at index two and so on so index one the letter found is B so B is printed index two C is printed index three D is printed index four E is not printed so this is how the slicing functionality works, the end index is ignored so this means when you reach the index four you stop the process you don't print that you don't included that is what this means so this portion of the list is sliced and taken so this is how slicing works and another thing is this is not mandatory to give both the indices the start index and the end index. That is it is optional so if i say list of i don't give a start index i just give a end index let us see what happens ok this is the output see you are getting A B C D just before the fourth index so if you don't give a start index the default value that is taken is zero, zero is taken as the default value and from zero till four minus one three the third index everything gets printed everything gets added to the output, aright what if i don't give the end index? I will say two i don't give an end index what happens from index two see A is at index zero, B is at index one, C is at index two from index two it prints the entire list so the default value of the end index if you don't give is taken as the length of the list so a recap or summary of what we have seen till now. Slicing is nothing but you cut the required portion of the list you just take the part of the complete list is what you call as the slicing you are taking a slicing of a list that is why you call it as slicing, give the square brackets and within that you need to give start index colon end index plus one please note that it is end index plus one because the last index won't be counted into the output it will ignore at the index it will stop at that index so you have to give end index plus one and it is

not mandatory you to give both start index and end index it is optional in case you don't give the start index the default value is taken as zero if you don't give the end index the default value is taken as the length of the list so what if i don't give both let me do that what if i do this? The entire list gets printed because the default values are taken so this works basically like L of zero to length of the list is six here so zero to six it will start at index zero print A, print B, print C, print D, print E, print F which is at index five and index six when the counter reaches six it will stop so it will stop here and it will print the entire list so this is how it will works if you give the value it will take that particular index the way you have to give is start index and end index and plus one please note this plus one fact end index plus one and if you don't give the index it will take the default values, the default value for start index is zero the default value for end index is length of the list, ok guys i guessed you have a good revision of the list slicing which you had studied some weeks back as well as you saw the new functionalities of upper, lower, replace in strings please do explore these functionalities also there are some more functionalities that are available please do explore them and get more familiarised with these features with this command also yes one more command that you will be using in this we will see of this is the list L and i want to get the index of an element i know the element i want to get the index that can be done by using the function index let us see that. Print L dot index is the function you need to pass the element here the element let me say is D, i want to know the index of the element D so if i print i am getting three, the index of the element is three. In case if there are multiple places this element D is present if your list is A B C D, A B C D, A B C D E F just assume if that is your list then the first occurrence of the value that is what is returned as the output here the index function return the first occurrence of this element in the given list, the index function returns the first occurrence of the given element in the given list right because how this functionality works so what all you have seen, we have seen the upper functionality which converts the strings into completely upper case letters, lower functionality which converts the strings into completely lower case letters replace functionality searches for a sub string and substitutes with another sub string then we have seen list slicing with start index and end index plus one then if you don't give the start index or end index what is the default values, how it works we have seen and we have seen the functionality of index which returns you the first occurrence of a given element in the list so this is what we have seen now this are the pre requisites that are needed to code flames so with this maybe you can start thinking and you can start trying the code flames so from this develop a logic and you check it with what we are telling in the next video that would be the best thing that's the best way you can develop programming logic and you can develop by yourself i would recommend that and please do explore what are the other features available here and by this is by exploring and practicing, you become a better programmer and so keep practising is what i would say ok guys, thanks for watching this. Will see how to implement flames in the next video.