lsWelcome to The Carpentries Etherpad!

This pad is synchronized as you type, so that everyone viewing this page sees the same text. This allows you to collaborate seamlessly on documents.

Use of this service is restricted to members of The Carpentries community; this is not for general purpose use (for that, try https://etherpad.wikimedia.org).

Users are expected to follow our code of conduct: https://docs.carpentries.org/topic_folders/policies/code-of-conduct.html

---

Welcome to the Library Carpentry Workshop

Southern Chapter, Medical Library Association

June 15-18, 2021

1:00pm - 5:00pm CT

Important Links

workshop website
https://ha0ye.github.io/2021-06-15-uthsc-online/

zoom link
https://ufl.zoom.us/j/99540310165?pwd=UFNWdFlaWngxVUp5aUpIM256VU1nZz09

software setup instructions:
https://ha0ye.github.io/2021-06-15-uthsc-online/index.html#setup

end of day minute cards:
https://forms.gle/C4CgyNuhkW2s7pQT9

surveys
pre - https://carpentries.typeform.com/to/wi32rS?slug=2021-06-15-uthsc-online
post - https://carpentries.typeform.com/to/UgVdRQ?slug=2021-06-15-uthsc-online

intro slides

https://docs.google.com/presentation/d/1JJBTP-VfUCcOsSQ_C5mjLGKc6lYNcxVvJGePlhPWVqU/edit?usp=sharing

Code of Conduct

To make clear what is expected, everyone participating in The Carpentries activities is required to abide by our Code of Conduct.
https://docs.carpentries.org/topic_folders/policies/code-of-conduct.html
Any form of behaviour to exclude, intimidate, or cause discomfort is a violation of the Code of Conduct. In order to foster a positive and professional learning environment we encourage you to:

- Use welcoming and inclusive language
- Be respectful of different viewpoints and experiences
- Gracefully accept constructive criticism
- Focus on what is best for the community
- Show courtesy and respect towards other community members

If you believe someone is violating the Code of Conduct, we ask that you report it to The Carpentries Code of Conduct Committee by completing this form: https://goo.gl/forms/KoUfO53Za3apOuOK2

Please use this pad to take notes, say hi to your fellow workshop attendees, and ask questions!

Day 1

Sign in: Name (Pronouns), Institution, What is the ***best*** punctuation mark?

Please sign in so we can record your attendance.

- Hao Ye (he/him), University of Florida Health Science Center Libraries, the emdash — (not to be confused with the endash, –, or the hyphen, -)
- Lynda Howell (she/her), University of Vermont
- Rosie Meindl (she/her), Rhodes College, Exclaimation Point!!!!!! (also I don't have a microphone :( ) but looking forward to learning about openrefine
- Trey Lemley (he/him), ellipsis
- Sarah Jackson, she/her, LSU Health Sciences Library, Period.
- Shannon Butcheck (she/her), Case Western Reserve University-Health Sciences Library
- Marianna Malecek (she/her), Rhodes College, ellipsis...
- Ashley Gosselar (she/her), University of Chicago Library, Oxford comma
- Erin Ware (she/her) Louisiana State University Health Sciences Shreveport Library !!!

Jargon Busting

https://librarycarpentry.org/lc-overview/03-jargon-busting/index.html

Individual Activity: Spend *3 minutes* writing down terms, phrases, or ideas around code or software development in libraries, that you would like to understand better.

docker, computing on high-performance clusters, anything related to the Julia language, MESH data structure !!, libcal integration

data wrangling, cleaning up messy legacy data, medical subject headings and authority terms/schemas, linked open data

Python, Linked Data

harvesting
R, Python
R, clean data/messy data

Best way to organize data/structures, what to do with missing data - leave blank or add N/A

Group Activity: In your breakout rooms, generate a common list of terms, then spend *10 minutes* to explaining some of the terms to each other. You are welcome to use the internet to look up definitions.

Sharing: Each breakout group will share one term that they are now comfortable with and one that they are still unsure about. (Groups should try to share new terms that haven't been listed by another group, if possible.)

Group 1

Group 2

Group 3

A Computational Approach

https://librarycarpentry.org/lc-overview/04-computational-approach/index.html

When is it useful to consider an automated or computational approach?

- You know how to automate the task.
- You think this is a task you will do over and over again.

Group Activity: In your breakout rooms, discuss some tasks that you would like to automate in your work. Take notes in the etherpad.

Group 1
Global updates in ILS cataloging
Identifying sensitive/personal information in born digital collections+1
Digital archive accessioning - migrating to preservation stable formats +1
Welcome emails and updates about what's happening in the library +1+1
Pull usage stats into single location+1
Migrate data out of old systems into new systems+1+1
Pull stats out of catalog to support analysis for annual reporting+1
Global updates to specific EAD fields across finding aids
Auto-link EAD finding aid URL to MARC field in catalog

Group 2
* Find and replace (currently on Excel)+1+1
* Input data in one column based on existing data in another column+1+1
* Scraping data from documents (litsearcher in R) +2+1
* Inputing data (multiple Excel sheets)+1+1
* Sending out emails to multiple departments+1+1+1

Group 3
* Usage statistics +1+1+1
* Lots already automated and/or done by someone else
* Reporting resources by subject area: always have to repeat the last one to confirm my methodology before doing the next, so automating would make that easier+1+1
* Keeping up with licenses, shepherding contracts through the renewal process+1+1

Takeaway points

Introduction to Working with Data (Regular Expressions)

https://librarycarpentry.org/lc-data-intro/

create "patterns" (like a search strategy for text data, similar to boolean operators)

- some characters are reserved (aka metacharacter) - these mean something different than their literal use
    - ^ [ ] \ $? * + ( ) |
    - if you want to use one of the characters, eg to "match" in the text you will need to preface with a /
- pattern match
- [ABC] A B C
- square bracket will match anything inside of it [ ] , operates similar to an "OR" in your search strategy
- [A-Z] will match any uppercase letter
- [A-Za-z] will match any letter regardless of case
- [A-Za-z0-9] any letter or number
    - in the above examples, you will not match the space
- [A-Za-z] will search space as well, must include
- **.** will match any character
- **\d** match any digit
- **\w** match any "word" character (letters, numbers, underscore)
- **\s** match any whitespace character (spaces, tabs, new lines)
- **^** does not match a character, forces a match to happen only at beginning of line
    - ^boolean will match "boolean" but only if the line betgins with "boolean"
- **$** similar to ^ , but at the end of a line
    - boolean$ will match "boolean" if at end of line
- **\b** match at a word boundary
    - \borganize\b would only match organize as its own word, and not "organized" or "disorganize" etc

Exercise 1: What will the regular expression ^[Oo]rgani.e match?

- either the british or american spelling of organize (organise) at the beginning of the line
- Organice
- organi%e
- Organi e

***** will match previous character 0 or more times

- ab will match ab or a or abbbb or etc

+ will match the previous character 1 or more times
? will match the previous character 0 or 1 time

Exercise 2: What will the regular expression \borganize\w*\b match?

- organized
- organizer
- organizers
- organize

- organizeeeeee
- organizer2
- wont match - disorganizer2
- organize12345

{2} will match the previous character exactly 2 times

- {2,4} will match previous character 2, 3, or 4 times

Exercise 3: What words will the regular expression a[a-z]{2,5} match?
Notes - {2,5} refers to the number of characters (total word limit 6 bc "a"), word must start with a lower case "a", any lowercase letter after "a" is allowed

- abacus
- add
- any lower case letter a-z I guess
- about
- across
- at (would work for a[a-z]{1-5})
- anon
- About
    - what patterns would match "About"? [A][a-z]{2,5}
        - A[a-z]{2,5}
- abbée (guess this doesn't work because [a-z] does not include accent marks)

| acts as an "OR" in boolean logic

- ((to type "|", shift + backslash))
- apple|banana will match apple OR banana

Exercise 4: What text will the regular expression [Oo]rgani.e|[Oo]rgani.e\w{1} match?
Notes - {1} means previous character must be matched 1 time, in this case the \w

- Organize OR organise
- Organi3e OR organi3e
- organiser
- OrganiYed OR Organi4es OR organi4es
- any word with an upper or lowercase o, the period can match any character
- organise_
- organize1
- organizek
- organi e
- organi!e
- organi(e8

Exercise 5: What text will the regular expression ten{1,2}es{1,2}e{1,2} match?
Note - for "ten{1,2}" this means a max of 2 "n" total, not additive

- tent esl eat (this would not work with the pattern we have - {1,2} modifies the number of times the character before the curly bracket can appear; this text would match ten.{1,2}es.{1,2}e.{1,2})
- tennessee
- tennessee
- tenese OR tennese OR tenesse OR tenesee OR tennesse OR tennesee OR tenessee OR tennessee
- tenese or tennese duh- never mind I'm replicating
- 237

Instructions:
1. Navigate to https://regex101.com/
2. paste this text in website above ("test string" area):
https://raw.githubusercontent.com/LibraryCarpentry/lc-data-intro/gh-pages/data/swcCoC.md

task: match phone numbers

- process:
    - match 10 or 11 digits [0-9]{10,11}
    - include hypens and parentheses \([0-9]{3}\)[0-9]{3}-[0-9]{4}
    - final (with country code)
    - [0-9]{0,1} {0,1}[\(-][0-9]{3}[\)-] {0,1}[0-9]{3}-[0-9]{4}

Day 2

Sign in: Name (Pronouns), Institution, favorite emoji

Please sign in so we can record your attendance.

- Erin Ware (she/her) LSU Health Shreveport, 😂 crying laughing emoji
- Shannon Butcheck Case Western Reserve University - Health Sciences Library
- Sarah Jackson, she/her, LSU Health Sciences Library
- Jess Newman, she/her, UTHSC Health Sciences Library, ¯_(ツ)_/¯
    - +1 :shrug: -> ¯_(ツ)_/¯ is the only autocorrect I have on my laptop
- Lynda Howell, she/her/hers, University of Vermont
- Rosie Meindl, she/her, Rhodes College, sideways crying laughing face
- Ashley Gosselar, she/her, UChicago Special Collections Research Center, thumbs up emoji
- Marianna Malecek, she/her, Rhodes College, smiley face with three little hearts
- Hao Ye, he/him, University of Florida Health Science Center Libraries, 😹

OpenRefine

https://librarycarpentry.org/lc-open-refine/

Data for today's lesson: https://raw.githubusercontent.com/LibraryCarpentry/lc-open-refine/gh-pages/data/doaj-article-sample.csv
* in the web browser, you can right-click and "save as" or "save page as" to download the file to your computer (you may need to add the .csv extension)

Links:

- user manual - https://docs.openrefine.org/
- Google Group - https://groups.google.com/g/openrefine
- Wikidata OpenRefine Editing documentation - https://www.wikidata.org/wiki/Wikidata:Tools/OpenRefine/Editing
- GREL syntax section of the user manual - https://docs.openrefine.org/manual/expressions#grel-general-refine-expression-language
- Recipes for GREL transforms - https://github.com/OpenRefine/OpenRefine/wiki/Recipes

What is OpenRefine?

- "Excel on steriods" - in terms of cleaning and standardizing data for analysis, a tool for messy data
- useful for splitting tabular data into more granular parts
- assists with interoperability by matching local data to other data sets
- can enhance data with external sources

Supported file formats: MARC, TSV, CSV, SQL, XML, Excel, JSON, Google spreadsheets, RDF

To launch - program launches in browser, Firefox not supported
Troubleshooting - program running slowly, memory problems - https://docs.openrefine.org/manual/installing#increasing-memory-allocation

Everything in openrefine is displayed in a tabular format (like excel, google sheets)
All data transformations are saved to your 'project,' not the original data file

Task: splitting author names in Author column

1. Edit cells -> split multi-valued cells
2. by separator -> | (pipe)

Task: combine cells

1. edit cells -> join multi-valued cells
2. select |

Activity: split Subjects column values, then rejoin

Facets

- allow you to see trends and distributions in the data

1. Column header -> facet -> text facet
2. Select "include" in facet box/panel to only view those records

Task: Get a Text Facet for the 'License' column and find a) the most common License text along with b) how many are blank
1. Column header -> facet -> text facet
2. Read counts from the side panel on the left that appears

Task: find all of the records that do not have a DOI

1. DOI column header -> facet -> customized facet -> facet by blank
2. false = there is a value, true = blank (no DOI)

* note: facets will stack on each other, so make sure to go back and check what is included/excluded if numbers seem off

Task: editing faceted value

1. Language column header -> facet -> text facet
2. in side panel/box to left, select "Edit" for English and change to "EN"

* note: when making edits with subset selected, edits will only impact selected facet

Undo/Redo will keep track of all transformations done to your data

Clustering

- cluster together similar data for normalization, eg. locations entered differently

description of clustering methods - https://docs.openrefine.org/manual/cellediting#clustering-methods

Task: cluster together author names for an individual whose name has been entered differently
1. Author column -> split multi-valued cells with |
2. Edit cells -> cluster and edit
3. openrefine will locate names it thinks are the same person (may change method and keying function, check user manual)
4. select records to merge and enter or review new cell value
5. merge selected

Columns and Sorting

Task: reorder columns

1. top right arrow for "all"
2. Re-order/Remove

Task: rename column
1. column arrow -> edit column

Task: sorting data
1. column arrow -> sort
2. will sort all based on values of selected column
3. to make permanent -> sort -> reorder pemanently

Export project after completion

- Can export data to a variety of formats
- Can also export history of transformations/actions on your data (JSON)
    - useful for data publication/FAIR data, frequently performed tasks (eg. usage data, reporting), sharing with others on team, reproducibility

Transformations

Transformation "Recipes" - https://github.com/OpenRefine/OpenRefine/wiki/Recipes

- Common transformations: trim leading/trailing whitespace, change text case

Task: remove extra (consecutive) spaces
1. drop down -> edit cells
2. common transformation -> collapse consecutive whitespace (e.g. "Title of book"

Task: collapse whitespace in citation column

Task: put titles into Titlecase
1. create facet by publisher -> facet -> text facet
2. select Akshantala Enterprices and Society of Pharmaceutical Technocrats
3. Title -> edit cells -> transform
4. in Expression box (GREL) -> value.toTitlecase()

Task: which author names are in reverse order(Last Name, First Name), put in correct order
1. split authors into separate rows
2. facet -> custom text facet
3. expression -> value.contains(",")
4. ok and select "True" from side panel
5. Author column edit cells -> transform
6. value.match(/(.*),(.*)/).reverse().join(" ")

Arrays

- a list of values (strings, dates, ets), represented by [] separated by ,

- a single object comprised of smaller objects, eg. days of the week

- Can pull in data from another source via the other source's API. Application Programming Interface is a set of functions and procedures provided by one software library or web service through which another applicant can communicate with it. An API is not the code, the database, or the server: it's the access point.
- Edit column > Add column by fetching URLs based on column x
- Comes in as JSON. Can use transform functions to edit or extract data.
    - For instance, Edit column > Add column based on column x > Value.parseJson().message.title = pulls title out of JSON and dumps it into a new column

Day 3

Sign in: Name (Pronouns), Institution, Pie, Cake or Pastry? ( )

Please sign in so we can record your attendance.

- Ashley Gosselar, she/her/hers, UChicago Library's Special Collections, Pie (served that instead of cake at my wedding)
- Esther Jackson (she/they), Columbia University, cake!
- Sarah Jackson she/her, LSU Health Sciences Library, Pie
- Shannon Butcheck, Case Western Reserve University-Health Sciences Libary, Pie
- Frances Wong, she/her, Univeristy of Toronto, Pastry - (my) world is better with butter!
- Hao Ye, he/him, University of Florida Health Science Center Libraries, a year ago I would have said pie, but I'm working my way through the recipes in "Snacking Cakes", which has been fun!
- Megan Bell, she/her, The University of Alabama at Birmingham, Cake
- Lynda Howell (she/her/hers), University of Vermont, cake.
- Erin Ware (she/her) LSU Health Shreveport, cake, but really, I'm not going to turn any of the options down!
- Trey Lemley, University of South Alabama, Pie

The Unix Shell

setup instructions (including git bash download) - https://librarycarpentry.org/lc-shell/setup.html
data download link - https://librarycarpentry.org/lc-shell/data/shell-lesson.zip
basic shell cheatsheet - https://librarycarpentry.org/lc-shell/reference.html

List of Commands

- pwd
    - Present working directory. Tells you which directory you are in
- ls
    - Shows you the content of your current directory
- ls -l
    - Shows more detail of content in current directory such as last modified date and byte count.

- ls -lh
  - Shows more “human-legible” byte counts of files in current directory.
- cd
  - Change directory. Primary way for moving around.
- cd ..
  - Move to parent directory
- cd -
  - Move back to the directory you were just in.
- ls --help
  - Gives a description of the command (in Windows)
- explorer .
  - Pops open Window's graphical file explorer window
- ls -S
  - Sort files by size
- ls -lS
  - Lists detailed info about files, sorted by file size
- mkdir
  - makes new directory
- head Shows first 10 lines of a file
- cat Prints out entire file within the Shell screen
- tail
  - lists the last 10 lines of a file
- less
  - stream the content of file. Press q to exit the mode. Press space to page through
- head -n 20 filename Shows custom number of top lines of a file (in this case, 20)
- clear
  - clears the screen; it does not clear the history on the screen you see.
- mv
  - used to rename files, move files and rename directories (folders)
- cp
  - used to copy files
- echo
  - prints the text you specify
  - you can use variable with this command example NAME=Walker
  - when using echo must use $ before variable example $NAME
  - when using "$NAME is a human" (quotation marks tell program to keep all words together on same line)
- rm
  - used to delete (use carefully because it will not verify you are deleting correct file; also file does not go into trash bin for later recovery)
- touch
  - creates files
- for
  - begins loop command
- in
  - used in loop command
- do
  - used in loop command
- done

        - ends loop command
- bash
    - interpreting command
- wc
    - counts of specified file(s); counts appear as lines, words, bytes respectively
- sort
    - sorts files least to greatest. If use -n it will sort by number, if -n not used will sort a-z.
- date
    - inserts date
- grep
    - generates regular expressions

Tips:

- When file names have a space, bracket file name with quotes
- Commands are case-sensitive
- using Tab key serves autocomplete function
- Up arrow repeats prior command. Keep pressing Up arrow, and it runs further back through your command history.
- Down arrow does the reverse of the above.
- You can use head command with multiple file names to see the first few lines of multiple files. The name of the file will be listed prior to the file contents.
- There are wildcards you can use when you want to work with a large number of files
    - ? single character and only one
    - * any file name and any number of files
- Ctrl C will exit you from and command

Setup

- Ensure the shell-lesson directory is on your Desktop

Why the Shell?

- Get familiar with command line interface
- Increase automation potential
- Work with many files with ease
- You won't need software to have a graphical client
- Foundation for programming

Navigating

- How do you list files by a) filesize and b) last created or modified?

Files and Directories

- Copying a file
- Renaming a directory
- Move a file into a directory
- Wildcards in the shell and regex
- Using echo and environment variables

Loops

```
for thing in list_of_things do
    operation_using $thing
done
```

Q: Complete the blanks in the for loop below to print the name, first line, and last line of each text file in the current directory.

```
for file in *.txt
do
        echo "$file"
        head -n 1 "$file"
        tail -n 1 "$file"
done
```

Bash script

```
#!/bin/bash

for filename in *.txt
do
 echo $filename
 head -n 5 $filename
 tail -n 5 $filename
done
```

Counting and Mining

Free Text

Day 4

Sign in: Name (Pronouns), Institution,

Please also make an account on GitHub if you do not already have one (You'll need your log in info handy): https://github.com/

Please sign in so we can record your attendance.

- Sarah Jackson she/her, LSU Health Shreveport
- Erin Ware (she, her) LSU Health Shreveport
- Lynda Howell (she/her/hers), University of Vermont
- Megan Bell (she/her) The University of Alabama at Birmingham
- Ashley Gosselar (she/her), University of Chicago Library, Special Collections
- Trey Lemley, University of South Alabama+
- Shannon Butcheck, (she/her) Health Sciences Library at Case Western Reserve University

Introduction to Git

Installation on Mac terminal.app

- xcode-select --install

Git Commands

- git config --list
    - shows configuration of the git software
- git config --global user.name "<name>"
    - set the name associated with git to <name>
- git config --global user.email "<email>"
    - set the email associated with git to <email>
- git config **--global** core.editor "notepad"
    - set the default editor for commit messages to the notepad application (for Windows)
    - note: you likely want to set a different default editor depending on your operating system
- mkdir
    - makes directory
- ls
    - list files in directory
- git init
    - initialize a new git repository in the current directory (i.e. prepare to start tracking changes

to files inside the current directory)

- git status
    - lets you know you are on branch master and gives status update of files
- git add <file>
    - adds <file> to the staging area
- git commit
    - saves the changes in the files in the staging area as a new snapshot
- git diff
    - will show differences since last commit
- git log
    - shows history of all commits in current repository
- git push
    - upload all changes to cloud
- git pull
    - download changes others have made to your local working copy
- git remote add origin
    - add link to remote file
- git checkout -b
    - creates branch

Q: Have you been using any form of version control? Whats your record for document_finalFINAL_FINALLL_v99.docx?

Share the output of your pwd once you're on your desktop, thanks!:

- /c/Users/ashleylocke/Desktop
- ~/desktop - shannon
- /c/Users/eware/Desktop
- /Users/ganymede/Desktop
- /c/Users/sjack7/Desktop
- g
-
- /Users/hye/Desktop
-
-

Refresher on Git:

1. Ashley
git init - start git in a folder

2. Erin
git status
git add - move the file to the staging area

3. Lynda
git commit - take a snapshot of the changes

4. Sarah
git diff - changes for the files
git log - history of all the commits in the current repository

Github

Troubleshooting our git hub problem

-- In Git Bash/Terminal --

Remove previous remote link:
git remote remove origin

Confirm that there is not link
git remote -v

Still in your terminal, type:
ssh-keygen -o
(hit enter a bunch of times at all the prompts)

Find where it says "Your public key has been saved in <path>"

Type:
cat <path> (the path is from the previous command)

Copy this public key for inputting in GitHub, starting at and including "ssh-rsa ... " all the way through to the "@<user>" part

-- In GitHub --

Click on your profile icon on the top right corner of any page and navigate to settings

Find SSH and GPG keys in the left panel

Press Add new

Input name for your machine ""

Paste the public key portion you saved

Navigate back to your repository (Click the top left icon and find a list of your repositories in the left side)

Make sure to flip in the "Quick Set-up" from HTTPS to SSH

Re-run the lines under "…or push an existing repository from the command line"

It will look something like this:

git remote add origin git@github.com:FrancesWong/hello-world.git
git branch -M main
git push -u origin main

You may get a prompt about an unrecognized ssh id, you can type in "yes" and enter to proceed.

Creating a new repository on github:

1. On github.com, if you are logged in, you should see your user icon in the top right (if you don't, you may need to log in first)
2. click on the + sign in the top right to bring up a menu and select new repository

Share your beautiful webpages here!

- https://franceswong.github.io/hello-world/
- https://ha0ye.github.io/hello-world/
- https://agosselar.github.io/hello-world/
-

MLA CE Credit Instructions

Congratulations on successfully completing Library Carpentry. Please follow the instructions in the etherpad to complete an evaluation and claim an MLA Certificate of Credit for your participation.

You have 30 days from the date you completed the course to complete an evaluation and claim credit.

Enrollment code: LCW621

Instructions:

1. Go to www.medlib-ed.org.
2. Login. If you do not have a current MLANET login, please Register as an MLA guest. After you've set up your MLA account and you're logged in to MLANET, click MEDLIB-ED on the navigation bar to return to MEDLIB-ED.
3. Click My Learning on the blue bar near the top of the MEDLIB-ED home page.
4. Enter the [code] and complete the attestation and evaluation and claim credit.
5. To learn more about MEDLIB-ED, please see the FAQ in the About menu.
6. If you have questions or run into problems, please email MEDLIB-ED@mail.mlahq.org.