Regular Expressions (Regex)

They're Boxy But They're Good

# What do I mean by Boxy?

This is a Crazy People (movie) reference:

<https://www.youtube.com/watch?v=GTJZEK4JP0k>

Regular expression - email validation

<https://regexr.com/> OR <https://regex101.com/>

How can I find an email.address?

myemail@email.com

myemail@email.com.au

my\_email@email.co.nz

fred.smith@email.com.au

(?:[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+(?:\.[a-z0-9!#$%&'\*+/=?^\_`{|}~-]+)\*|"(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]|\\[\x01-\x09\x0b\x0c\x0e-\x7f])\*")@(?:(?:[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]\*[a-z0-9])?|\[(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?|[a-z0-9-]\*[a-z0-9]:(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21-\x5a\x53-\x7f]|\\[\x01-\x09\x0b\x0c\x0e-\x7f])+)\])

But you can get close with:

^(\w\.{0,1})+@(\w+\.)+\w+$

# Is it seriously used - too slow???

C:\Users\glen\.vscode\extensions\alefragnani.pascal-9.3.0\syntaxes\pascal.tmLanguage

# How do I write Regular Expressions?

## Email example.

Get code from above.

## Pascal Code Example:

Procedure DoStuff;

begin

end;

Procedure DoOtherStuff();

begin

end;

procedure Someprocedure(param1: string);

begin

result := '';

End;

function Some\_Function(

param1: string):string;

begin

result := '';

End;

Function TForm1.SomeFunction(): boolean;

begin

end;

## Step by step.

Procedure

[Pp]rocedure

Use case insensitive flag

\bprocedure\b

\bprocedure|function\b

\b(procedure|function)\b

\b(procedure|function)\s+\b

\b(procedure|function)\b\s\w+( --- error - bracket escape

\b(procedure|function)\b\s\w+\(

\b(procedure|function)\b\s\w+(\(){0,1}

\b(procedure|function)\b\s\w+(\(\)){0,1}

\b(procedure|function)\b\s\w+(\(.\*\)){0,1}

\b(procedure|function)\b\s\w+(\(\s\*.\*\)){0,1}

\b(procedure|function)\b\s\w+(\(\s\*.\*\)){0,1}:{0,1}

\b(procedure|function)\b\s\w+(\(\s\*.\*\)){0,1}(:\w+){0,1}

\b(procedure|function)\b\s\w+(\.\w+){0,1}(\(\s\*.\*\)){0,1}(:\*\w+){0,1};

-- why does this not work? -- space after colon

\b(procedure|function)\b\s\w+(\.\w+){0,1}(\(\s\*.\*\)){0,1}(:\s\*\w+){0,1};

# How do you read them?

Simple answer is you can’t.

Extended Flag PCRE.

\b(procedure|function)\b #Method Type

\s+ #Any Amount of whitespace

\w+ #Method Name

(\.\w+){0,1} #Optional Class Method

# Optional Parameters Section

(

\( #Open Param

\s\* #Optional space (including CR)

.\* #Any number of any char (except CR)

\) #Close Param

){0,1} #Make Param section Optional

# functions need return type

(

: #colon

\s\* #optional Space

\w+ #return type

){0,1}; #make return type optional

## Reviewing our answer

We see now there are number of holes - we can work on this eg

the “dotall” or single line option “/s” OR the “All space or All Not space [\s\S] options

# Optional Parameters

(

\( # Open Param

\s\* # Optional space (including CR)

#.\* # Any number of any char (except CR)

[\s\S]\*? #Any character non-greedy

\) # Close Param)

){0,1} # Make Param section Optional

This brings us to GREEDYness

Keep gooing until you Find the LAST place it matches

Vs Non-Greedy

Stop when you find the FIRST place it matches

You can overcome greedyness using the ? option after the **Quantifier**