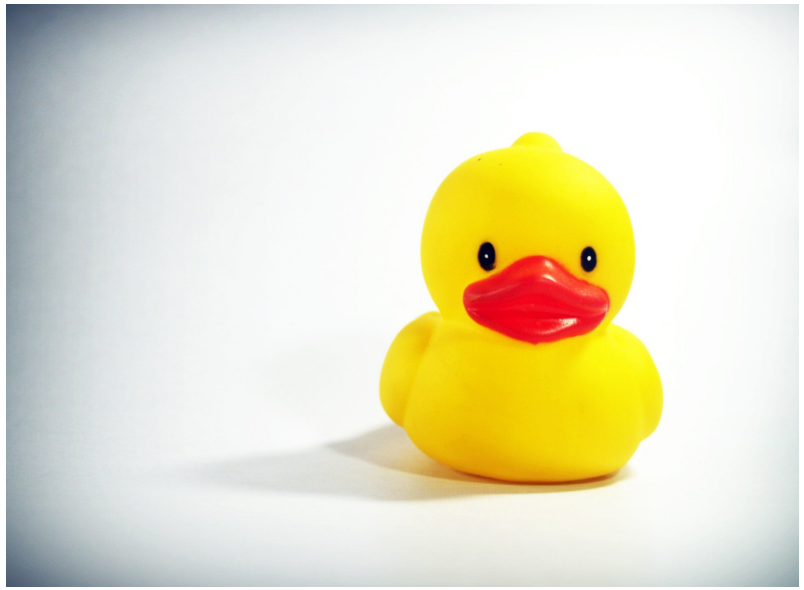
What Is a Regular Expression and Why Should I Care?

Jeff Hicks
AUTHOR/TEACHER/SENSEI

@jeffhicks | https://jdhitsolutions.com



"If it looks like a duck and quacks like a duck, it is probably a duck".

Regular Expressions are a way to identify the "ducks" you need to work with.

Patterns

192.168.10.123

roygbiv@globomantics.com

\\srv1\public

Patterns


```
\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}
```

```
\S+@\w+\.((net)|(com)|(org)|(edu)|(gov))
```

```
\\\\\S+\\\w+
```

Don’t worry about deciphering these patterns yet.

Pattern

A regular expression that defines the text you want to match.

It can be case-sensitive.

It can be a literal string.

```
# Character Classes
\w
\W
\d
\D
\s
\S
```

- Any single word character (alphanumeric)
- Any single non-word character
- Any single numeric character
- Any single non-numeric character
- Any white space character
- Any single non-white space character

```
# Values

.

"text"

[abc]

[^abc]


[x-z]
```

- Match any single character
- Match a literal string of text
- Match at least one of these characters
- Match any character *except* these characters
- Match any letter in this range (Case sensitive)

```
# Quantifiers

{n}

{n,}

{n,m}
```

- match exactly N times
- match at least N times
- match at least N times but no more than M times

Quantifiers

*

+

?

- Match 0 or more times
- Match 1 or more times
- Match 0 or 1 time

Other Options

(Jeff)

Group characters

a|b|c

Match one of these

\

The escape character

Pattern Matching

Text	Pattern
jeff-2020	\w+-\d+
	\w+-\w+
	\w{4}-\d{4}

```
PS C:\> help about_regular_expressions
```

Ask for Help

Regular expression patterns don’t necessarily validate and almost can be designed in multiple ways

```
\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}
```

IPv4 Address

Match a number (\d) at least 1 time and no more than 3 {1,3}

Followed by a literal period - \.

Repeat

```
(\d{1,3}\.){3}\d{1,3}
```

IPv4 Address – Alternate Pattern

1 to 3 numeric values followed by a period grouped in ()

Repeated exactly 3 times – {3}

Last set of 1 to 3 numeric values

```
^((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]?)$
```

IPv4 Address – Designed to Validate

Or more complex

This is a single line pattern

```
\S+@\w+\.((net)|(com)|(org)|(edu)|(gov))
```

Email Address

One or more non-whitespace characters: \S+

Followed by a literal character : @

One or more word characters : \w+

```
\S+@\w+\.((net)|(com)|(org)|(edu)|(gov))
```

Email Address

Followed by a literal period - \.

Ending in one of the specified combinations

The | separates possible domain choices

```
\\\\\S+\\\w+
```

Universal Naming Convention

Each slash in \\ needs to be escaped: first 4 slashes

Followed by one or more non-whitespace characters: \S+

**Separated by a slash: **

\\\\\S+\\\w+

Universal Naming Convention

Ending in one or more word characters: \w+

Defining a regular expression implies you know what you are looking to match.

Floating and Anchoring

Pattern	Matches
((LON)|(NYC)|(SFO))-\d{1,4}-\w{2}	Lon-3333-AB
	lon-3333-Abcde
	lon-3333-Abcde
	Zlon-3333-Abcd
	Zlon-3333-Abcd

Anchors

^

Anchor at the beginning of the line

$

Anchor at the end of the line

Floating and Anchoring

Pattern	Matches
^((LON)|(NYC)|(SFO))-\d{1,4}-\w{2}	lon-3333-AB
	lon-3333-Abcde
	~~Zlon-3333-Abcd~~

Floating and Anchoring

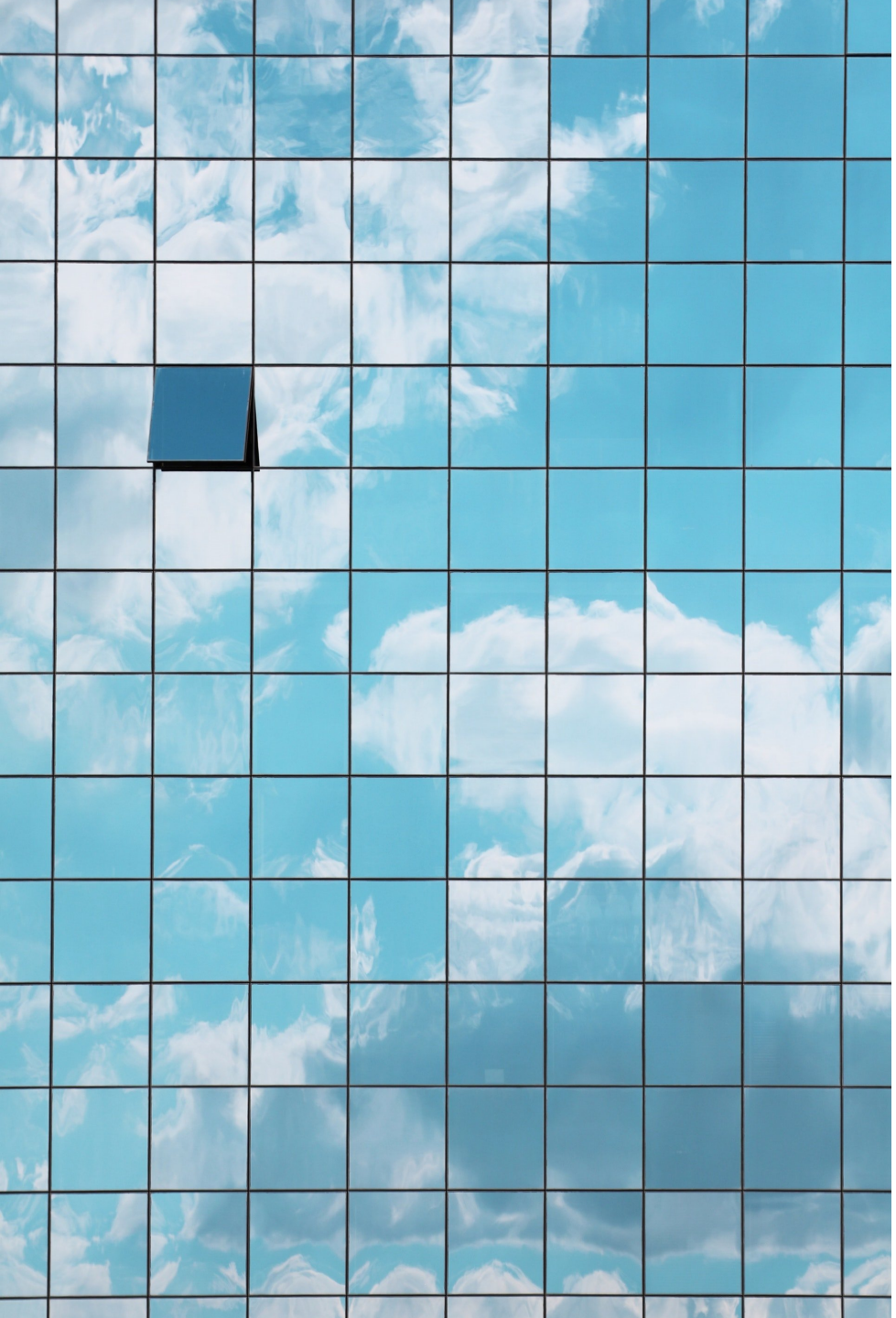
Pattern	Matches
((LON)|(NYC)|(SFO))-\d{1,4}-\w{2}$	lon-3333-AB
	~~lon-3333-Abcde~~
	~~Zlon-3333-Abcd~~

Floating and Anchoring

Pattern	Matches
^((LON)|(NYC)|(SFO))-\d{1,4}-\w{2}$	lon-3333-AB

Demo

Designing Regular Expression Patterns



Find key information in log files

Find values that match a pattern

Validate parameter input for your functions and scripts

Manipulate text

Summary

You learned the basics in designing a regular expression pattern

Beware of floating matches

Start simple

Know your data

Help about_regular_expressions