LECTURE 31 : Regular Expression

Hello learners, in this lecture I will be discussing about regular expression which is also known as rejects. So what this rejects is, rejects is also known as regular expression is a powerful tool for matching the patterns in the data set. In the analysis of text data it is very important used for all kind of pre-processing and manipulation. It is a sequence of a character that define a given pattern. Rejects are used for finding the text, manipulating the text data or validation of the text data and many more. Rejects are supported in all the environments and text processing tools.

Let us understand this rejects with an example. So take an example you have a data where it consists of all the abstract of the publications. Now you want to know what all license under which these articles have been published. So this information you won't get when you will be downloading the data from either of the bibliographic database.

So this information will be in the text either in the abstract or maybe on the full text. So you need to extract this information from this text data. How you will do? So first we need to identify that what information we are extracting. So this is the information. So let first see how we do manually to identify this particular information in the text.

So we are very habitual of doing this. How we do? We generally use this control plus F to find any text in particular file. So we will use that only. So here I have taken up all those last line from the text and paste it on this slide. So I have to identify that what license they are using.

So in Google slide we can do by control F like license. So by searching simply license we can easily find out that these are the text where it matches the license. But it did not matches license 1.0, 2.

0, 3.0, 4.0. How we will do that? So one way is that we will just simply write the number but it will match only this license only or maybe this license if we write 3. But it won't match license 1 or 2 or 4. So what we can do here is so we need a some kind of a solution that matches license also and this number also.

So this is our text and 1.0 license 2.0 license 3.0 license 4.

0. So if you see here there is a pattern is there. What pattern is there? So this license word is common in all these four statements. So and another thing is that then this is common. So these are numbers. Some numbers are there.

So what kind of pattern is there that there is a word then numbers. So this is what we need to identify in the text. So we will write here license and space 1, 2, 3, 4 and if we go to these three dots we will see here we will just use the regular expression and just put this here. So

if you see here that we have written license then space and 1, 2, 3, 4. So what exactly it means? It means that search for license then space and then either 1, 2, 3 or 4.

There is a another way also of doing this. So we can say so there is a word and after that space is there and then from 0 to 9 any digit can come. So that is why it is searching here this license 1.

0, 2.0, 3.0, 4.0. So this is what rejects is it makes our task easier to find a pattern in a given text dataset. So say for example instead of this sc we write c because in different language style we write license we have also seen in R that there are two functions to know about the license. So if we do like this and then now if we do control F and we go to the our. Now if we are doing simple search then like say for example license it will search only this only but if we can do like this okay.

So we have identified so with this regular expression it identify all kind of license all kinds of writing license. So this is license is there and license 1.

0, 2.0, 3.0, 4.0. Now you can restrict also that you want to see in a text data that what all documents are there which are using the license either 2.0 or 4.0. So for that what you will do you can write that simply 2, 4. So it will retrieve you the text where license 2.

0 is there and license 4.0 is there. So this is what the power of rejects is and how it helps in analyzing the text data. So do not worry if you do not know that how this 2 and 4 is coming we will discuss again. So this is just an example to discuss about that what is exactly rejects do okay.

So rejects is a kind of a special code that identify and matches the patterns available in the text data okay. So if we see the word like this grey or maybe colour or maybe behaviour okay. So these all words are conveying the same meaning but they have been written in different standards okay. So that is why this rejects helps in matching up the similar words but written in different way okay. And this is not only important when you are doing the text analysis but it is also important when you are searching the data in particular bibliographic database okay.

So say for example you are doing a research on behaviour mapping of university students okay. But there when you are searching in the that particular interface you are using simply this word okay. So using some wild card you can include this particular word also in your search query. So this is where you can extract a good data set from that particular bibliographic database and then you can do the analysis okay. So let us now start understanding regular expression from very basic okay.

So take an example that these are the 4 words okay. So one is Mohit then one is Mukesh then one is Vijay then one is Prinkha. Now what is common in this old 4 words can you

identify any patterns in this particular 4 words? So the first pattern if you see here that there are 2 words that starts with M okay. What are those 2 words? One is this one, one is this one okay. Then we have another pattern in this particular word is that there are 2 words which have the 5 letters okay.

What all those 2 words? One is this Mohit and another word is Vijay okay. So there are 2 words are there that have 5 letters. Any other pattern you can see in these words? So there is a another pattern is that there are 2 words which are having even number of letters. What all those 2 words are? One is Mukesh and another is Prinkha because here we have 6 letters and here we have 8 letters. So there are 2 words that have even number of letters okay.

Any other pattern you can see in this particular 4 words? So another pattern in this particular 4 words are that all the 4 words starting with capital letter okay. So what exactly it is that all the 4 words are starting with a capital letter. Any other pattern you can see in this particular words? There is one more pattern is there. What exactly that pattern? So that pattern is that all the 4 words are also ending with a letter but that is in lower case okay. So in starting it was a upper case okay.

So this was a upper case. This was pattern. Now all the 4 words are ending with a lower case letter okay. So these are some of the patterns in there. So if we have this particular kind of a data set and if we have to say for example we have to extract the words which are starting with M okay or maybe the words which are having the 5 letters. How we will do that? So we will see that in R okay.

So that then you will have more clarity on that how exactly this rejects is working. So we already know that how we can create this object. These are our 4 values are there. So 4 words. So I am creating this reg data as our object now.

So now our task is that we have to match the words which starts with M. So these are the these 2 words. So what we have to do? So we have to instruct this R that there are some words are there which start with M okay or we start with M which is in upper case. Then there are some letters are there okay. So this is the only thing which we have to express that there is some words are there which start with M and we are not specifying that those words are either ending with a lower case or upper case or maybe they have the digit in that.

So this is the only thing. So how we will do that? So for that the first thing is that we have to use this B. What exactly it is? It is a boundary. It says that boundary of a word. So for word Mohit there are 2 boundaries are there. One is this boundary and another is this boundary okay.

So we are saying that this is the boundary and this is the boundary. So first thing is done

our boundary and our boundary is starting with M. So this is our boundary here and now we are saying that it should match with M. So our first thing is done. Now next thing is that we want to say that after M there are some letters can come okay.

So for that we will use W. After M any number of letter size can come. So we are just putting plus. So what this plus is? So this plus is known as quantifier okay. So we have discussed now until now one is this boundary and then this plus sign okay and this plus is quantifier and this B is boundary of a word okay. So in rejects we have some thing called as anchors okay.

So this B is one anchor which define the word boundary okay. So this is the particular boundary of a word we have already seen. This is a caret symbol which represent the starting of a string and then we have a dollar okay which represent the end of a string okay. So where this caret symbol is there? This generally is on the keyboard number 6. So if we see here so this is here caret symbol is there.

So if you press the shift button and use so here is this caret symbol for starting of a string and then for dollar we have here on this 4 key okay. So here again we have to press the shift key and we can use this symbol okay. Then we have some quantifiers also okay. So what all these quantifiers are? So one quantifier is this what this symbol star signify that? That star is 0 or more occurrence okay.

Then we have plus, plus means 1 or more okay. Then we have question mark, question mark means 0 or 1. Then we have this kind of quantifier where it represent exactly 5 okay. If we have to match say for example we have to match a word with exactly 5 letters okay. So we will use like this way or if we have to match the words which are having at least 5 number of letters then we have to use in this way okay. Or if we have to use the range of letters so say for example if you want to extract the words which are having a length of from 5 letters to 7 letter okay.

So this we have to use and this is how we can extract the words based on some criteria okay. So we will understand this by example in R okay. So this is our object it is already created. Now if we go here so now we will create our pattern okay. So what pattern is? So this exactly showing that the boundary of a word okay.

This is a boundary then after this boundary we are giving the letter M okay. So that is M is there okay. After that we are saying that after that any number of letters can come that is why we are using this W plus and then I am saying that the boundary should be closed okay.

So this is my pattern. Now if I run this. So in R we have a function called graph which matches the word in a particular data set. So before using any of the function what first thing we do? We understand how we have to use that function and how we can do that using the question mark sign. So we will use this question mark and then we put the graph.

So if I do like this. So the help is there. We have to do so we have to give the pattern. So what pattern is then we have to give that particular data set okay that X is a data set and we have examples also okay. So this is an example. So this is our object where we have these four words and this is our pattern okay. Now I will call this function where my pattern is this and where my object is this if I do like this.

So it gave me the output 1, 4. What this 1, 4 signify that? So if you remember we have discussed about the indexing of a element okay. So where the first value is located, where the second value is located, where third value is located. So this particular function is giving us the output that whatever the pattern you are matching okay that pattern is available on this data set but on this location first and fourth. So the only values located at first and fourth are matching your given pattern okay. So if we see that so what exactly value at our first location this and what exactly at the fourth location? So this is what we got from here.

If we see this reg data. So this is our first location and this is our fourth location okay and if we see the output of first location it is Mohit and the fourth location okay. So it is matching our pattern. So what our pattern was that it start with m and after that number of letters can come and then the boundary should be closed okay. So our thing is done. We have identified that there are two words are there in this particular object we start with m.

So there is another argument available in this particular function is that we can simply call this value equal to true okay. If we write this value equal to true now it will not give me the location that at first and fourth location the value is there what are the pattern you are matching. So now it will give the words which are matching with our pattern okay. So if I do like this so now I got my matching words okay understand.

So now move to another pattern in our data set okay. So if we see our words we have these words and among these words these are the two words which are starting with m but among these two words which we have only one word which are having the five letters okay. Mohit and Mukesh they are starting with m but they are having the different number of letters. Mohit is having the five number of letters Mukesh is having six number of letters. So how we can do that? So now our task is only retrieve the word which are having the five letters. So what we will do? We will simply so instead of plus sign what we will do? We will specify that what exactly the length we are looking for.

So if we do like this okay. So what this four is saying that there is only one letter is already m is there after m there will be four letters okay. If you do here five and now if you do like this it will come Mukesh because m we have already specified that m is a letters and after m there are four letters are there for Mohit and for Mukesh after m there are five letters okay. Now we have also seen that there are some words are there which are having exactly five length okay. So how we will do that? We will be using the pattern that there is a boundary then after boundary whenever the boundary starts from that boundary there are only five

letters okay. So for words which are having the five letter what all those words are one is Mohit and another is Vijay.

How we will define the pattern? So we will say that this is our boundary after boundary we want only five letters okay. There is no any restriction whether it is an upper case or lower case or it start with m. So what we will do? We will simply specify this is a word and which having of a length of five okay and then our boundary will be closed okay. So this will be our pattern so we will see in our how we will identify the words which are having a length of five letters okay.

So this is our pattern so we will make a new pattern. Now if I call this my output is Vijay and Mohit. So what exactly it is showing here is that there are two words are there which are having a length of five letters okay. Now we have also seen that there are two words are also there which are having six length and another word is of eight length. So if I create a new pattern and instead of five I just write six and now I create my new pattern and if I run this so Mukesh okay. So for five letters we have Mohit and Vijay but for six letter we have Mukesh okay.

Similar way we can extract the words for eight letter also. So we will define our new pattern and then if we run this okay. So it is working fine okay. Let us now discuss a kind of a scenario where we want to divide these four words into two parts. So the first part is we want to have the words which are having the length of five letters and another set of words where we want to have the words which are having more than five letters okay. So how we will do that we have already seen about the words which are having the five length of letters.

So we will see again once more time okay. So Mohit and Vijay and now for more than five what we will do? We will do okay. So among the four words there are two words are there Mohit and Vijay which are having the length of five letters and then we have another set of words which is Prengka and Mukesh which are having the more than five letters. So Mukesh is having six letter and Prengka is having eight letters okay. So until now we have taken up the very simple words okay. So now take some a little complex words okay and these complex words are basically the email id.

So if you have a text data set and you want to extract the email id okay. So if you see here there are five words are there but among these only four are valid email id okay. So this is one, this is one, this is one, this is one okay. So what will be our pattern for identifying an email id in the text data set? So what exactly an email id okay. So email id looks like that there are some words or number or some characters are there then we have a this at the rate symbol and then we have some domain name okay.

Maybe gmail.com okay or another radif mail or anything okay or institution domain okay. So our basic pattern is what exactly that some letters are before at the rate and some letters

are after at the rate okay. So if we say in simple words our email id will be like this but if we use this this kind of a pattern what it will happen it will also retrieve this word also but this is not a valid email id. So what we have to do we have to go like this that there are some letters are there then at the rate then there are some letters are there and then we may go with like say for example dot-com okay. But if we use this kind of a pattern for email id what it will happen it will exclude this email id okay.

It will take care only of this gmail this three emails it will not take care of this fourth email id for this email id we have to design another ejects for taking up this email id and also this email id we have to think of a different pattern what pattern it can be it can be like this at the rate then dot then this then dot okay and this can be optional okay. If we create our pattern in such a way that there should be some letters before at the rate so here letters will come after this there will be something there will be at least one occurrence then here also at least one occurrence so these two occurrence is mandatory and this is optional okay. So our rejects for this kind of a data set can be done okay. So this is what exactly you have to notice whenever you are creating your rejects to identify particular pattern in a data set okay.

Let us now see in our all these examples okay. So this is our object so I am creating it. Now my pattern is my initial pattern what exactly it is it is that identify at the rate okay. I am not taking care of anything else I am just identifying if there is any at the rate is there that means that is a email id okay. I am considering this fact only so what I am doing here so this is a boundary then some letters are there then at the rate and then some letters are there.

So if I do like this so this is my pattern. So now if we see here it has given us five possible email phrases based on our pattern which includes an email id having digit in it and another is just having at the rate between the words which is not a valid email id. So the first one m bhaira28attherategmail.com is matched with another email id because w plus is not only representing letters but also numbers basically it is matching alphanumeric characters. So now we will see that how we will extract only the email id which are having the dot com in the end.

So let us take this example. So now my pattern is so this is my boundary then letters are there okay at the rate then some letters and after that letter there is dot com is there okay. So this is my pattern and now if I use this pattern and see okay. So now these two email so now here these two are excluded okay among which one is a valid email id but another is not a email id okay. So this is how our rejects identified only those email id which is having dot com in the end and at the rate is in between okay these two criteria we have taken up okay. So if you see this pattern again that this is a boundary of a word okay then we have some letters after that those letters we have some at the rate then we have some again letters and then we have dot com okay.

Now if you see here that among these three email id there are two email ids which is having

some kind of a more pattern okay. But that pattern is that pattern is that it start with info okay it has info in starting okay. So we are adding one more condition in our pattern so what exactly that is so our pattern is info so the boundary the boundary start with info then we have some more words then we have at the rate and then some more words or now we will add some okay. So now my pattern is this and if I run this okay so in earlier email extraction we have three email id but now we have a another restriction that we want only those email id which start with info okay. So let us now discuss if we have a text where we have some digit also some numbers also how we will do that okay.

So for that let us assume that we have this object where we have this Mohit Mohit and this 1234 this Mohit 1234 okay. So here a one practice question for everyone is that what exactly this 1234 will be in this particular object reg data set okay. So you have to answer this in the discussion forum and please let us know what exactly this 1234 is here okay. So if we have to match a any digit or a presence of any numbers in a in the data set how we will do that so we will be using this d what d is so I am focused only on extracting the words which are having some digits okay. I am not bothered about of any letters or symbol I am just focused on only on the these particular digits okay.

So for that we have used d okay. So now we will discuss about the groups and the range in the rejects okay. So if you have remember when we were searching in the this Google slide about license so we have written li okay but it is also matching the letters in uppercase case also because in that case the license the first letter of that particular word license its its uppercase but we are searching with l small okay. So sometimes it is very useful for us to search only on the uppercase or maybe on the lower case or maybe mix of uppercase and lower case okay or maybe we want to see that the first letter should be uppercase or maybe end of the that letter is uppercase or lowercase or maybe it contains any of the numbers okay. So for that we will discuss about what all the groups and range we have. So let us assume that we have this particular object where we have different kind of words are there.

So we have taken up the all uppercase then one uppercase letter then we have all lowercase then we have one letter lowercase one letter uppercase three letter lowercase two letter lowercase then we have digit then we have a mix like we call it as alphanumeric then this is again alphanumeric but in both have some different pattern okay. So if you see here that digits the numbers are in the end but here the numbers are in the starting okay. Then we have another word called gurg if you see here the gurg though it has four letter word both gurg but this particular gurg has uppercase G in the starting and this gurg word has both uppercase G in the starting also and in the end also okay. And then we have a another word where we have these one uppercase then two lowercase and then we have digit also okay.

So I will create this object. So now our focus is only thing that we want to extract only those words which are having the lowercase okay. We are not focused on any of the numbers or any of the uppercase we are just simply focused on lowercase we want only lowercase words okay. So for that we have this called a to z okay. This particular a to z is representing

that that all the words whatever it will be retrieving those words should have the alphabets in the lowercase only.

So it cannot have the letters in uppercase okay. So if we do like this so my pattern is this and now if I call this function. So what exactly it is extracting? So if you see here it has this word also. So all the words where there is a presence of lowercase it has met only those words okay. So if we see here it has retrieved nine words from this particular object and all these nine have somehow occurrence of a lowercase alphabets. So if you see this it has ar then it is full lowercase then it is one then this is three lowercase then it is two lowercase then again here lowercase but digit also and here also lowercase then lowercase and this is also lowercase okay.

Now we will explicitly mention that we want only those words which are starting with lowercase okay. So here if you see we have a words which are starting with the uppercase alphabets or maybe some digits okay. So now we will explicitly mention that starting with lowercase okay. After that I do not care okay. I am just caring about the words which are starting with lowercase.

So this is my new pattern and now if I do this so if you see here okay. So in our previous pattern in this pattern if you see it has retrieved this nine words okay. But now it has retrieved five words only those five words which are having the lowercase alphabet in starting okay. So there these words are starting with lowercase alphabet okay. So in similar way we can have a another symbol is this where we want that ending should be with the lowercase okay. I do not care about the words which are starting with any of the things but it should be end with the lowercase.

So how it will be so this is our pattern so our lowercase a to z any of the alphabet can come but it should be in the end okay. So now if I create my pattern like this so this is my pattern. Now if I see for this see there is a difference between this output okay. So earlier here our restriction was on the starting of the word okay.

But now we have a restriction in the end of a word. So the end of a word should have a letter in lowercase. So this is similar this is similar this is similar only thing is there earlier we have this moheet one two three four now we have one two three four moheet okay. Because it is matching that in the end this t is from the lowercase alphabets okay. Now if you want to have the both the restriction that it should start with the lowercase alphabet and it should end also with the lowercase alphabet. What we will do we will call these both the symbols in this particular pattern okay. So what we will do so this is our starting so what we are saying starting should be from lowercase and then ending should be also from lowercase.

So if we create this pattern and now if I run this okay. So now it gave me the only word m which have only one alphabet which is of course like one alphabet that is in the lowercase

that means it is matching that it is starting also and it is ending also in lowercase. But I was looking for the output like this I want this or maybe this. So somewhere something is missing okay. So until now if we are doing like this our focus was just in the end of that letter okay.

So our focus is this only if you see for this end and for starting our focus is this okay. So what exactly it is representing it is representing that retrieve the word which is in the lowercase and it have only one alphabets okay. If we have to extract these object what we have to do we have to use plus symbol okay. So if I do like this okay. So now it is matching if you remember this plus is for one or more so what exactly saying that if we have to see the data okay first see the our data okay.

So this was our data here this word is in the lowercase full lowercase this word then this word then this word. We have only four words which are having the lowercase alphabet okay. So this particular word is not a in lowercase only Mohit text M this MOH and MO okay. So this is what exactly it has retrieved. So Mohit text is one word then this M then this MOH and then this MO okay. That is what exactly this plus sign signify that that starting from a lowercase alphabet and then it can have either one alphabet or more alphabets okay.

So we have example here where there is only one alphabet then we have three alphabets then we have two alphabets and then we have these seven alphabets okay. Now if we want to have only two lowercase letter okay. So this is where we have discussed about plus one or more but now if we want to have only two lowercase letter so we have to call like this so our pattern is this and now if I okay. So this is the only word where lowercase alphabets are there but also in the length of two okay. Then if we have to see for three we can see three like this or if we have to see like more three or three more than three so we can put this like in this way and if we run this okay.

So there are three and then these are seven okay. So what exactly this representing that extract the words which are having the three lowercase alphabet or more lowercase alphabets okay. So now similar to lowercase we can do the same kind of pattern matching for uppercase also. So for that what we have to do for lowercase what we are writing a small and z in lowercase for uppercase what we have to do we have to call it like this okay. So what exactly it representing this particular pattern so can you guess this if you can guess please post the video here and post your answer in the discussion forum okay. It representing that match the words which are starting with capital alphabets and it should also end with capital alphabets but those capital alphabets can be one or more than one okay.

So if we do like this so this is our pattern. Now if I go like this see okay. So our first letter is also in uppercase then our last alphabets is also in uppercase and this is only one example where the one alphabet is there which is like in uppercase okay. Until now we have taken up the lowercase example then uppercase but we can combine this also. So there may be the

case where we have like words which are starting with uppercase alphabet or maybe ending with lowercase alphabet or maybe they have in between uppercase or lowercase alphabets okay. And this generally we see when we do like there is a kind of a writing style but we call it as a camel writing kind okay. So if you see this is also so this particular word is also a mix of uppercase and lowercase just for enhancing the reading because if I write like this it won't be that much reading friendly okay but as soon as I write this in capital G it can be easily readable okay.

So how to make a pattern of uppercase and lowercase? So before creating any rejects for any of your pattern in that particular data set first see that particular text data set okay. What exactly it is what kind of patterns and what kind of occurrence are there okay. And accordingly you can step by step you can do the rejects for that particular pattern okay. In our particular example we have a word like this G A R then we have G then we have this word called gurg and then we have this okay.

So if we see these three words what are the exactly the pattern here okay. So pattern is that so in starting there is a uppercase letter okay. For here also uppercase letter is there and here also uppercase alphabet is there okay. Then we have two lowercase alphabets okay. So for here what we will do we will do A to Z this is Z is small then we have A to Z then we have A to Z okay. And then we have like occurrence exactly two here so we can do like this two and for here we have three so because our data set is limited to this way so we can use plus also one or more and then here also plus or maybe like if we have a big data set and we have a different number of words okay.

So there is a chance is that if we only want to focus on this then we have to use this way but we can use plus also and then here we have A to Z in uppercase then we have this so this is sold if we do like this this our pattern is done and here we have then numbers okay. So similar to A to Z but we have four alphabets for numbers we have 0 to 9 okay. So our pattern is done in this particular example. Now we will see in our whatever we have made it whether it is working or not I have taken up that in starting A to Z okay the capital that uppercase alphabet is there then one or more occurrence of lowercase alphabet and then in the end it should be the uppercase alphabet okay or instead of this what we can use we can use this also it is also giving same thing okay. So once you will be like practicing more about rejects so there will be more way of doing the same thing okay but whenever you want to match exactly the same kind of a pattern so you can like use this way also but for our case because our example is small because we have less number of words that if we take this plus also it is giving us the right answer okay okay.

So now like we have for these alphabets form like uppercase or lowercase similar way we have rejects for like digits so from 0 to 9 okay. So let us see quickly about those examples also say for example now our focus is only extracting the words which are starting with digits okay. So this caret symbol we are using and then 0 to 9 okay. So if I create my pattern and if I call it okay so these are the two words are there 1 2 3 4 and this 1 2 3 4 muhi if you

see our data set so there are four words are there which have digits one is this one is this then one is 1 2 3 4 muhi and another is this okay but for this particular kind of a pattern we are focused on only on the words which are starting with a digit so one is this one and another is this one okay. Now in similar fashion we can do extract the words which have the digit in the end so we will put the dollar in the end so this is our pattern and if I call it.

So now we have the three words which have ending with a digit so here it was starting with a digit but here we have a words which have digit in the end okay. Now if we want to have a starting and ending with a digit how we will do that so we will put the caret symbol then our condition that 0 to 9 and then we will have the dollar symbol so between this we are only focused okay our focus is that we want that it should start with a digit and it should be end with a digit in between anything can come okay our focus is this only. So if I do like this and if I call this so what exactly this character 0 is showing this character 0 is showing that there is no such word is available in this particular object where it should start with this digit and end with a digit okay. Remember here we have not given that plus symbol or any other kind of thing okay this particular pattern is representing the word which have only one digit okay but we have one word that is 1 2 3 4 which have 4 digit so if we write here plus and then if we do like this so our 1 2 3 4 that word is retrieved okay. So this you have to remember whenever you are creating a rejects for your pattern because if you do not write this plus symbol and you are interested in the word like 1 2 3 4 okay so then it will not extract that particular word okay.

So but if your focus is only on the words which are having the digits maybe 2 3 or 4 or maybe more than that so you can use the plus symbol or maybe that you can specify the exact length of a number of digit you want okay. So now say for example if you want to have this particular kind of a situation where you want to have the words which is upper case also lower case also and then a digit also okay. So how you will do that so you will make a combination of upper case lower case and the digit so we have already seen upper case a to z so in this particular pattern what we are doing is that it should start with the upper case alphabet then it can have the lower case alphabet and then we have used plus symbol okay. What plus means that there can be one or more occurrence of that particular lower case alphabet and then it should have the digits it should have the numbers and for numbers also there can be one or more occurrence okay. So if I do like this so our this thing is done so there is a one word is there we start with this upper case alphabet then it has two lower case alphabet and then it have a digit okay.

So as I already discussed that the same rejects we can write in different way also let us see the example of this here. So what exactly I am doing here we want the word this one so here I am saying that a to z uppercase alphabet then one or more lower case and then exactly I am mentioning that we want only those word where we have the occurrence of two digit only okay. So if we do like this and now if I call our this graph function it is met okay. If I do here 3 now if I call this it is showing character 0 that means there is no such word is available in this particular data set which end with exactly 3 digits okay or like if you want

to go with this similar kind of a pattern only okay you do not want to use this plus and all okay.

So the simplest way is use this way okay. So what exactly it is showing we are going by alphabet to alphabet okay. So if you see here this is one character so we are matching it with this so this is this task of this is done then we are saying there are two lower case okay o and h is in the lower case and then we are saying that two digits are there okay 0 to 9 0 to 9 okay. So what is that another is 2 okay we can do like this and now if I like this so this can be done okay. So you can do by creating a pattern like this way also you can exactly mention that you want capital M then o then h small then 1 and 2 okay. So this is how you can create some rejects to match the words in that particular data set so thank you.