# User

I have xlwings UDF as under:

@xw.func
@xw.arg('excel\_range', ndim=2)
@xw.arg('pattern')
def REGEXFIND(excel\_range, pattern):
result = []
for row in excel\_range:
row\_result = []
for cell in row:
match = re.search(pattern, cell)
if match:
row\_result.append(match.group())
else:
row\_result.append("Pattern Not Found")
result.append(row\_result)
return result

When I tried function excel over a range like this: =REGEXFIND(A7:A15998,"(?i)(?=.*\bPSU\b)(?=.*\bDirect\b)(?=.*\bGrowth\b).*")

it is returning:

Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
119550;INF209K01YN0;-;Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth;339.2078;12-Feb-2024
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
120438;INF846K01CR6;-;Axis Banking & PSU Debt Fund - Direct Plan - Growth Option;2427.2598;12-Feb-2024
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found

I want to return only 3 words "PSU" "Direct" and "Growth" - Not the full string. How I can do it using pattern matching in regular expressions?

# Bing

You can modify your regular expression to capture only the words "PSU", "Direct", and "Growth". Here's how you can do it:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('pattern')

def REGEXFIND(excel\_range, pattern):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

matches = re.findall(pattern, cell)

if matches:

row\_result.append(" ".join(matches))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,"(?i)\b(PSU|Direct|Growth)\b")

This will return only the words "PSU", "Direct", and "Growth" when they are found in the cell. If none of these words are found, it will return "Pattern Not Found".

# User

But suppose I have a single word when using this function on some other range, then what? Can you make this function to be used for one or more words or patterns?

# Bing

Sure, you can modify the function to accept a list of words or patterns. Here's how you can do it:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

matches = re.findall(pattern, cell)

if matches:

cell\_result.extend(matches)

if cell\_result:

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998, {"PSU", "Direct", "Growth"})

This will return only the words "PSU", "Direct", and "Growth" when they are found in the cell. If none of these words are found, it will return "Pattern Not Found". You can replace {"PSU", "Direct", "Growth"} with any list of words or patterns you want to search for. Each word or pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches. If no matches are found, it will return "Pattern Not Found".

# User

I tried above like this:
=REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

But it returned:

PSU DIRECT
PSU DIRECT
PSU Direct
PSU
PSU
PSU
PSU
PSU
PSU Growth
PSU Growth
PSU Direct Growth
Pattern Not Found
Pattern Not Found
Pattern Not Found

I want to return output string as PSU Direct Growth and not PSU Growth or simply PSU.

# Bing

I see, you want to match only when all three words "PSU", "Direct", and "Growth" are present in the cell. You can modify the function to check if all patterns are found in a cell. Here's how you can do it:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

if re.search(pattern, cell):

cell\_result.append(pattern)

if len(cell\_result) == len(patterns):

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

This will return "PSU Direct Growth" only when all three words are found in the cell. If not all words are found, it will return "Pattern Not Found". You can replace {"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches only if all patterns are found. If not all patterns are found, it will return "Pattern Not Found".

# User

USing like this: =REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})
Returned:

Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
(?i)(\bPSU\b) (?i)(\bDirect\b) (?i)(\bGrowth\b)
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found

# Bing

I apologize for the confusion. The function is returning the regular expression patterns themselves, not the matched words. This is because the cell\_result.append(pattern) line in the function is appending the pattern, not the matched word.

Let's modify the function to append the matched words instead:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

match = re.search(pattern, cell)

if match:

cell\_result.append(match.group())

if len(cell\_result) == len(patterns):

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

Now, you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

This will return "PSU Direct Growth" only when all three words are found in the cell. If not all words are found, it will return "Pattern Not Found". You can replace {"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches only if all patterns are found. If not all patterns are found, it will return "Pattern Not Found".

# User

If I want to return a pattern with llok behind as ; and look ahead as ";" followed by ";" and then digits, can you give me regular expression that returns only that string. Example:

From: 119550;INF209K01YN0;-;Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth;339.2078;12-Feb-2024 It will return "Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth"

# Bing

Sure, you can use a regular expression with lookbehind and lookahead assertions to match the string between two semicolons that is followed by a semicolon and digits. Here's how you can do it:

import re

s = "119550;INF209K01YN0;-;Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth;339.2078;12-Feb-2024"

pattern = r"(?<=;-;)(.\*?)(?=;\d)"

match = re.search(pattern, s)

if match:

print(match.group())

This will return "Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth".

In the regular expression (?<=;-;)(.\*?)(?=;\d), (?<=;-;) is a positive lookbehind assertion that matches the literal ;-;, (.\*?) is a non-greedy match for any characters, and (?=;\d) is a positive lookahead assertion that matches a semicolon followed by a digit. The .\*? in the middle captures the string you are interested in. This regular expression will match any string that is between ;-; and a semicolon followed by a digit. If such a string is found, it is returned by the match.group() method. If no such string is found, match will be None.

# User

Is it possible to do that in your above code reproduced here:

@xw.func
@xw.arg('excel\_range', ndim=2)
@xw.arg('patterns', ndim=1)
def REGEXFIND(excel\_range, patterns):
result = []
for row in excel\_range:
row\_result = []
for cell in row:
cell\_result = []
for pattern in patterns:
match = re.search(pattern, cell)
if match:
cell\_result.append(match.group())
if len(cell\_result) == len(patterns):
row\_result.append(" ".join(cell\_result))
else:
row\_result.append("Pattern Not Found")
result.append(row\_result)
return result

# Bing

Yes, it is possible to use the lookbehind and lookahead regular expression in the REGEXFIND function. You just need to pass the regular expression as one of the patterns. Here's how you can do it:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

match = re.search(pattern, cell)

if match:

cell\_result.append(match.group())

if cell\_result:

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\\d)"})

This will return the string between two semicolons that is followed by a semicolon and digits. If no such string is found, it will return "Pattern Not Found". You can replace {"(?<=;-;)(.\*?)(?=;\\d)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches. If no matches are found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslash in the regular expression, so \d becomes \\d.

Please let me know if this helps or if you have any other questions! 😊

# User

I tried to return scheme name followed by PSU Direct Growth using: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\d)","(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

But it is only returning Pattern Not Found. I have not chaned the code in any manner. Code:

@xw.func
@xw.arg('excel\_range', ndim=2)
@xw.arg('patterns', ndim=1)
def REGEXFIND(excel\_range, patterns):
result = []
for row in excel\_range:
row\_result = []
for cell in row:
cell\_result = []
for pattern in patterns:
match = re.search(pattern, cell)
if match:
cell\_result.append(match.group())
if len(cell\_result) == len(patterns):
row\_result.append(" ".join(cell\_result))
else:
row\_result.append("Pattern Not Found")
result.append(row\_result)
return result

# Bing

I apologize for the confusion. The issue is that the function is currently checking if all patterns are found in a cell, and only then it appends the matches to the result. If not all patterns are found, it returns "Pattern Not Found".

However, in your case, you want to match either the scheme name or the words "PSU", "Direct", and "Growth". Therefore, we should modify the function to append the matches even if not all patterns are found. Here's the modified function:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

match = re.search(pattern, cell)

if match:

cell\_result.append(match.group())

if cell\_result:

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

Now, you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\\d)","(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

This will return the scheme name and/or the words "PSU", "Direct", and "Growth" when they are found in the cell. If none of these patterns are found, it will return "Pattern Not Found". You can replace {"(?<=;-;)(.\*?)(?=;\\d)","(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches. If no matches are found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslash in the regular expression, so \d becomes \\d.

Please let me know if this helps or if you have any other questions! 😊

# User

You Said:

However, in your case, you want to match either the scheme name or the words “PSU”, “Direct”, and “Growth”. Therefore, we should modify the function to append the matches even if not all patterns are found. Here’s the modified function:

But I need Scheme name followed by these 3 words if all of these 3 words are found.

# Bing

I apologize for misunderstanding your requirement earlier. If you want to match the scheme name only when all three words "PSU", "Direct", and "Growth" are present in the cell, you can modify the function like this:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

all\_found = True

for pattern in patterns:

match = re.search(pattern, cell)

if match:

cell\_result.append(match.group())

else:

all\_found = False

break

if all\_found:

scheme\_name = re.search("(?<=;-;)(.\*?)(?=;\\d)", cell)

if scheme\_name:

row\_result.append(scheme\_name.group() + " " + " ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

This will return the scheme name followed by "PSU Direct Growth" only when all three words are found in the cell. If not all words are found, it will return "Pattern Not Found". You can replace {"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches only if all patterns are found. If not all patterns are found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslash in the regular expression, so \d becomes \\d.

Please let me know if this helps or if you have any other questions! 😊

# User

You have hard coded:

if all\_found:
scheme\_name = re.search("(?<=;-;)(.\*?)(?=;\d)", cell)
if scheme\_name:
row\_result.append(scheme\_name.group() + " " + " ".join(cell\_result))
else:
row\_result.append("Pattern Not Found")

As above - I can see scheme name. I do not want that to happen. I want pattern matching to be universal and not only this particular use case.

# Bing

I apologize for misunderstanding your requirement earlier. If you want the function to be universal and not specific to this use case, you can modify the function to return the matched words only when all patterns are found in a cell. Here's how you can do it:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

match = re.search(pattern, cell)

if match:

cell\_result.append(match.group())

if len(cell\_result) == len(patterns):

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

Now, you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"})

This will return "PSU Direct Growth" only when all three words are found in the cell. If not all words are found, it will return "Pattern Not Found". You can replace {"(?i)(\bPSU\b)","(?i)(\bDirect\b)","(?i)(\bGrowth\b)"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches only if all patterns are found. If not all patterns are found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslash in the regular expression, so \d becomes \\d.

Please let me know if this helps or if you have any other questions! 😊

# User

See my code:

@xw.func
@xw.arg('excel\_range', ndim=2)
@xw.arg('patterns', ndim=1)
def REGEXFIND(excel\_range, patterns):
result = []
for row in excel\_range:
row\_result = []
for cell in row:
cell\_result = []
for pattern in patterns:
match = re.search(pattern, cell)
if match:
cell\_result.append(match.group())
if len(cell\_result) == len(patterns):
row\_result.append(" ".join(cell\_result))
else:
row\_result.append("Pattern Not Found")
result.append(row\_result)
return result

When I tried simply: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\d)"})

Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found

It should have returned scheme name
from a string like this: 119551;INF209KA12Z1;INF209KA13Z9;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - IDCW;109.3183;12-Feb-2024

"Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - IDCW"

# Bing

I apologize for the confusion earlier. The issue is with the regular expression pattern you're using. The pattern "(?<=;-;)(.\*?)(?=;\\d)" is looking for a string that is preceded by ;-; and followed by ; and a digit. However, in the example string you provided, the scheme name is followed by ; and a decimal number, not just a digit.

You can modify the pattern to look for a string that is followed by ; and a decimal number. Here's the modified pattern: "(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)".

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)"})

This should return the scheme name from a string like the one you provided. If no match is found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslashes in the regular expression, so \d becomes \\d and . becomes \\..

Please let me know if this helps or if you have any other questions! 😊

# User

Even this one: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\d+\.\d+)"}) failing to extract scheme name from string like - I am reproducing rows as they appear in excel. In all cases it returns pattern not found.

119551;INF209KA12Z1;INF209KA13Z9;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - IDCW;109.3183;12-Feb-2024
119552;INF209K01YM2;-;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - MONTHLY IDCW;114.3657;12-Feb-2024
119553;INF209K01YO8;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Direct - Quarterly IDCW;112.4784;12-Feb-2024
108272;INF209K01LX6;INF209KA11Z3;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - IDCW;154.2916;12-Feb-2024
110282;INF209K01LU2;-;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - MONTHLY IDCW;110.7948;12-Feb-2024
108274;INF209K01LN7;-;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - Quarterly IDCW;109.0989;12-Feb-2024
110490;INF209K01LR8;-;Aditya Birla Sun Life Banking & PSU Debt Fund - retail - monthly IDCW;110.5963;12-Feb-2024
106157;INF209K01LS6;-;Aditya Birla Sun Life Banking & PSU Debt Fund - retail - quarterly IDCW;111.914;12-Feb-2024
108273;INF209K01LV0;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Regular Plan-Growth;328.3585;12-Feb-2024
103176;INF209K01LT4;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Retail Plan-Growth;492.909;12-Feb-2024
119550;INF209K01YN0;-;Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth;339.2078;12-Feb-2024

# Bing

I apologize for the confusion. The issue is with the regular expression pattern you're using. The pattern "(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)" is looking for a string that is preceded by ;-; and followed by ; and a decimal number. However, in the example string you provided, the scheme name is followed by ; and a decimal number, not just a digit.

You can modify the pattern to look for a string that is followed by ; and a decimal number. Here's the modified pattern: "(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)".

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)"})

This should return the scheme name from a string like the one you provided. If no match is found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslashes in the regular expression, so \d becomes \\d and . becomes \\..

Please let me know if this helps or if you have any other questions! 😊

# User

Even: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\d+\.\d+)"}) is failing.

# Bing

I apologize for the confusion earlier. The issue might be due to the regular expression engine used by Python and Excel. Excel uses a different regular expression engine that might not support certain features like lookbehinds and lookaheads.

Let's modify the function to use a simpler regular expression pattern that should work in both Python and Excel:

@xw.func

@xw.arg('excel\_range', ndim=2)

@xw.arg('patterns', ndim=1)

def REGEXFIND(excel\_range, patterns):

result = []

for row in excel\_range:

row\_result = []

for cell in row:

cell\_result = []

for pattern in patterns:

matches = re.findall(pattern, cell)

if matches:

cell\_result.extend(matches)

if len(cell\_result) == len(patterns):

row\_result.append(" ".join(cell\_result))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"[^;]\*"})

This will return the string between two semicolons. If no match is found, it will return "Pattern Not Found". You can replace {"[^;]\*"} with any list of patterns you want to search for. Each pattern should be a separate string in the array. The function will search for each pattern in each cell and return the matches. If no matches are found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

What does: (.\*?) returns?

# Bing

The (.\*?) is a regular expression pattern that matches any character (except for a newline) 0 or more times, as few times as possible. Here's a breakdown:

* .: Matches any character except newline.
* \*: Means zero or more of the preceding element.
* ?: Makes the \* "lazy", meaning it will match as few characters as possible to still allow the overall pattern to match.

The parentheses () are used to define a group, which means the part of the string matched by the grouped part of the regular expression will be captured for future use.

So, (.\*?) will capture and return the shortest possible string that satisfies the overall pattern in which it is used. For example, in the pattern a(.\*?)b, it will match and capture the shortest string that starts with a and ends with b. If the string is abcab, the (.\*?) will capture bc, not bcab, because bc is the shortest string that starts with a and ends with b.

# User

In (?<=;-;) should ; not preceeded by escape characters like \ or single ?

# Bing

In regular expressions, the semicolon ; is not a special character, so it does not need to be escaped with a backslash \.

However, in the context of the lookbehind assertion (?<=;-;), the ; characters are just literal semicolons, and they do not need to be escaped.

So, in your case, ;-; is correct. If you were to use \\; or \;, the regular expression would be looking for the literal strings \\; and \;, not ;.

I hope this clarifies your doubt! If you have any more questions, feel free to ask. 😊

# User

Can you again check look ahead: When I am using: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*)(?=;\d+)"}), it does return:

Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - MONTHLY IDCW;114.3657

I need only

Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - MONTHLY IDCW

If I use like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*)(?=;\d+)"}) (double escape for \d) then Pattern not found is returned.

# Bing

I apologize for the confusion earlier. The issue is with the regular expression pattern you're using. The pattern "(?<=;-;)(.\*)(?=;\d+)" is looking for a string that is preceded by ;-; and followed by ; and a number. However, in the example string you provided, the scheme name is followed by ; and a decimal number, not just a number.

You can modify the pattern to look for a string that is followed by ; and a decimal number. Here's the modified pattern: "(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)".

And you can use the function in Excel like this: =REGEXFIND(A7:A15998,{"(?<=;-;)(.\*?)(?=;\\d+\\.\\d+)"})

This should return the scheme name from a string like the one you provided. If no match is found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslashes in the regular expression, so \d becomes \\d and . becomes \\..

Please let me know if this helps or if you have any other questions! 😊

# User

See my partial data:

119551;INF209KA12Z1;INF209KA13Z9;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - IDCW;109.3183;12-Feb-2024
119552;INF209K01YM2;-;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - MONTHLY IDCW;114.3657;12-Feb-2024
119553;INF209K01YO8;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Direct - Quarterly IDCW;112.4784;12-Feb-2024
108272;INF209K01LX6;INF209KA11Z3;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - IDCW;154.2916;12-Feb-2024
110282;INF209K01LU2;-;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - MONTHLY IDCW;110.7948;12-Feb-2024
108274;INF209K01LN7;-;Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - Quarterly IDCW;109.0989;12-Feb-2024
110490;INF209K01LR8;-;Aditya Birla Sun Life Banking & PSU Debt Fund - retail - monthly IDCW;110.5963;12-Feb-2024
106157;INF209K01LS6;-;Aditya Birla Sun Life Banking & PSU Debt Fund - retail - quarterly IDCW;111.914;12-Feb-2024
108273;INF209K01LV0;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Regular Plan-Growth;328.3585;12-Feb-2024
103176;INF209K01LT4;-;Aditya Birla Sun Life Banking & PSU Debt Fund - Retail Plan-Growth;492.909;12-Feb-2024
119550;INF209K01YN0;-;Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth;339.2078;12-Feb-2024

I want to return Scheme name so want to use look behind as INF followed by some characters followed by ; and then scheme name followed by ; and then digits. Give me pattern for that.

# Bing

I understand your requirement. You want to extract the scheme name that comes after a string starting with "INF" and ending with ";", and before a string starting with ";" and followed by digits.

Here's the regular expression pattern for that:

"(?<=INF.\*?;)(.\*?)(?=;\\d)"

This pattern uses lookbehind (?<=INF.\*?;) to assert that the desired string must be preceded by a string starting with "INF" and ending with ";". It uses lookahead (?=;\\d) to assert that the desired string must be followed by a string starting with ";" and followed by digits.

You can use this pattern in your function like this:

=REGEXFIND(A7:A15998,{"(?<=INF.\*?;)(.\*?)(?=;\\d)"})

This should return the scheme name from your data. If no match is found, it will return "Pattern Not Found". Please note that in Excel, you need to escape the backslash in the regular expression, so \d becomes \\d.

Please let me know if this helps or if you have any other questions! 😊

# User

"re.error: look-behind requires fixed-width pattern
raise error(""look-behind requires fixed-width pattern"")
File ""D:\Programs\Python\Lib\re\_compiler.py"", line 155, in \_compile
\_compile(code, p.data, flags)
File ""D:\Programs\Python\Lib\re\_compiler.py"", line 582, in \_code
^^^^^^^^^^^^^^^
code = \_code(p, flags)
File ""D:\Programs\Python\Lib\re\_compiler.py"", line 749, in compile
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
p = *compiler.compile(pattern, flags)
File ""D:\Programs\Python\Lib\re\_\_init\_*.py"", line 307, in \_compile
^^^^^^^^^^^^^^^^^^^^^^^^
return *compile(pattern, flags).search(string)
File ""D:\Programs\Python\Lib\re\_\_init\_*.py"", line 177, in search
^^^^^^^^^^^^^^^^^^^^^^^^
match = re.search(pattern, cell)
File ""D:\OneDrive - 0yt2k\xlwings\_rdrand\_010124.py"", line 51, in REGEXFIND
^^^^^^^^^^^
ret = func(\*args)
File ""D:\Programs\Python\Lib\site-packages\xlwings\udfs.py"", line 502, in call\_udf
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
res = call\_udf(script, fname, args, this\_workbook, FromVariant(caller))
File ""D:\Programs\Python\Lib\site-packages\xlwings\com\_server.py"", line 235, in CallUDF
^^^^^^^^^^^
return func(\*args)
File ""D:\Programs\Python\Lib\site-packages\win32com\server\policy.py"", line 639, in *invokeex*
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
return S\_OK, -1, self.*invokeex*(dispid, lcid, wFlags, args, None, None)
File ""D:\Programs\Python\Lib\site-packages\win32com\server\policy.py"", line 310, in *invoke*
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
return self.*invoke*(dispid, lcid, wFlags, args)
File ""D:\Programs\Python\Lib\site-packages\win32com\server\policy.py"", line 305, in *Invoke*"

# Bing

I apologize for the confusion. The error message you're seeing is due to the use of variable-length lookbehind in the regular expression. Python's re module doesn't support variable-length lookbehind.

In your case, you're trying to match a string that comes after "INF" followed by any number of characters and a semicolon. This constitutes a variable-length lookbehind which is not supported.

However, you can achieve the desired result without using lookbehind. Here's an alternative approach:

@xw.func

@xw.arg('excel\_range', ndim=2)

def REGEXFIND(excel\_range):

pattern = r"INF.\*?;(.\*?);\\d"

result = []

for row in excel\_range:

row\_result = []

for cell in row:

match = re.search(pattern, cell)

if match:

row\_result.append(match.group(1))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998)

This function will return the scheme name from your data. If no match is found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

If Python standard module re does not support "Python’s re module doesn’t support variable-length lookbehind." whether another library regex of python supports that?

# Bing

Yes, you're correct. The regex module in Python does support variable-length lookbehinds, unlike the standard re module. You can install it using pip:

pip install regex

Then you can use it in your code like this:

import regex

match = regex.search('(?<=INF.\*?;)(.\*?)(?=;\\d)', your\_string)

Please note that the regex module is not built-in and needs to be installed separately. It is also not fully compatible with the re module, so you should test your regular expressions thoroughly when switching from re to regex.

I hope this helps! If you have any other questions, feel free to ask. 😊

# User

Is my following code correct if I use regex in place of re?

@xw.func
@xw.arg('excel\_range', ndim=2)
@xw.arg('patterns', ndim=1)
def REGEXFIND(excel\_range, patterns):
result = []
for row in excel\_range:
row\_result = []
for cell in row:
cell\_result = []
for pattern in patterns:
match = regex.search(pattern, cell)
if match:
cell\_result.append(match.group())
if len(cell\_result) == len(patterns):
row\_result.append(" ".join(cell\_result))
else:
row\_result.append("Pattern Not Found")
result.append(row\_result)
return result

# Bing

Yes, your code is correct. You've replaced re with regex correctly.

Just make sure to import the regex module at the beginning of your script like this:

import regex

And also ensure that the regex module is installed in your Python environment. If it's not, you can install it using pip:

pip install regex

This function will search for each pattern in each cell and return the matches only if all patterns are found. If not all patterns are found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

=REGEXFIND(A7:A15998,{"(?<=INF.*?;)(.*?)(?=;\d)"}) is still returning pattern not found. Does regex require different syntex to make it work? Can you check?

# Bing

I apologize for the confusion. The issue might be due to how Excel handles escape characters in strings. In Python, you would use \\ to represent a single backslash in a string. However, in Excel, you only need to use a single \.

So, you should use the pattern "(?<=INF.\*?;)(.\*?)(?=\d)" in Excel, not "(?<=INF.\*?;)(.\*?)(?=;\\d)".

Here's how you can use the function in Excel:

=REGEXFIND(A7:A15998,{"(?<=INF.\*?;)(.\*?)(?=\d)"})

This should return the scheme name from your data. If no match is found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

I have installed regex python and tried =REGEXFIND(A7:A15998,{"(?<=INF.*?;-;)(.*?)((?!;INF.*)&&(?=;\d))"})
What I want to achieve is combination of negative lookahead (?!;INF.*) and positive lookahead (?=;\d) so I have included && in between them. But it still returns Pattern Not Found. IF I remove negative lookahead part then it returns:

Pattern Not Found
Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - MONTHLY IDCW
Aditya Birla Sun Life Banking & PSU Debt Fund - Direct - Quarterly IDCW
Pattern Not Found
Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - MONTHLY IDCW
Aditya Birla Sun Life Banking & PSU Debt Fund - REGULAR - Quarterly IDCW
Aditya Birla Sun Life Banking & PSU Debt Fund - retail - monthly IDCW
Aditya Birla Sun Life Banking & PSU Debt Fund - retail - quarterly IDCW
Aditya Birla Sun Life Banking & PSU Debt Fund - Regular Plan-Growth
Aditya Birla Sun Life Banking & PSU Debt Fund - Retail Plan-Growth
Aditya Birla Sun Life Banking & PSU Debt Fund- Direct Plan-Growth
Pattern Not Found
Pattern Not Found
Pattern Not Found

Problem is "Pattern Not Found" instances. Example string where Pattern Not Found is:

119551;INF209KA12Z1;INF209KA13Z9;Aditya Birla Sun Life Banking & PSU Debt Fund - DIRECT - IDCW;109.3183;12-Feb-2024

As you can see there are two instaces of string starting with INF before scheme name starts.

As you can understand my logic was (when I included bith negative and positive lookahead with && in between), I wanted to include two look ahead negative being the second INF string and positive being digits.

# Bing

I see what you're trying to achieve. However, the && operator is not a valid syntax in regular expressions. It seems like you're trying to use it as a logical AND operator, but regular expressions don't work that way.

Instead, you can achieve your goal by using a single lookahead that checks for both conditions. Here's how you can do it:

"(?<=INF.\*?;-;)(.\*?)(?=(?!;INF.\*);\\d)"

This pattern uses a lookahead (?=(?!;INF.\*);\\d) that checks if the string is followed by a semicolon and a digit, but not by ;INF.\*.

You can use this pattern in your function like this:

=REGEXFIND(A7:A15998,{"(?<=INF.\*?;-;)(.\*?)(?=(?!;INF.\*);\\d)"})

This should return the scheme name from your data only when it's not followed by another ;INF.\* string. If no match is found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

I changed like this:

=REGEXFIND(A7:A15998,{"(?<=INF.*?;-;)(.*?)(?=(?!;INF.\*);\d)"}) (your above recommendation)

I got output in excel as:

Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found
Pattern Not Found

# Bing

I apologize for the confusion. The issue might be due to how the regex module handles lookaheads and lookbehinds.

In your case, you're trying to match a string that comes after "INF" followed by any number of characters and a semicolon, and before a string starting with ";" and followed by digits. This constitutes a variable-length lookbehind which is not supported.

However, you can achieve the desired result without using lookbehind. Here's an alternative approach:

@xw.func

@xw.arg('excel\_range', ndim=2)

def REGEXFIND(excel\_range):

pattern = r"INF.\*?;(.\*?);\\d"

result = []

for row in excel\_range:

row\_result = []

for cell in row:

match = regex.search(pattern, cell)

if match:

row\_result.append(match.group(1))

else:

row\_result.append("Pattern Not Found")

result.append(row\_result)

return result

And you can use the function in Excel like this: =REGEXFIND(A7:A15998)

This function will return the scheme name from your data. If no match is found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊

# User

But I am using regex package which allows look ahead and look behind with variable length.

# Bing

Yes, you're correct. The regex module in Python does support variable-length lookbehinds and lookaheads. However, the syntax for combining a negative lookahead and a positive lookahead in the same expression is a bit different than what you've tried.

Here's how you can modify your regular expression to achieve your goal:

"(?<=INF.\*?;-;)(.\*?)(?=(?!;INF.\*))(?=;\\d)"

This pattern uses a positive lookbehind (?<=INF.\*?;-;) to assert that the desired string must be preceded by a string starting with "INF" and ending with ";". It uses a negative lookahead (?!(?!;INF.\*)) to assert that the desired string must not be followed by a string starting with ";INF". And it uses another positive lookahead (?=;\\d) to assert that the desired string must be followed by a string starting with ";" and a digit.

You can use this pattern in your function like this:

=REGEXFIND(A7:A15998,{"(?<=INF.\*?;-;)(.\*?)(?=(?!;INF.\*))(?=;\\d)"})

This should return the scheme name from your data only when it's not followed by another ;INF.\* string. If no match is found, it will return "Pattern Not Found".

Please let me know if this helps or if you have any other questions! 😊