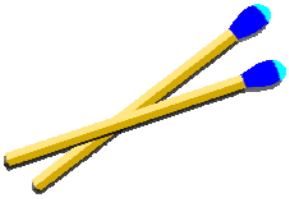


PLAYING WITH MATCHES

Regular Expressions

WHO I AM

- Mike Pullin, Systems Librarian
- mike.pullin@unthsc.edu

"UNT Health Science Center is one of the nation's premier graduate academic medical centers, with five schools that specialize in patient-centered education, research and health care."

"We help make sick people well and keep vigorous people healthy"

PLAYING WITH MATCHES

- NOTE: Title and some examples borrowed from a presentation by Richard V. Jackson, Huntington Library, San Marino, CA. Used by permission.

REGEX CHECK SITE

- We will use https://regex101.com/ to check examples
- Firas Dib is the owner/creator of the site
- Graciously gave permission for it to be used

OBJECTIVES

- Upon completion of this course, attendees will be able to:
  - Define “regular expression”
  - Detail appropriate uses
  - Describe the format and syntax
  - Design simple and complex

DOWNLOAD

- Available at http://www.txmike.com/TLA
- Under 2017
- “Playing with Matches”
- Presentation and handout are available

WHAT ARE REGULAR EXPRESSIONS?

- "Regular" – a formal designation specifying that the "language" (in this case the expressions) can be defined by a set of symbols and rules that it must follow
- "Expression" – a subset of the language constructed in such a way as to perform a particular task (in this case match a set of characters)
- "A regular expression (regex or regexp for short) is a special text string for describing a search pattern. You can think of regular expressions as wildcards on steroids."
  - From http://www.regular-expressions.info/

WHY USE REGULAR EXPRESSIONS?

- Far more versatile in string comparison than >, =, <, etc.
- Facilitates finding patterns in addition to exact matches

WHY USE REGULAR EXPRESSIONS?

- Example 1 – Text file of MARC records:
  - Find all that have:
  - 245 with a 2nd indicator not 4
  - but title starts with 'The' (with or without |a)
  - case not important
- Example 2 – Checking patron data:
  - Confirm that email address is in correct form
- Both may be accomplished EASILY using regular expressions

WHY USE REGULAR EXPRESSIONS?

- Useful in programs/programming
- Useful on Websites – checking input data
- Useful in Excel – but must use VB scripting
- Useful in Notepad++ & MarcEdit search/replace

HOW TO USE REGULAR EXPRESSIONS?

- Following the set of rules (“regular”) …
- Compose a string of characters (the “expression”)
- Use that regular expression to compare the target data

CHARACTERS USED IN REGULAR EXPRESSIONS

- **Literal characters**
- Brackets / Parentheses
- Metacharacters
- Quantifiers

CHARACTERS USED IN REGULAR EXPRESSIONS: LITERAL CHARACTERS

- Any character other than:
  - Brackets / parentheses
  - Metacharacters
  - Quantifiers
- Search: /cat/i
- Matches:
  - The cat in the hat /|c by Dr. Seuss
  - Catching fire /|c Suzanne Collins – with "i" qualifier
  - A bibliography of Austin Dobson /|cattempted by Francis Edwin Murray

CHARACTERS USED IN REGULAR EXPRESSIONS: LITERAL CHARACTERS

- Any character other than:
  - Brackets / parentheses
  - Metacharacters
  - Quantifiers
- Search: /pad+/i
  - Matches:
  - Notepad++ -- matches "pad"
- Can make a special character into a literal by "escaping" it
  - \+ becomes plus sign instead of quantifier
- Search: /pad\+/i
  - Matches:
  - Notepad++ -- matches "pad+"

CHARACTERS USED IN REGULAR EXPRESSIONS

- Literal characters
- **Brackets / Parentheses**
- Metacharacters
- Quantifiers

CHARACTERS USED IN REGULAR EXPRESSIONS: BRACKETS / PARENTHESES

- Brackets specify a "character class"
- Group of characters
- Match if any within the class are found

- Search: /wom[ae]n/i
  - Matches:
    - Woman or women
- Search: /b[aeiou]t/i
  - Matches:
    - The albatross of Midway Island
    - Robots, men, and minds
  - Does NOT match:
    - Much ado about nothing
    - Shadow of a doubt

CHARACTERS USED IN REGULAR EXPRESSIONS: BRACKETS / PARENTHESES

- Brackets specify a "character class"
- Group of characters
- Match if any within the class is found
- Can include a range

- Search: /[0-9]/
  - Matches:
    - Plan 9 from outer space
    - In the year 2525 – no "g" option so only 1st match
- Search: /[a-z]/
  - Matches:
    - In the year 2525 – lowercase so matches "n"

CHARACTERS USED IN REGULAR EXPRESSIONS: BRACKETS / PARENTHESES

- Parentheses set off a group
- "Captured group" is available using a "backreference"

- Search: /([0-9][0-9][0-9])/
  - Matches:
    - A123b – "123" stored in a backreference
  - Does NOT match:
    - A12b

CHARACTERS USED IN REGULAR EXPRESSIONS

- Literal characters
- Brackets / Parentheses
- **Metacharacters**
- Quantifiers

CHARACTERS USED IN REGULAR EXPRESSIONS: METACHARACTERS

- Characters with a special meaning
- Four characters: ^ $. |
  - ^ two meanings
    - match only at beginning
    - "not" when in a class – [^0-9] means any character except 0 through 9
- Search: /^bat/i
  - i = ignore case
  - Matches:
    - Bats of Austin
  - Does NOT match:
    - The albatross of Midway Island
- Search: /[^0-9]/
  - Matches:
    - In the year 2525
  - Does NOT match: 8175551212

CHARACTERS USED IN REGULAR EXPRESSIONS: METACHARACTERS

- Characters with a special meaning
- Four characters: ^ $. |
  - ^ two meanings (start or "not")
  - $ means match only at end
- Search: /[0-9]$/
  - Matches:
    - In the year 2525
  - Does NOT match:
    - Plan 9 from outer space

CHARACTERS USED IN REGULAR EXPRESSIONS: METACHARACTERS

- Characters with a special meaning
- Four characters: ^ $. |
  - ^ two meanings (start or "not")
  - $ means match only at end
  - . is any character
  - | means OR (i.e., this OR that)

- Search: /wom.n/i
  - Matches:
    - Woman or women or womzn but not womn
- Search: /wom(a|e)n/i
  - Matches:
    - Woman or women but not womzn

CHARACTERS USED IN REGULAR EXPRESSIONS: METACHARACTERS

- Characters with a special meaning
- Four characters: ^ $. |
- Others with leading \
  - See handout for characters
    - \d is any digit – equivalent to [0-9]
    - \D is any non-digit

- Search: /[0-9]$/ or /\d$/
  - Matches:
    - In the year 2525
  - Does NOT match:
    - Plan 9 from outer space
- Search: /[^0-9]$/ or /\D$/
  - Matches:
    - Plan 9 from outer space
  - Does NOT match:
    - In the year 2525

CHARACTERS USED IN REGULAR EXPRESSIONS

- Literal characters
- Brackets / Parentheses
- Metacharacters
- **Quantifiers**

CHARACTERS USED IN REGULAR EXPRESSIONS: QUANTIFIERS

- Quantifiers express how many characters must exist to match
  - + – match one or more
  - * – match zero or more
  - ? – match zero or one
- Search: /(\d\d\d)+/
  - Matches:
    - 817-555-1212
    - 8175551212
- Search /[\-\.]*/
  - Matches:
    - 817-555-1212 – matches before "8"
    - 817.555.1212
    - 8175551212

CHARACTERS USED IN REGULAR EXPRESSIONS: QUANTIFIERS

- Quantifiers express how many characters must exist to match
  - x{X} – match exactly X quantity
  - x{X,Y} – match X to Y quantity
  - x{X,} – match at least X quantity
- Search/\d{3}[\-\.]?\d{3}[\-\.]?\d{4}/
  - Matches:
    - 817-555-1212
    - 8175551212
    - a8175551212b
- Search/^\d{3}[\-\.]?\d{3}[\-\.]?\d{4}$/
  - Matches:
    - 817-555-1212
    - 8175551212
  - Does NOT match
    - a8175551212b

CHARACTERS USED IN REGULAR EXPRESSIONS: QUANTIFIERS

- Quantifiers express how many characters must exist to match
  - usually grouped with ()
  - ?=x – match anything followed by "x"
  - ?!x – match anything not followed by "x"
- Search /q(?=u)/
  - Matches:
    - quit
  - Does not match
    - Iraq
- Search /q(?!u)/
  - Matches:
    - Iraq
  - Does NOT match
    - quit

CHARACTERS USED IN REGULAR EXPRESSIONS: QUANTIFIERS

- Quantifiers express how many characters must exist to match
  - usually grouped with ()
  - ?=x
  - ?!x
  - ?<=x – match if preceded by "x"
  - ?<!x – match if not preceded by "x"
- Search /(?<=a)q/
  - Matches:
    - Iraq
  - Does not match
    - Quit
- Search /(?<!a)q/
  - Matches:
    - quit
  - Does NOT match
    - Quit
    - Iraq

TEST YOUR KNOWLEDGE

TEST YOUR KNOWLEDGE #1

- Let's assume we are looking at patron data
- Construct a regular expression that would check the US zip code
- Valid formats are ##### and #####-####
- Either 5 digits or 5 digits plus dash and 4 more digits

TEST YOUR KNOWLEDGE #1 – ANSWER

- Must begin the string
  - /^/

TEST YOUR KNOWLEDGE #1 – ANSWER

- Must begin the string
  - /^/
- Must have 5 digits
  - /^\d{5}/

TEST YOUR KNOWLEDGE #1 – ANSWER

- Must begin the string
  - /^/
- Must have 5 digits
  - /^\d{5}/
- Option (1 or none): dash plus 4 digits
  - /^\d{5}(\-\d{4})?/

TEST YOUR KNOWLEDGE #1 – ANSWER

- Must begin the string • /^/
- Must have 5 digits • /^\d{5}/
- Option (1 or none): dash plus 4 digits • /^\d{5}(\-\d{4})?/
- No characters allowed after • /^\d{5}(\-\d{4})?$/

TEST YOUR KNOWLEDGE #1 (EXTRA CREDIT)

- Let's assume we are looking at patron data
- Construct a regular expression that would check the US zip code
- Valid formats are ##### and #####-####
- Either 5 digits or 5 digits plus dash and 4 more digits
- Various "prefixes" (1st 3 numbers) are not used
  - For our example, let's say 000, 002, and 005 cannot begin a zip code

TEST YOUR KNOWLEDGE #1 – ANSWER

- Match 000 or 002 or 005 • /(000|002|005)/

TEST YOUR KNOWLEDGE #1 – ANSWER

- Match 000 or 002 or 005
  - /(000|002|005)/
- "Not followed by" is ?!
  - /(?!000|002|005)/

TEST YOUR KNOWLEDGE #1 – ANSWER

- Previous regex
  - /^\d{5}(\-\d{4})?$/
- Insert new part: cannot begin with 000 or 002 or 005
  - /^(?!000|002|005)\d{5}(\-\d{4})?$/

TEST YOUR KNOWLEDGE #2

- Let's assume we are looking at a subset of MARC records
- Each line is a 245 field
- Each has format:
  245nnxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
  - nn are the two indicators
  - xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx is the content of the 245 field
- Create a regex search string that will match records where the content starts with "the" but the second indicator is not 4
- Confirm that 245 is at beginning

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string
  - /^/

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string
  - /^/
- Followed by 245
  - /^245/

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string
  - /^/
- Followed by 245
  - /^245/
- Followed by any 1st ind
  - /^245./ -- dot (any char) is hard to see

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string • /^/
- Followed by 245 • /^245/
- Followed by any 1st ind • /^245./
- Followed by not 4 • /^245.[^4]/

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string • /^/
- Followed by 245 • /^245/
- Followed by any 1st ind • /^245./
- Followed by not 4 • /^245.[^4]/
- Either |a or not • /^245.[^4](\|a)?/

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string • /^/
- Followed by 245 • /^245/
- Followed by any 1st ind • /^245./
- Followed by not 4 • /^245.[^4]/
- Either |a or not • /^245.[^4](\|a)?/
- Followed by 'The' • /^245.[^4](\|a)?The/

TEST YOUR KNOWLEDGE #2 – ANSWER

- Start of string
- Followed by 245
- Followed by any 1st ind
- Followed by not 4
- Either |a or not
- Followed by 'The'
- Or 'the'

- /^/
- /^245/
- /^245./
- /^245.[^4]/
- /^245.[^4](\|a)?/
- /^245.[^4](\|a)?The/
- /^245.[^4](\|a)?[Tt]he/

TEST YOUR KNOWLEDGE #3

- Validate a US phone number
- Define the requirements

TEST YOUR KNOWLEDGE #3

- Define the requirements

Component	Digits	Number Ranges
Numbering Plan Area Code	3	[2–9] for the first digit, [0-8] for the second and [0-9] for the third digits. When the second and third digits are the same, that code is called an easily recognizable code (ERC). ERCs designate special services; e.g., 888 for toll-free service. The NANP is not assigning area codes with 9 as the second digit.
Central Office (exchange code)	3	Allowed ranges: [2–9] for the first digit, and [0–9] for both the second and third digits (however, in geographic area codes the third digit of the exchange cannot be *1* if the second digit is also *1*). 911 is never valid.
Subscriber Number	4	[0–9] for each of the four digits.

TEST YOUR KNOWLEDGE #3

- Other requirements
- The area code and central office code cannot be the same
  - 817-817-xxxx is not valid
- The central office code cannot be the same as adjacent area codes
  - 817 and 682 are both Tarrant County (Fort Worth)
  - So, 817-682-xxxx and 682-817-xxxx are not valid
- Central office codes 950, 958, 959 and 970 are reserved
- See https://en.wikipedia.org/wiki/North_American_Numbering_Plan

TEST YOUR KNOWLEDGE #3

- With what we know now, we cannot fulfill all of the restrictions (like duplicate numbers)
- We will get as much as we can
- The rest for the next part of class
- Start with a simple check then build on it

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
  - /^\d{3}/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
- /^\d{3}/
- Followed by – or . or nothing
- /^\d{3}[\-\.]?/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
- /^\d{3}/
- Followed by – or . or nothing
- /^\d{3}[\-\.]?/
- Followed by 3 digits
- /^\d{3}[\-\.]?\d{3}/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
- /^\d{3}/
- Followed by – or . or nothing
- /^\d{3}[\-\.]?/
- Followed by 3 digits
- /^\d{3}[\-\.]?\d{3}/
- Followed by – or . or nothing
- /^\d{3}[\-\.]?\d{3}[\-\.]?/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits — /^\d{3}/
- Followed by – or . or nothing — /^\d{3}[\-\.]?/
- Followed by 3 digits — /^\d{3}[\-\.]?\d{3}/
- Followed by – or . or nothing — /^\d{3}[\-\.]?\d{3}[\-\.]?/
- Followed by 4 digits at the end — /^\d{3}[\-\.]?\d{3}[\-\.]?\d{4}$/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Can combine a little
- First pattern \d{3}[\-\.]?
- Do it twice {2}
- Turns it into
- Will leave it as it was for now

/^\d{3}[\-\.]?\d{3}[\-\.]?\d{4}$/

/^(\d{3}[\-\.]?){2}\d{4}$/

TEST YOUR KNOWLEDGE #3 – ANSWER

- What are the problems with this simple regex?
  - Does not allow for ()
  - Could be nnn-nnn.nnnn or nnn.nnn-nnnn or nnn-nnnnnnn or …
  - Area code cannot start with 0 or 1
  - Middle digit of area code cannot be 9
  - Area code and central office code cannot be 911
  - Central office codes 950, 958, 959 and 970 are reserved

TEST YOUR KNOWLEDGE #3 – ANSWER

- Next level
- Going to get MORE complicated very quickly
- But, you can already do some great checking with simple regex
- Hang together and we will make it

TEST YOUR KNOWLEDGE #3 – ANSWER

- Cannot start with 911
  - /^(?!911)/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Cannot start with 911
  - /^(?!911)/
- Begins with 3 digits
  - /^(?!911)[2-9][0-8]\d/
  - First digit cannot be 0 or 1
  - Second digit cannot be 9

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
  - First digit cannot be 0 or 1
  - Second digit cannot be 9
- /^(?!911)[2-9][0-8]\d/
- Area code can have () around it with space at end
  - \s is metacharacter for a space
- /^\((?!911)[2-9][0-8]\d\)\s/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Area code can have () around it with space at end
- /^\((?!911)[2-9][0-8]\d\)\s/
- OR no () with optional . or -
- /(?!911)[2-9][0-8]\d[\.\-]?/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Area code can have () around it with space at end
- /^\((?!911)[2-9][0-8]\d\)\s/
- OR no () with optional . or –
- /(?!911)[2-9][0-8]\d[\.\-]?/
- Combine the two with vertical bar
- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
- Area code can have () around it with space at end
- OR no () with optional . or – after
- Followed by 3 digits
  - Which cannot be 911
  - 1st digit 2-9; others 0-9

- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)/
- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)(?!911)[2-9]\d\d/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits
- Area code with () and space OR optional . or –
- Followed by 3 digits
- Followed by . or – or nothing
- Followed by any 4 digits

- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)(?!911)[2-9]\d\d/
- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)(?!911)[2-9]\d\d[\-\.]?\d{4}/

TEST YOUR KNOWLEDGE #3 – ANSWER

- Begins with 3 digits …
- The end
- Not perfect, but ok

- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)(?!911)[2-9]\d\d[\-\.]?\d{4}/
- /^(\((?!911)[2-9][0-8]\d\)\s|(?!911)[2-9][0-8]\d[\.\-]?)(?!911)[2-9]\d\d[\-\.]?\d{4}$/

TEST YOUR KNOWLEDGE #4

- Real world example
- Checking EZProxy config.txt file to get all Title lines
- Use Notepad++
- Select all lines starting with Title or T or #T or #Title
- # can repeat

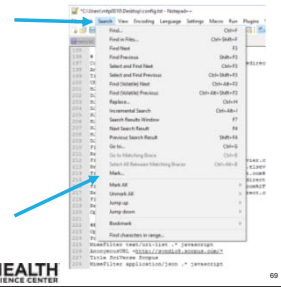
TEST YOUR KNOWLEDGE #4 – ANSWER

- Start at the very beginning
- Zero or more '#'
- Zero or more spaces
- Either 'T' or 'Title'
- A space
- Any number of characters (at least one)

- /^/
- /^#*/
- /^#*\s*/
- /^#*\s*(T|Title)/
- /^#*\s*(T|Title)\s/
- /^#*\s*(T|Title)\s.+/

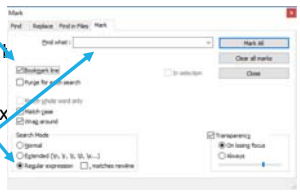
TEST YOUR KNOWLEDGE #4 – ANSWER

- Open the file you want to use
  - File -> Open -> select file
- Click on Search then Mark



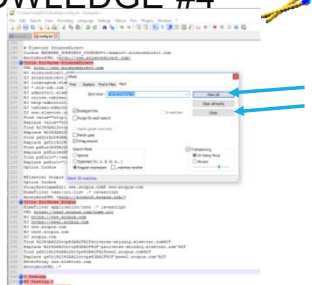
TEST YOUR KNOWLEDGE #4 – ANSWER

- Check the Bookmark line box
- Click in the Regular expression box
- Enter regex in the Find what box
  - Notepad++ box doesn't need / /



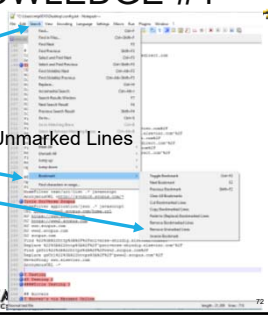
TEST YOUR KNOWLEDGE #4 – ANSWER

- Click Mark All
- Click Close



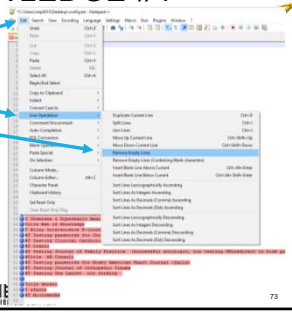
TEST YOUR KNOWLEDGE #4 – ANSWER

- Click Search
- Click Bookmark then Remove Unmarked Lines



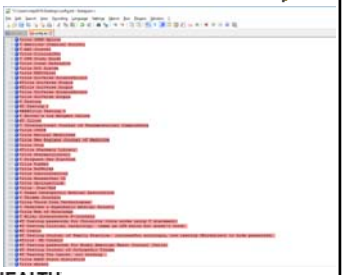
TEST YOUR KNOWLEDGE #4 – ANSWER

- Remove empty lines with Edit -> Line Operations -> Remove Empty Lines
- (Containing Blank characters)



TEST YOUR KNOWLEDGE #4 – ANSWER

- You are left with lines that match the regular expression
- Clear marks with Search -> Mark -> Clear all marks



BACKREFERENCING

- When you set something apart in a group, the match (if any) is saved in a backreference
- First match is stored in backreference \1
- Can go up to \99
- Use it by giving the reference, i.e., \1

TEST YOUR KNOWLEDGE #5

- Check a phone number to be sure area code and central office code are not the same
- Phone number in form nnn-nnn-nnnn

TEST YOUR KNOWLEDGE #5 – ANSWER

- Begin with 3 digits
- Followed by –
- Followed by 3 digits
- Followed by –
- Followed by 4 digits

- /^\d{3}$/
- /^\d{3}\-$/
- /^\d{3}\-\d{3}$/
- /^\d{3}\-\d{3}\-$/
- /^\d{3}\-\d{3}\-\d{4}$/

TEST YOUR KNOWLEDGE #5 – ANSWER

- Basic regex
- Capture 1st 3 digits

- /^\d{3}\-\d{3}\-\d{4}$/
- /^(\d{3})\-\d{3}\-\d{4}$/

TEST YOUR KNOWLEDGE #5 – ANSWER

- Basic regex
  - /^\d{3}\-\d{3}\-\d{4}$/
- Capture 1st 3 digits
  - /^(\d{3})\-\d{3}\-\d{4}$/
- Backref is in \1; check to be sure \1 does not follow -
  - /^(\d{3})\-(?!\1)\d{3}\-\d{4}$/

YOUR PROBLEMS

OTHER RESOURCES

- https://regexone.com/lesson/introduction_abcs

QUESTIONS

- Contact info:
  - Mike Pullin
  - Systems Librarian
  - UNTHSC Lewis Library
  - mike.pullin@unthsc.edu
  - www.txmike.com/TLA