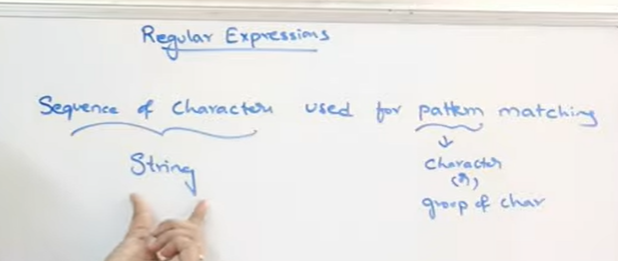
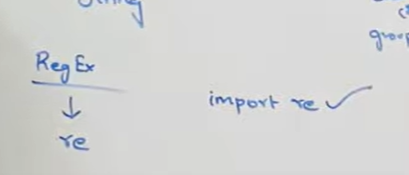
Regular Expressions

Sequence of characters used for pattern matching

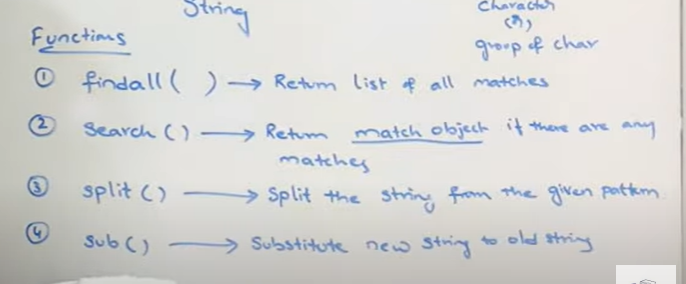
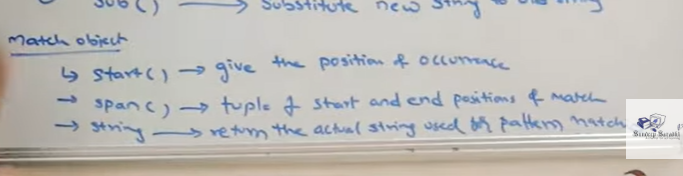


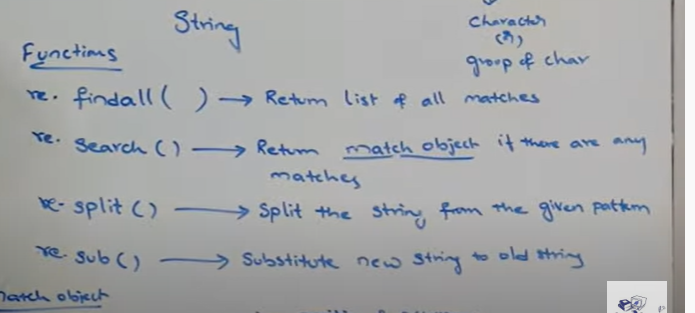
Regex module

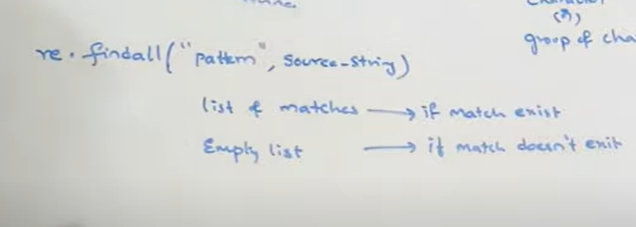
Re package

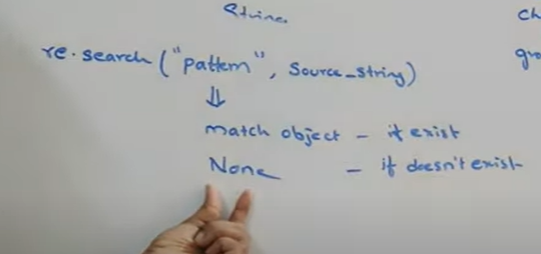


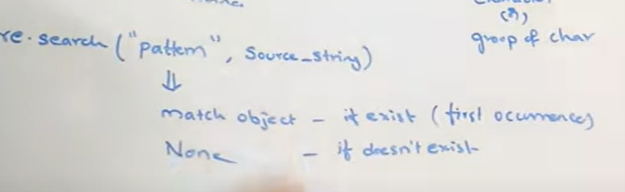
Basic functions or methods available

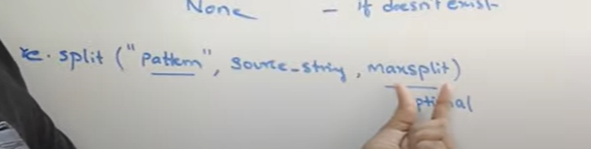
1. Findall: it will return list of all matches
2. Search – it will return match object
3. 
4. 

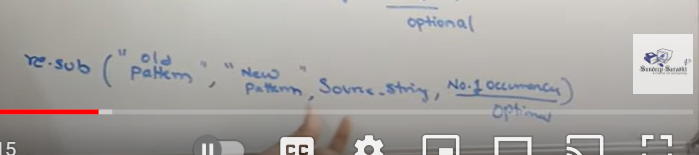




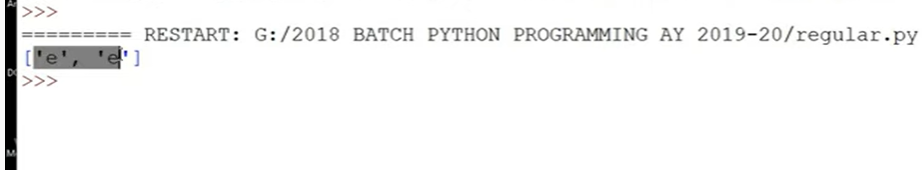


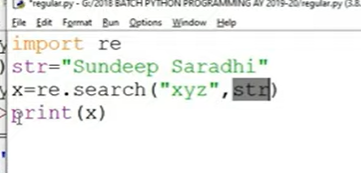




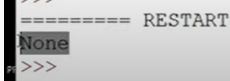






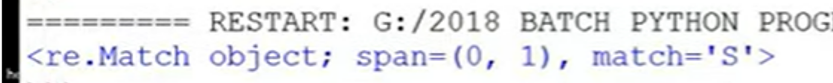


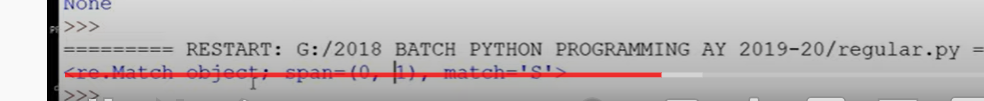
If pattern not there

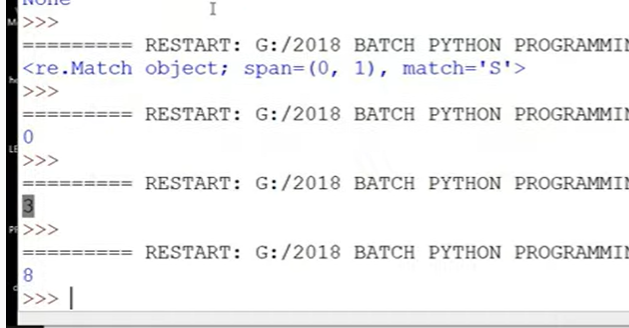


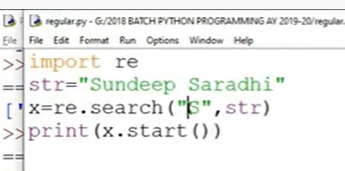


If its there itwill return match object

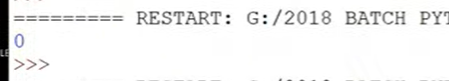






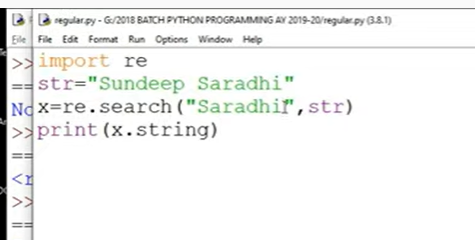


This will give index position of first match





It gives range of starting to ending index



Complete string





Split at space

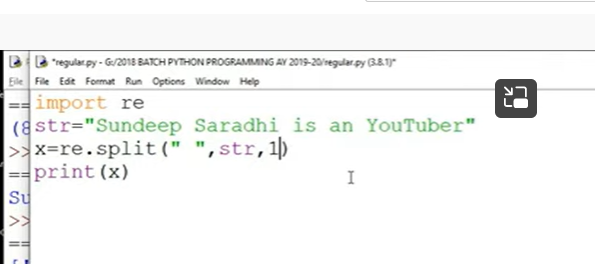


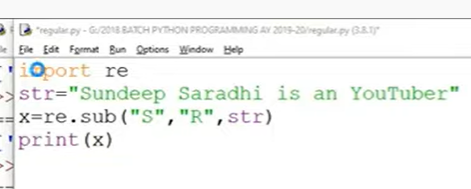
No space



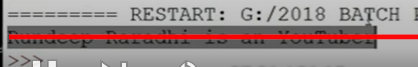


Split only at 1st space

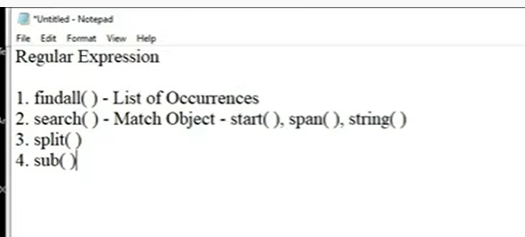




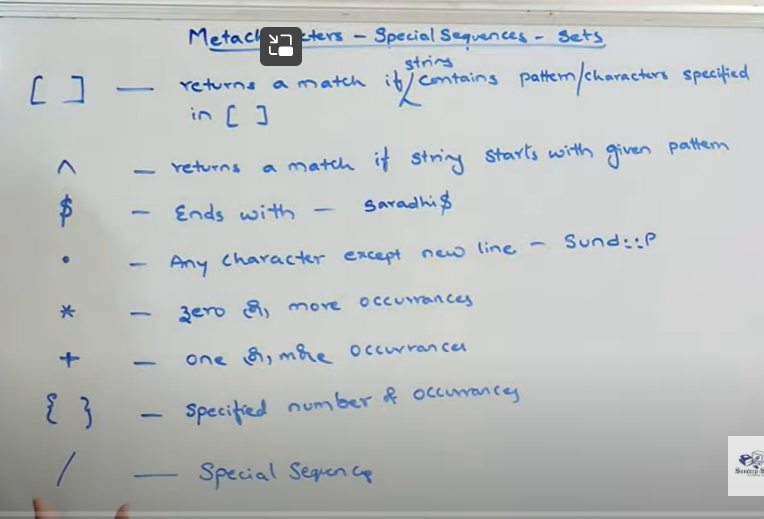
Sub is substitute

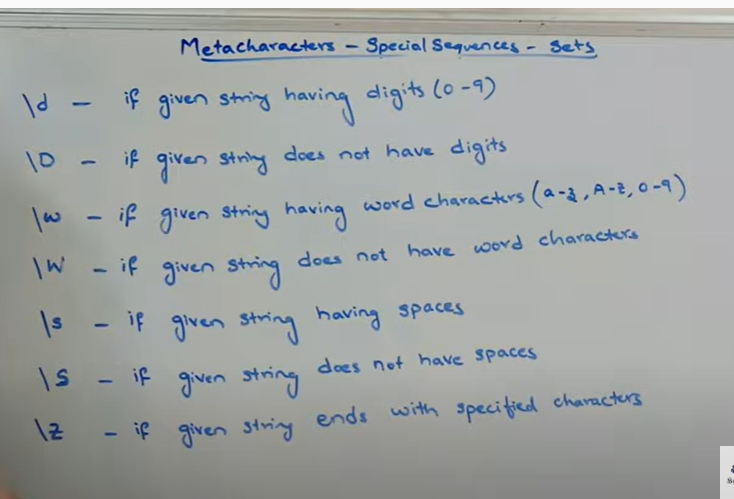


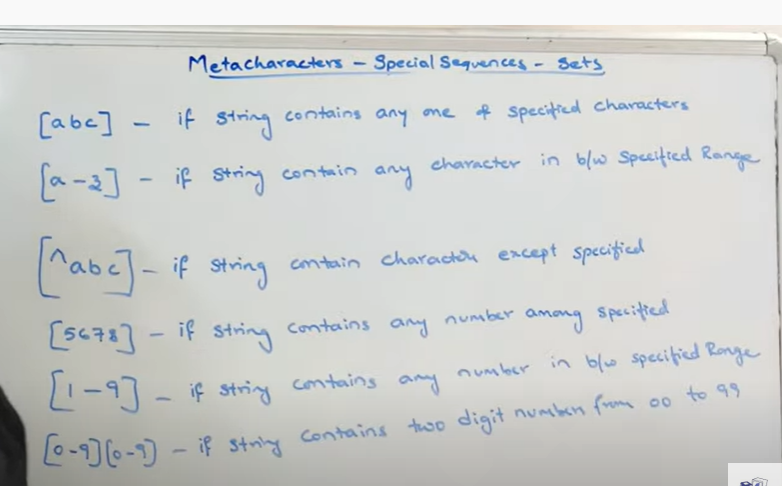
s is replaced with r

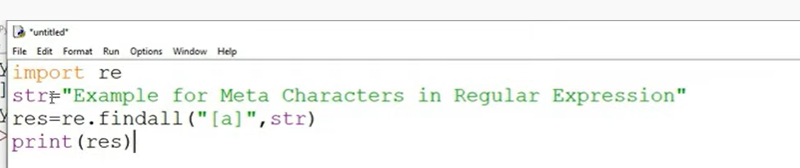


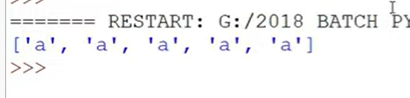
**META CHARACTERS AND SPECIAL SEQUENCES IN REGULAR EXPRESSION - PYTHON PROGRAMMING**

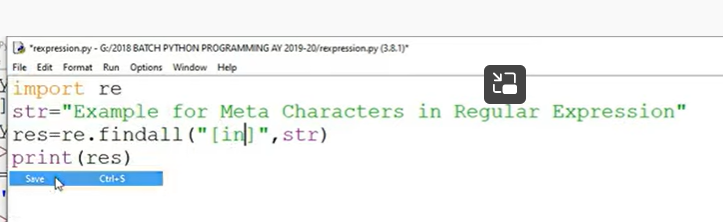




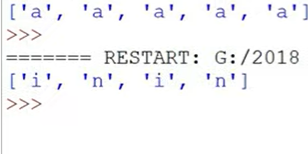


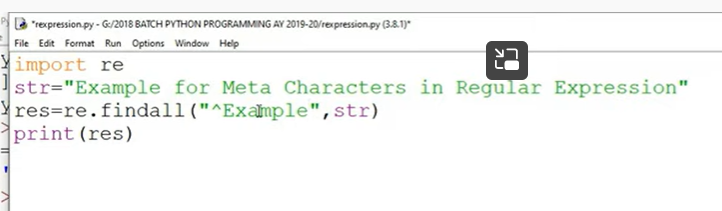




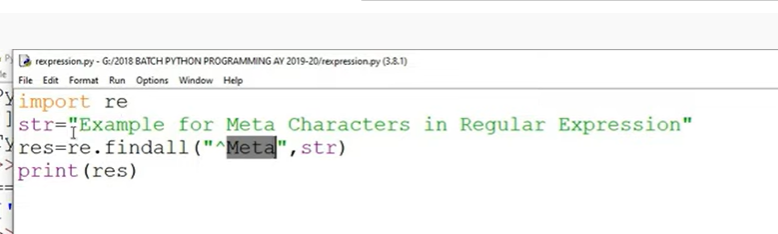


In will be taken as I separate character n separate character





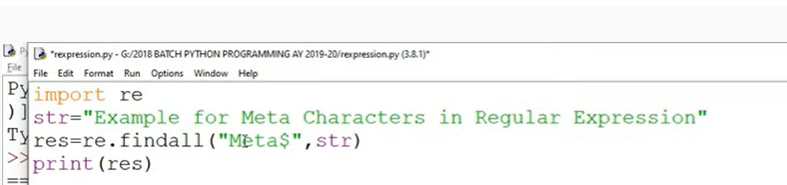




Empty list as string doest start with meta



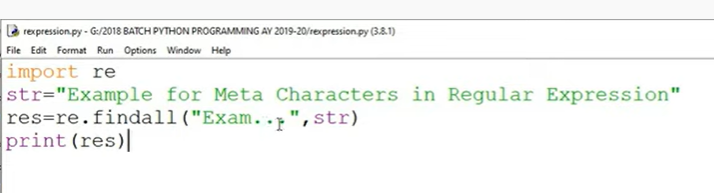
Ends with





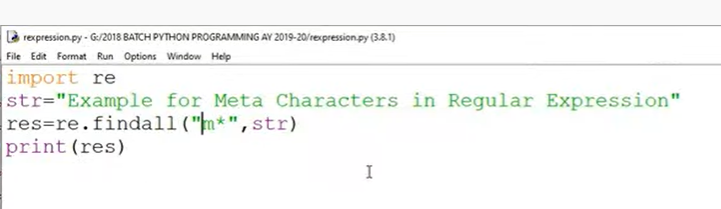






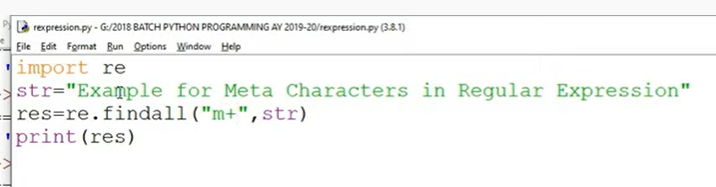
It will accept three characters



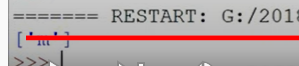


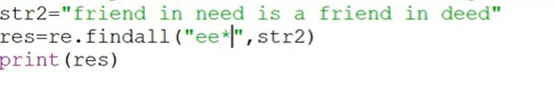
O or more occurrences



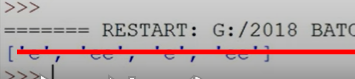


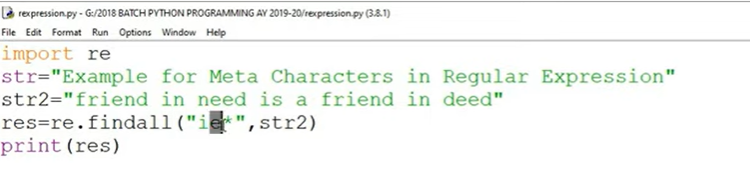
One or more

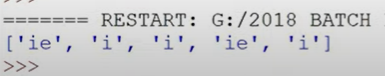


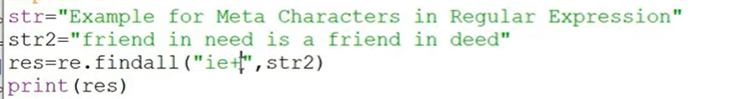


E is separate and star will apply only to last e

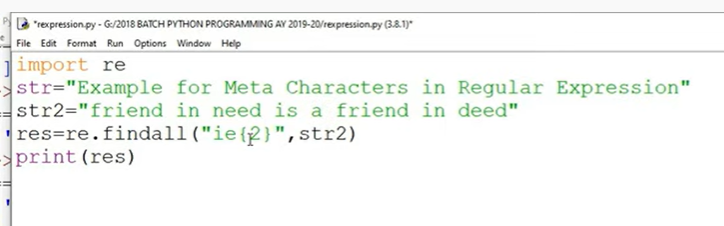




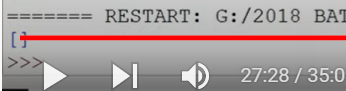








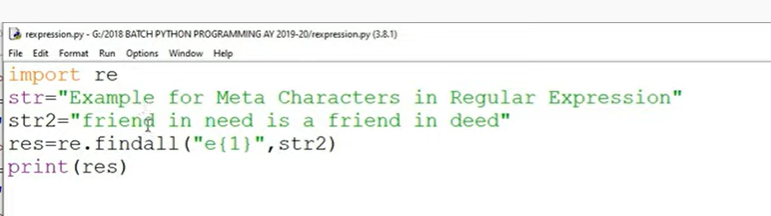
It will search for exact occurrence



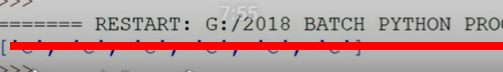
2 times of ie

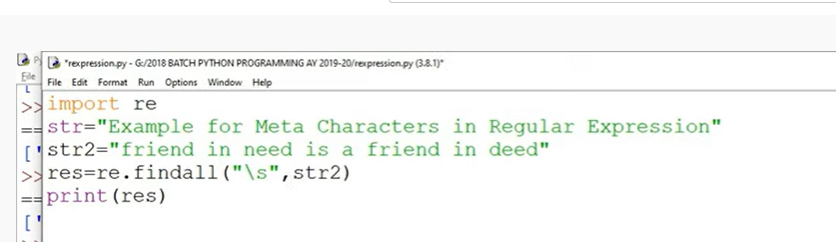




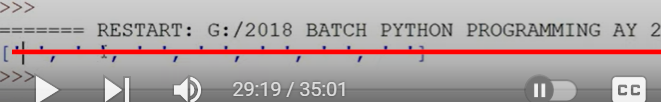


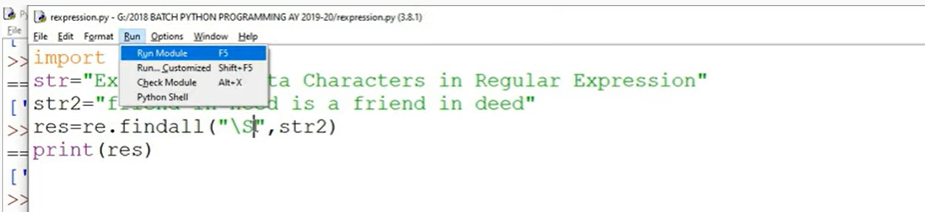
One occurrence



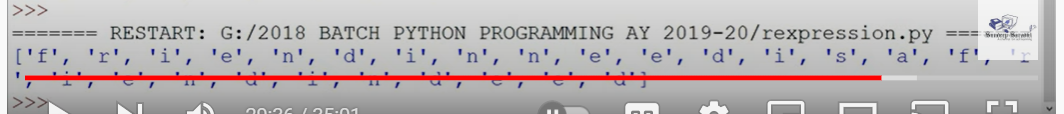


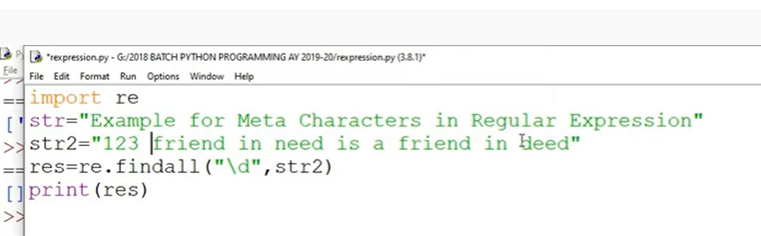
Spaces



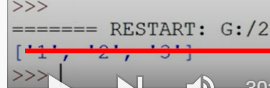


S is character without space

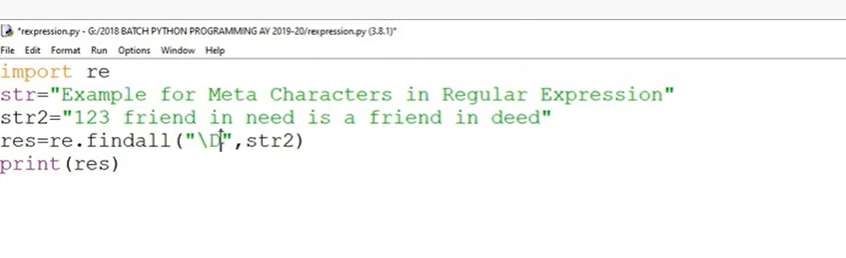


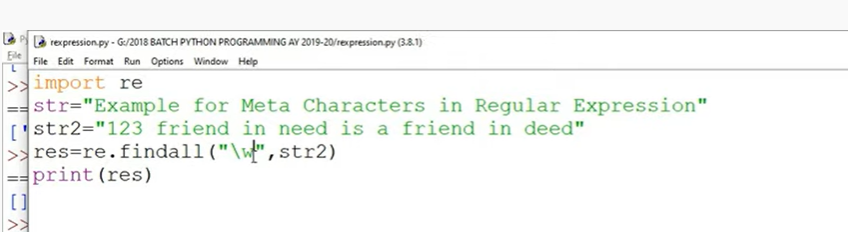


Digits

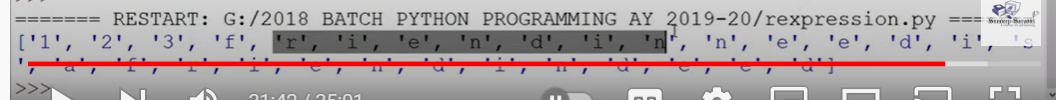


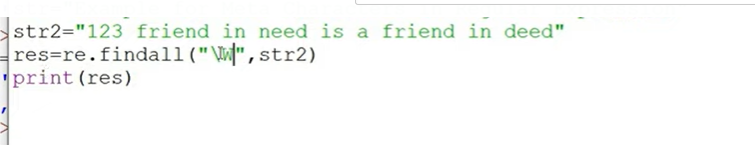
D – except digits everything



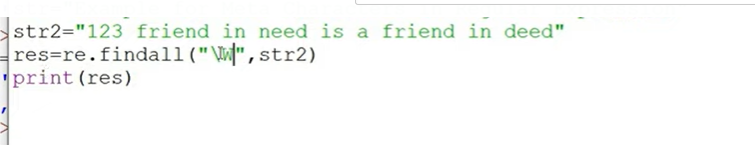


Word characters – lower, upper and numbers except space



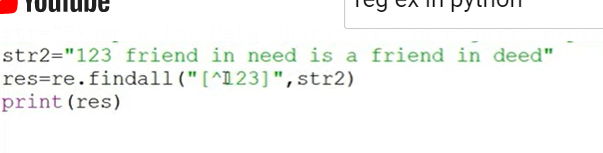


Except upper lower letters and number------ like space

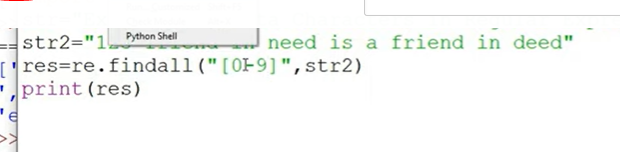


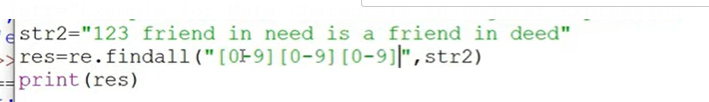


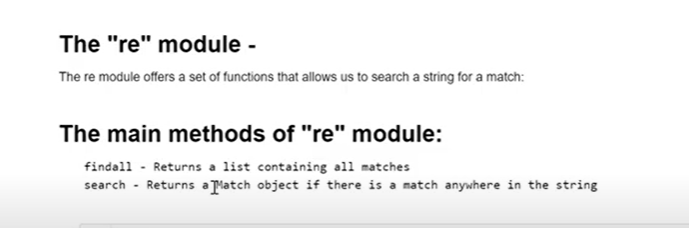
Return 1,2,3

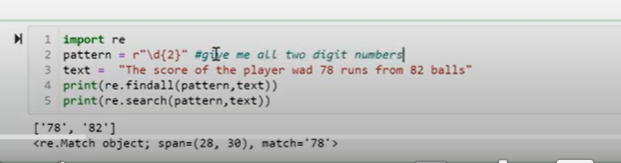


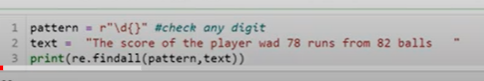
Except 1,2,3



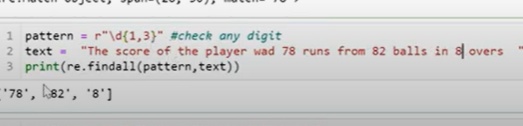




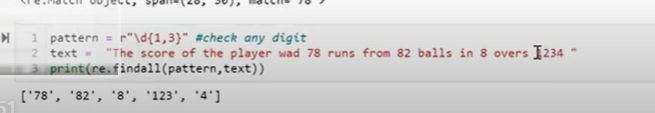




Without anything in curly braces is any digit





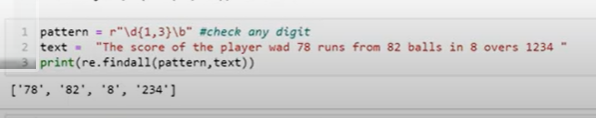


It will give splitting 4 digit number to 3 and 1

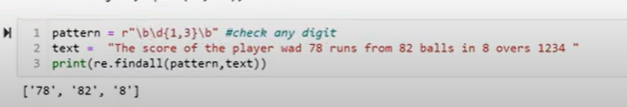
If I don’t want that number at all

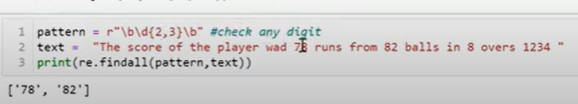
Then

Boundary line

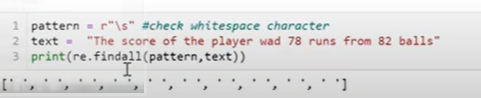


234 comes still

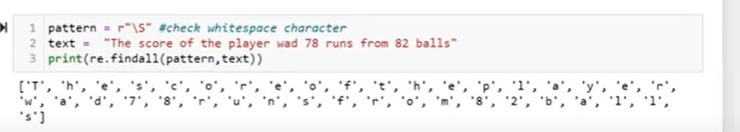




White space

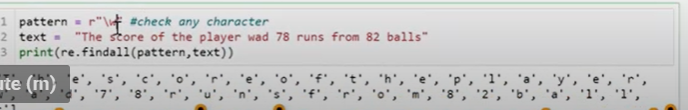


S- everything without space

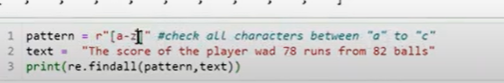


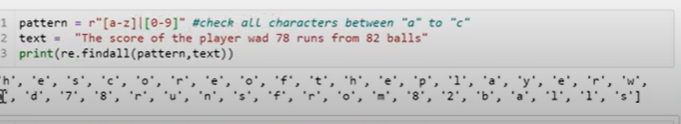
Any character .(dot)



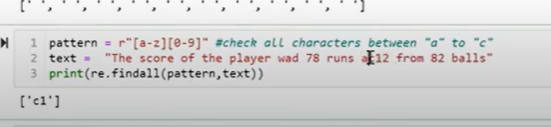


W gives everything without character

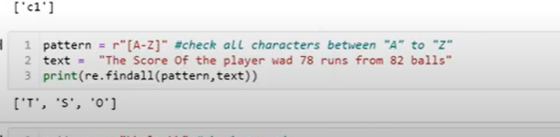


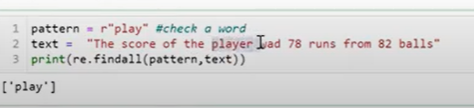


| or



Upper

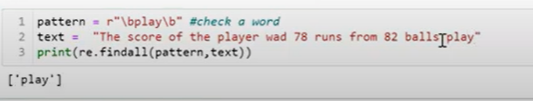


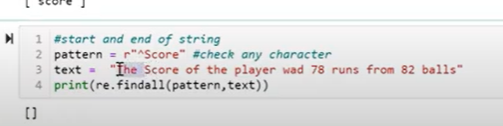


Its taking play from player

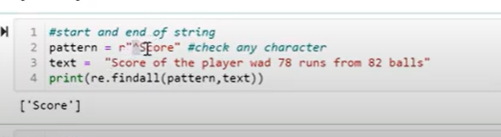
It shouldnot

Then use boundary

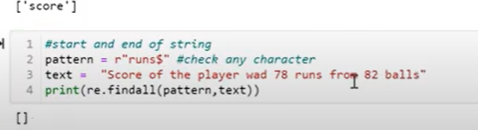


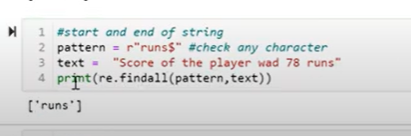


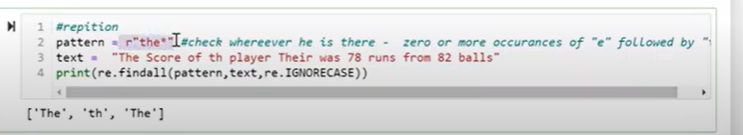
To find beginning



END word

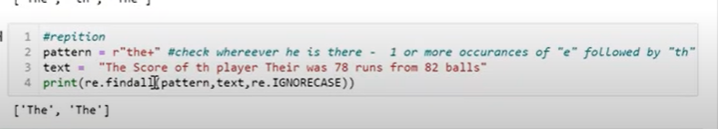






Check wherever the is there

Zero or more occurrence (\*)



One or more occurrences(+)



