Regular Exprerssions

They may look intimidating…

```
(?:(?:\r\n)?[ \t])*(?:(?:(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t]
  )+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:
  \r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(
  ?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[
  \t]))*"(?:(?:\r\n)?[ \t])*))*@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\0
  31]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\
  ](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+
  (?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:
  (?:\r\n)?[ \t])*))*|(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z
  |(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\
```

Problems

s = “call numbr 054.304.4433 and also numbr 050.442.3384”

Basics of re

```
import re
pattern = re.compile(“054”)
matches = pattern.finditer(s)
for match in matches:
        print(match)
```

Expressions…

- let’s find the dots

pattern = re.compile(“.”)

- What is going on?
- until now we’ve dealt with fixed strings not generalised patterns

pattern = re.compile(r“\d”)

pattern = re.compile(r“\w”)

pattern = re.compile(r“\s”)

pattern = re.compile(r“\D”)

Boundaries

- \b
- \B
- ^
- $

[]

- This and this
- [Tt]his
- ranges: [0-5], [A-Za-z]

Quantifiers

- Israeli cellphone: \d\d\d-\d\d\d\d\d\d\d
- basically it’s num*3 - num*7
- so: \d{3]-\d{7}
- 054-9998080 or 054-999-8080
- 054-999-?8080
- ? - 0 or one
- * - 0 or more
- + - 1 or more
- {2,4} - range

Groups

- ()
- | or
- (a|b|c)
- match.group(i)
- re.sub(pattern, “\1\-\2”, s)

Exercise

- [A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}
- what’s that?
- what can we improve here?
- https?//(www\.)?\w+\.\w+

some modules

- find
- findall
- match
- finditer