Learning Journal

Kylie Reynolds

Semester 2, 2019

Contents

1 Learning journal: General Info

12/08/2019 - 11:31am

Instructions for this week

Before class next week, in your learning journal, finish reading to "Formatting Problems" and document the two exercises we have (hopefully) done today in your GitHub repository.

Problem: not sure how to do this or what I am meant to be documenting?? Try and find out through looking through slack comments/questions or re-listening to echo recording, if no answer found, ask on Slack.

Solution: Notes from Brian on Slack for journal entry (09/08/2019, 7:57pm):

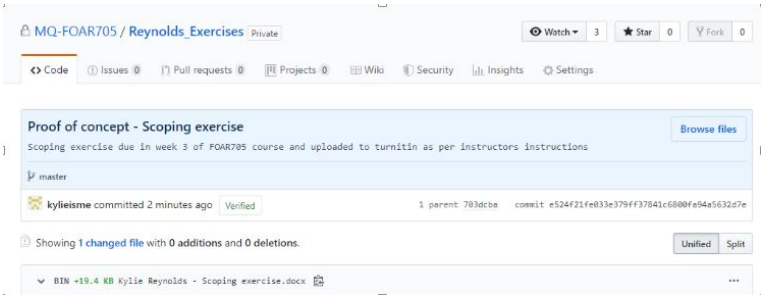
All technical work outside of class should be recorded in a laboratory notebook (document kept with the code or on Cloudstor) which documents the intention of the action, the specifics of the action taken, and the results, along with any marginal notes for improvement or updating your mental model of what should have had happened.

This documentation includes: an answer to the specific objective: "What do you intend to be the result of the action you are about to take", the action to be taken containing timestamp, and commands or actions performed, and the result, documenting what happened, success or failure in relation to the objective, and error states. Documentation of errors (each in its own entry) and their remediation are strongly encouraged.

2 GitHub Commit

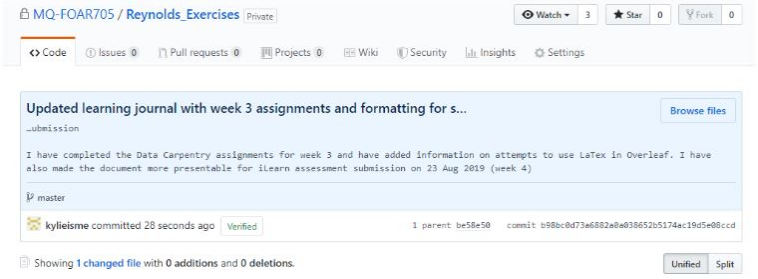
16 August 2019

- Logged in to GitHub - MQ-FOAR705/Reynolds_Exercises
- Clicked on Upload files
- Uploaded the document from my laptop file
- In Heading text bar for commit comments, I entered the file name which was "Proof of concept - scoping exercise"
- I added a comment to describe the document (see below)
- this was successfully completed with no errors (as far as I know).



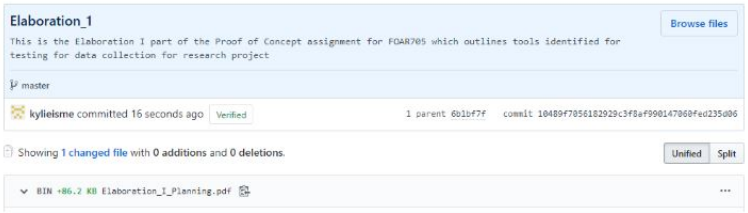
GitHub commit: Learning journal

GitHub commit: Learning Journal - Week 3 22/08/2019

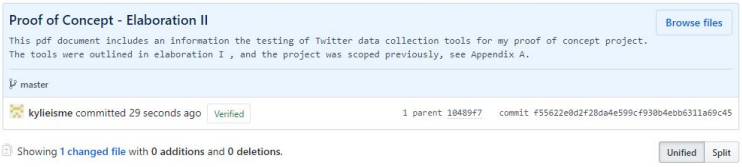


GitHub commit: Proof of Concept Scoping exercise II 24/08/2019 8:06pm

GitHub commit: Elaboration I 04/09/2019 7:54pm
I forgot to commit my PoC Elaboration I document the other day so I am doing it now since I just



remembered. See below.
GitHub commit: Elaboration II 06/09/2019 10:06pm



3 Data Carpentry

3.1 Data Organization in Spreadsheets for Social Scientists

14/08/2019 8:30pm

Objective: to read and complete exercises in DC Data Organization in Spreadsheets for Social Scientists (https://datacarpentry.org/spreadsheets-socialsci/)

3.1.1 Introduction

(https://datacarpentry.org/spreadsheets-socialsci/00-intro/index.html). Three spreadsheets were required for this and subsequent lessons in this section of the course. The spreadsheets were downloaded in the first week under instruction from presenters in the of the Macquarie University (MQ) FOAR705 class. Therefore, I did not need to download the spreadsheets for the lesson today. The spreadsheets are downloadable from:
https://datacarpentry.org/spreadsheets-socialsci/setup.html

- The three spreadsheets that were downloaded are:
  - SAFI_clean.csv
  - SAFI_messy.xlsx

  - SAFI_dates.xlsx
- they were saved in the "Data Carpentries for Social Scientists/Spreadsheets" folder on my laptop desktop.
- This task was straight forward and the instructions in this lesson were easy to follow and to execute.
- I did not have any problems or come across any errors.

Reflection: Maybe it would have been best practice to download the files again today as they may have been updated since last week.

Solution: I re-downloaded the spreadsheets and replaced the old ones in the same folder.

The introduction outlined what will be covered in this and subsequent sections of the Data Organization in Spreadsheets for Social Scientists lesson. The main objective of this lesson is to teach us best practice in the organization of data and data entry practices, and the critical importance of the data we collect for our research projects.

- To complete tasks for this lesson, a spreadsheet software is needed. The program that I will be using is excel.
- This section of the lesson lists some of the problems that can arise with spreadsheets. It spells out the usefulness of spreadsheets for data entry purposes, table and figure creation,statistics etc. However, it also warns of some of the dangers associated with these types of processes which could cause problems with research data analysis further on, some of these are:
  - White space
  - Highlighting of cells
  - Merging of cells
  - Applying calculations

Reflection: I often do all of the things listed above, and now that I understand that this can cause big problems, I will be more aware and mindful of when and how I use these features of the spreadsheet software.

Solution: Learning more about good data processing and organisational skills by continuing through the next lessons will help. And discontinuing bad practices and habits which may compromise my research data.

Exercises:

- How many people have used spreadsheets in their research?
  - I assume most researchers have used spreadsheets in their research to organise and analyse data. I have used spreadsheets to record data for research projects in uni and at work.
- How many people have accidentally done something that made them frustrated or sad?
  - I have accidentally deleted data from an original data source, and I have also organised my data in sloppy ways (as I have learnt today) which has caused problems for me and has most likely caused problems for my co-workers in the analysis stages.

3.1.2 Formatting data tables in Spreadsheets

15/08/2019 8:30 pm - 9:45 pm

Objective: To successfully complete the Data Carpentry for social scientists: Formatting data tables in Spreadsheets module
https://datacarpentry.org/spreadsheets-socialsci/01-format-data/index.html

This module is teaching us how to avoid making mistakes when working with data and how to avoid creating challenges when formatting of spreadsheets for data collection, storage and analysis.

Common mistakes are:

- adding contextual elements to the data spreadsheet - e.g. margin notes, the layout/design of the spreadsheet, and using fields to convey meaning to contextualise data (e.g. notes, colours etc., I think)
  - may confuse the computer and mess with the validity of calculations and analysis of data

Important:

- good formatting of spreadsheets from the very beginning
- Data organization is the foundation of your research project
- Data entry/analysis tip - automate conversion of files

To keep track of analyses:

- Keep the original data set and create a new file for any changes/cleaning up of data/analysis
- Track data clean-up/analysis/ step-by-step (experiment note stylz) in a new file or tab (to keep it together)

Structuring data - cardinal rules of using spreadsheet programs for data

1. Put all your variables in columns
2. Put each observation in its own row.
3. Don't combine multiple pieces of information in one cell.
4. Leave the raw data raw - don't change it!
5. Export the cleaned data to a text-based format like CSV (comma-separated values) format.
   - QUESTION: how do you do that??

Exercises:

Take a messy version of the SAFI data and describe how we would clean it up.

1. Download the messy data
    - Messy data downloaded previously (see notes and links for journal entry 14/08/2019)
    - original files saved on laptop desktop Data Carpentry for Social Scientists, Spreadsheets file.
2. Open the data in a spreadsheet program.
    - I went onto the laptop desktop, double clicked on Data Carpentry for Social Scientists folder, and then Spreadsheets folder, and double clicked on the messy excel file
    - The file auto opened in Excel.
    - I clicked on enable editing button at the top of the page to access the file
    - as this is my raw data file, I then clicked on "File" and then "Save as" and created a copy which I named - SAFI_messy_ cleaned_15082019 in the same folder as the original
    - I moved the raw data SAFI_messy file into a newly created folder in the Spreadsheets folder, which I named SAFI_RAW_data for safe keeping
3. Notice that there are two tabs. Two researchers conducted the interviews, one in Mozambique and the other in Tanzania. They both structured their data tables in a different way. Now, you're the person in charge of this project and you want to be able to start analyzing the data.
    - I have surveyed both tabs to look at the data and analyse what I am working with
    - My first thoughts are that it would be impossible to analyse the data in this format and that I would need to merge the data and organise it in a way that which makes sense and according to the good data organization principles learnt in this module and in the introduction module
4. Identify what is wrong with this spreadsheet. Discuss the steps you would need to take to clean up the two tabs, and to put them all together in one spreadsheet.

See table below

What is wrong	Steps to be taken
Data separate - needs to be merged	Merging of data: • Create new tab • create columns for each variable • each observation must have own line • Carefully and correctly enter the data into the new sheet • Double-check data entry against raw data source Possible problem: not sure how to deal with the Key_IDs as they are, they should be different for each entry and there will be duplicates if using the original data from both locations) Possible solution: (need to find out if it is ok to do this??) renumber the key_ID and make a note in a notes section which will be created in a separate tab to log what has been done, step-by-step
Contextual information and formatting that may cause analysis problems - e.g. White space, colour coding, notes in random places	• In new spreadsheet cells only to be used for recording data and column names • no merging of cells • add a column for "notes" - any notes from observation can be put there • Any random notes or colour coding etc. can be logged in a separate notes tab or file • Merged cells and white space for design purposes are a no-no
The title of the study and "2017 Data Collection" info	This information can go in the Notes tab.
Blank A column	Organise and format according to Tidy Data principles: 1. Each variable forms a column. 2. Each observation forms a row. 3. Each type of observational unit forms a table (Wickham, 2014, p. 4). The first column should start with Key_ID to identify the observation line and all other columns should continue from there with no missed columns in between

Metadata exercise:

What is not immediately obvious to me about this data?

- I opened the previously downloaded copy of SAFI_clean spreadsheet saved on laptop desktop Data Carpentries for Social Scientists, Spreadsheets folder and noticed that some of the variable column titles are not self-explanatory.
- There are also no notes connected to the document and no references to any other files which could explain the meanings of the variable titles,values, or the questions that were asked which could give me some context information which I would need when thinking about during the analysis of the data.

What questions would I need to know the answers to in order to analyze and interpret this data?

- I would need to know exactly what the column headers stand for
- I would need to know what was observed, how it was observed, the type of data that was required for each response, how data was coded, description of each variable, value meanings, etc.

3.1.3 Formatting problems

16/08/2019 9:30 am - 9:55am

Objective: To successfully complete the Data Carpentry for social scientists: Formatting problems module https://datacarpentry.org/spreadsheets-socialsci/02-common-mistakes/index.html

Common spreadsheet formatting problems/solutions:

- Using multiple tables - Don't do it - it confuses the computer, which may not be able to read associations between the tables
- Using multiple tabs - Data split between tabs - don't do it - computer cannot make connections
- Not filling in zeros - empty spaces means no data collected (null) - zero are entered if questions has been answered and the value is zero
- Using problematic null values - Blanks (most applications) and NA (for R) are best to use - check for different software
- Using formatting to convey information - do not highlight or leave blank rows in data - don't use design features in data tables, e.g. merging cells etc.
- Using formatting to make the data sheet look pretty
- Placing comments or units in cells - do not write notes in cells with other units of data - create a new notes field
- Entering more than one piece of information in a cell - each cell should only have on unit of data - create new columns instead
- Using problematic field names - Yes: wall_type OR WallType - No: wall type - no spaces and clear naming instead of abbreviations that may not make sense down the track
- Using special characters in data - Word, formatting and fancy non-standard characters can mess up data - be careful when copying and pasting data and do not use.

Reflection: The common formatting problems that are listed and demonstrated on the SAFI_messy spreadsheet are many. When I picked up problems for the last module exercises, I seemed to have missed quite a few of the mistakes that were.

Solution: create a checklist of problems to be mindful of when formatting spreadsheets for research data and to check when I have been supplied data from another person.

3.1.4 Dates as data

21/08/2019 9:30pm

Objective: To successfully complete the Data Carpentry for social scientists: Dates as data module https://datacarpentry.org/spreadsheets-socialsci/03-dates-as-data/index.html

General notes:

- Single column for dates is not best practice
  - may cause problems for computer
  - allow ambiguity to creep into your data
- functions differ between spreadsheet programs - may not be compatible if exporting

Dates as integers (whole numbers not fractions of):

- Excel counts date from December 31, 1899 and stores numbers as serial numbers (e.g. 41834 same as 07/14/14)
- must check dates for accuracy when exporting data from Excel (this link in Data carpentry lesson doesn't work)
- Formula for adding data in Excel e.g. =B2(cell with date)+90(days) = new date

Regional date formatting:

- different countries format dates differently (e.g. US month/day backwards) - this is why should not put dates in same column
- put in separate columns - 3 pieces of data (month, day, and year)
  - avoids confusion
  - better for comparison

Exercise:

- download spreadsheet - SAFI_dates.xlsx https://ndownloader.figshare.com/files/11502827 (already completed this task in first week of class) file saved in DataCarpentry/Spreadsheets folder
- I opened the spreadsheet and pressed the enable editing button at the top of the page
- I created three new columns (B, C, D) on the right side of the date column (column A where dates are formatted as 17/11/2016)
- I selected columns B, C, and D, and formatted the columns to numbers
- I named column B interview_day, C interview_month, D interview_year
- In the first row of Column B i entered =DAY(A2) and pressed enter, it showed up as 17.00
- I selected Column B C D and reduced the decimal places to zero

- In the first row of Column C I entered =MONTH(A2) and pressed enter and it showed up as 11
- In the first row of Column D I entered =YEAR(A2) and it showed up as 2016
- Default year exercise - in Cell A2 I changed the date to 17/11 and the year changed to 2019 (default year is the current one)

	A	B	C	D	
1	interview_date	interview_day	interview_month	interview_year	year
2	17/11/2019	17	11	2019	
3	17/11/2016	17	11	2016	
4	16/11/2016	16	11	2016	
5	16/12/2016	16	12	2016	
6	21/11/2016	21	11	2016	
7	21/11/2016	21	11	2016	
8	21/11/2016	21	11	2016	

- Historical data (beware):
  - pre 31 Dec 1899 Excel will leave as is
  - Be very careful when mixing before and after dates

3.1.5 Quality assurance

22/08/2019 8:37pm

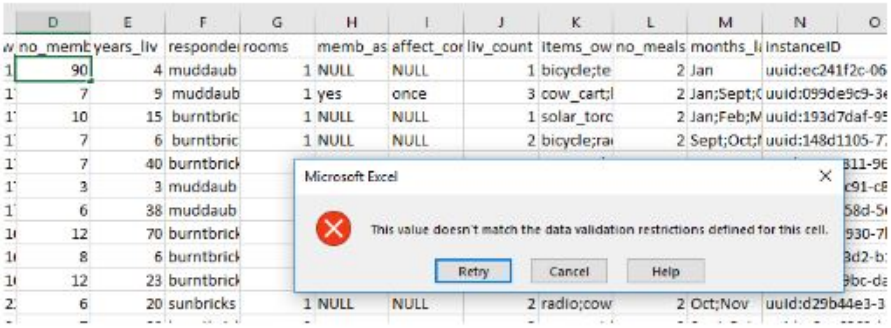
Objective: To successfully complete the Data Carpentry for social scientists: Quality assurance module https://datacarpentry.org/spreadsheets-socialsci/04-quality-assurance/index.html

Validate data on input:

- one type of data per column
- it is possible to add data restriction for values in columns and cells
- In Excel can apply data validations to cells which can raise errors to alert us and data is not entered
- In Excel, can also add validation criteria to cells with data already entered - data not removed and it is flagged by a triangle in top left of cell
- Excel validation rules available here: https://support.office.com/en-us/article/Apply-data-validation-to-cells-29FECBCC-D1B9-42C1-9D76-EFF3CE5F7249

Restricting data to a numeric range

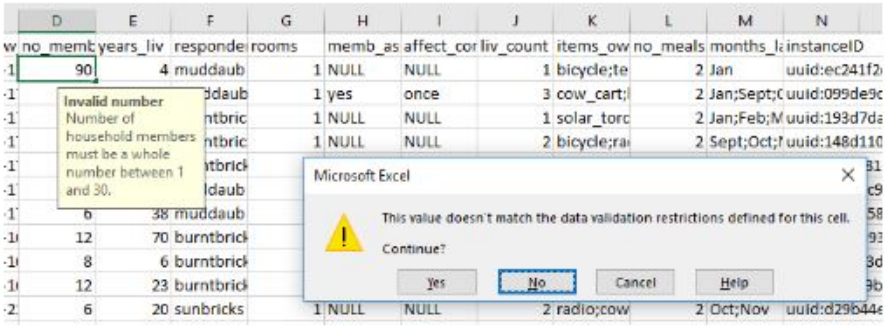
- I opened the clean SAFI file downloaded in previous lesson
- Checked column D as per instructions - numbers in this column should be integers and amount should be limited (in this instance 30 is the limit)
- I selected the Column D (no_members)
- I located the Data tab and selected Data tools and then Data validation and a pop up appeared
- I pressed the drop-down menu arrow on the right of the 1st drop down box
- I selected Whole number and entered Min. 1 and Max 30 as the range
- to test it out I entered 90 into the cell D2 and a pop up populated (see below)



- I pressed cancel to get rid of the error message
- selected Column D again and went to the data validation tool again
- in the pop up I selected Input message tab - types in the message to instruct user why the error had populated, and the rule applied to guide them in how to enter the correct data (1-30 number range)
- I then selected the Error alert tab and changed the Style to Warning.
- I then tried to re-enter the incorrect number in cell D2 to see if it was working (see below)

Exercise:

- Apply a new data validation rule to a numeric column
- Using the same clean SAFI file, I selected column G (rooms)
- I located the Data tab and selected Data tools and then Data validation and a pop up appeared
- I pressed the drop down menu arrow on the Settings drop down and I selected Whole number and entered Min. 1 and Max 7 (because there is one entry where the number is 8 and I want to test to see if the triangle appears to let me know that the data is incorrect)



- I then selected the Input message tab entered Invalid number in the title and "Number of rooms must be a whole number between 1 and 7." in the Input message
- I selected the Error Alert tab and changed the Style setting from Stop to Warning and pressed OK
- to test it out I entered 90 into the cell G2 and a pop up populated - the warning populated, and I pressed cancel and the original data remained
- I checked the cell where the number 8 was (G73) however no little triangle populated to tell me there was a validation error (hmmm..... Y?), however, there is an annoying little pink square (see below) which was sitting over the data at the top of the column which I can't get rid of, is that trying to tell me there is something wrong or is that just what happens when you add a validation rule to the column.

- To test the annoying box theory, I am going to change the 8 to a 3 to see if the box disappears
- The box did not disappear. Which is annoying but handy as I am unlikely to add the incorrect data with it in my face.
- Aha - Just found the tip at the bottom of Data Carpentry lesson that says if it is an existing spreadsheet with data already in it and it breaks the new validation rule it will not be flagged - that answers my question. Note to self: read the whole lesson so you don't waste time on unnecessary experiments when the answer is already there. Though I am more likely to remember that now, so all is well in the world.

3.1.6 Exporting data

22/08/2019 9:54pm

Objective: To successfully complete the Data Carpentry for social scientists: Exporting data module https://datacarpentry.org/spreadsheets-socialsci/05-exporting-data/index.html

- Don't save files in .xls or xlsx
  - may cause problems in the future or may not be able to open in other software or other versions of Excel
  - Some data repositories may not accept this file format
- Save files in universal, open and static format like TSV or CSV
  - "tab-delimited (tab separated values or TSV) or comma-delimited (comma separated values or CSV"
  - can use nearly any software to open and view
  - importation easy for other uses
  - may give you a warning when saving but that's ok

Exercise:

- Click File and Save As
- in File format section select CSV
- Check save location and click Save
- Success!

Note: check for commas before saving in CSV or enclose data including commas in quotation marks.

3.2 Unix Shell

4 Overleaf

13/08/2019 10:00 pm

Create an Overleaf account - http://overleaf.com/edu/macquarie

I clicked on the link to the Macquarie University link on Overleaf which was posted on the FOAR705 Slack chat (instant messaging and collaboration tool) that @Brian had put up. It is suggested that assignment and learning journal submissions be in LaTex format, and I believe this website will allow me to create this type of document. I believe that LaTex is a typesetting design program but am not sure. ACTION: find out what it does and how to operate or create documents.

I set up my sign in using my Macquarie Uni HDR ID, an email was sent to my email account for verification. Verification was actioned, and this took me to a profile page where I added details of my department (Sociology) and the degree that I am enrolled in (Masters). I noticed that I could

link the Overleaf account to my GitHub account which I did, however, I am unsure what this does or how this works. I received an email to notify me that it the GitHub account was successfully linked.

ACTION: I will need to find out about this.

- I found a button on the web page which indicated that I could start a project.
- I clicked on the button where some options came up to set up a blank document, certain other types of documents, or to select a template.
- First,I clicked on the templates link and had a quick look at what was available.
- I then decided to click back and to select a blank document. The blank page came up with some details on it. The details looked like code.
- I clicked on a PDF button on the left side of the page and it showed me a readable version of the text which was coded on the previous link.
- I went back to the code link and didn't know what to do so closed the app.

Reflection: I need to discover how to use Overleaf.

Solution: I looked up "adding text to a document in overleaf" on google and clicked on a video that showed up in the search "LaTeX video tutorial for beginners (video 1) to learn how to use the program https://www.overleaf.com/learn/latex/LaTeX_video_tutorial_for_beginners_(video_1)

14/08/2019 10:25 pm

Watched video 1 and 2 about basics and adding text to LaTex document - need to trial this process out.

Reflection: Note sure if Overleaf is called a program.

Solution: need to find out about the lingo - ask for help in Slack chat, or at consultation on Friday.

16/08/2019 12:00pm

I attempted to create a learning journal in Overleaf, and things seemed to be going well with me copying and pasting from a text document. I then recompiled the document to see what it would look like and I noticed a little note on top of the viewing panel, which I hovered over, and which said that it was a "Logs and output file". I opened the note by clicking on it and a whole bunch of information that I could not understand showed up.)See below for an example)

This overwhelmed me so I logged off and thought maybe next task I will feel more confident, I do not know how to deal with this information. I will attempt to do the scoping exercise on Overleaf instead.

18/08/2019 9:33 am

Objective: to create an Overleaf document for my Proof of concept: Scoping exercise

- I logged on to Overleaf
- Added a new project
- Named it Scoping Exercise and it automatically took me to the document.
- I copy and pasted the Introduction paragraph from my word document into the already set up Introduction section
- I clicked on recompile on the viewing screen and the text seems to have translated properly.
- I then added a subsection and named it "The jobs I will need to do"
- I copy and pasted a list of jobs underneath this heading
- I clicked the recompile button to see on the viewing page

Problem: The text showed up in a paragraph instead of a list. See image below.

Solution: I need to figure out how to change this by learning the command codes for creating a bullet list (see below)

Scoping Exercise

kylie.reynolds

August 2019

1 Introduction

Part of the research that I am currently thinking of undertaking for my MRES thesis is focused around active bystander interventions in the context of violence against women (VAW) activism online. As part of the research, I am thinking of incorporating an investigation into Twitter responses to male and female activists for the prevention of VAW, the negative responses to this support for violence prevention, and online positive bystander action in support of the allies:

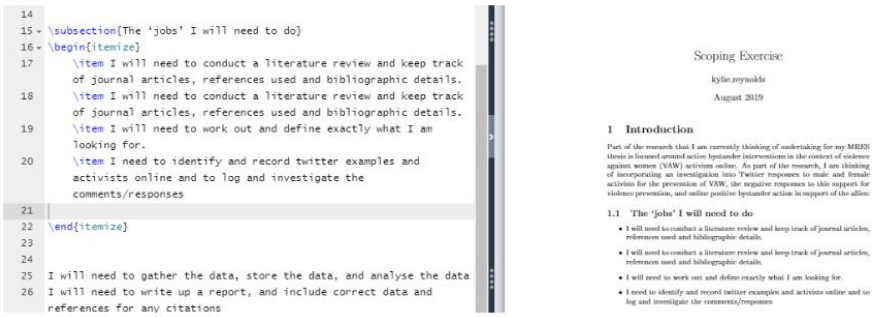
1.1 The 'jobs' I will need to do

I will need to conduct a literature review and keep track of journal articles, references used and bibliographic details I will need to work out and define exactly what I am looking for I need to identify and record twitter examples and activists online and to log and investigate the comments/responses I will need to gather the data, store the data, and analyse the data I will need to write up a report, and include correct data and references for any citations I will need to present my research in written form for submission of thesis paper for marking I would have to format a paper for publication in a journal If research is published I will need to keep track of the publication impact after publication – eg. Citations etc.

Steps:

- I looked up on Google how to create a bullet list in LaTex
- One of the links that came up was: https://latex.org/forum/viewtopic.php?t=12143 which took me to a LaTex Community forum (saved in bookmarks in Uni/FOAR705) question and answer. The answer included the command codes used to create both numbered and bullet lists, which will come in handy.
- I went to my Scoping exercise project on Overleaf and entered the commands that were presented in the answer referred to above, added a couple of list items and pressed on the recompile button to see if it was successful, and it was. See Figure A below.
- I will now continue to add the rest of the items to the list, and to create the rest of my subsections, paragraphs, and bullet lists.
- This was successful and there were no additional problems
- Once finished I downloaded a PDF version and uploaded to iLearn and Cloudstor.
- Committ to GitHub, See Figure B.

Figure A



```
14
15  \subsection{The 'jobs' I will need to do}
16  \begin{itemize}
17      \item I will need to conduct a literature review and keep track
        of journal articles, references used and bibliographic details.
18      \item I will need to conduct a literature review and keep track
        of journal articles, references used and bibliographic details.
19      \item I will need to work out and define exactly what I am
        looking for.
20      \item I need to identify and record twitter examples and
        activists online and to log and investigate the
        comments/responses
21
22  \end{itemize}
23
24
25  I will need to gather the data, store the data, and analyse the data
26  I will need to write up a report, and include correct data and
    references for any citations
```

Scoping Exercise

kylie.reynolds

August 2019

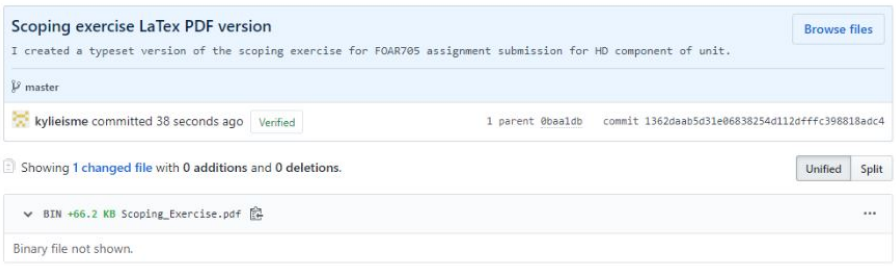
1 Introduction

Part of the research that I am currently thinking of undertaking for my MRES thesis is formed around active bystander interventions in the context of violence against women (VAW) activism online. As part of the research, I am thinking of incorporating an investigation into Twitter responses to male and female activists for the prevention of VAW, the negative responses to this support for violence prevention, and online positive bystander action in support of the allies:

1.1 The 'jobs' I will need to do

- I will need to conduct a literature review and keep track of journal articles, references used and bibliographic details.
- I will need to conduct a literature review and keep track of journal articles, references used and bibliographic details.
- I will need to work out and define exactly what I am looking for.
- I need to identify and record twitter examples and activists online and to log and investigate the comments/responses

Figure B



Scoping exercise LaTex PDF version — Browse files

I created a typeset version of the scoping exercise for FOAR705 assignment submission for HD component of unit.

master

kylieisme committed 38 seconds ago — Verified — 1 parent 0baa1db — commit 1362daab5d31e06838254d112dfffc398818adc4

Showing 1 changed file with 0 additions and 0 deletions. — Unified | Split

BIN +66.2 KB Scoping_Exercise.pdf

Binary file not shown.

21/08/2019 8:09pm

I just had a look at the marks on my scoping exercise I on iLearn and realised that I had doubled up on some bullet points in my LaTex document. I went back in to edit the document and deleted the duplicate lines. I will need to submit on GitHub and will have to remember to check/proofread my work before handing it in (see below)

Scoping Exercise I with duplicate lines deleted — Browse files

I realised that my orginal file had duplicate lines so I deleted duplicate lines and this is the updated version

master

kylieisme committed 14 seconds ago — Verified — 1 parent 1362daa — commit be58e509b28cf0474cf0f61645055dad55a95457

Showing 1 changed file with 0 additions and 0 deletions. — Unified | Split

BIN -13 Bytes (100%) Scoping_Exercise.pdf

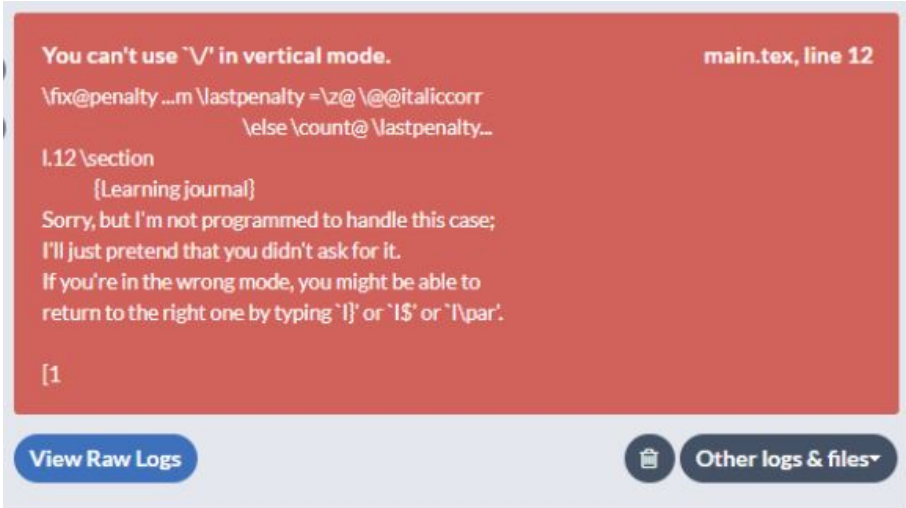
Binary file not shown.

21/08/2019 11:10pm

Objective: To typeset my learning journal to prepare for iLearn submission on Friday 23/08/2019

- I had a little trouble with the command codes the other day so I did a google search for LaTex commands and came up with a cheat sheet
  https://wch.github.io/latexsheet/latexsheet-a4.pdf
- I am going to attempt to change the font for my report
- I typed in the code and then pressed recompile to see if anything changed

Problem: an error was populated, see below:



Solution:

- I deleted the code and recompiled the document to take it back to how it was originally
- I will need to find out how to change text properly on another day when I have more time (set task in Asana for a reminder)

Objective: To reformat the Learning Journal document how I would like it

- First, I need to know how to create a blank line and to get rid of indentations
- On the cheat sheet referred to earlier it says double forward slash (can't demonstrate as it creates an error in overleaf) is the code to add a line break, and forward-slash noindent (can't demonstrate as it creates an error in overleaf) to get rid of the indent
- My strategy is to try one thing at a time so that I keep track of what works or what doesn't work
- I have copied and pasted some unformatted text from word version of learning journal
- I added the double forward-slash (can't demonstrate as it creates an error in overleaf) in front of the text where I wanted to create the new line, and pressed recompile, but that did not seem to work

- I tried adding the double forward-slash (can't demonstrate as it creates an error in overleaf) to the end of the line before where I wanted the line added and then pressed enter and then recompile and it worked.
- The line was indented so I entered forward-slash noindent (can't demonstrate as it creates an error in overleaf) at the beginning of the paragraph and it got rid of the indent
- Yay!! success.

And now it is time to go to sleep (11:43pm)

22/08/2019 10:35pm

Reflection: Turns out last night's effort was not a success – as self-punishment I have deleted everything and will start again from scratch – this week the journal will have to be submitted in Word format as I don't have time to learn how to do all that I need to do in LaTex, but, I will not give up!!

24/08/2019 10:49am

Objective: To duplicate my 1st LaTex scoping exercise and to edit the name of the document to create Scoping Exercise II. What I did:

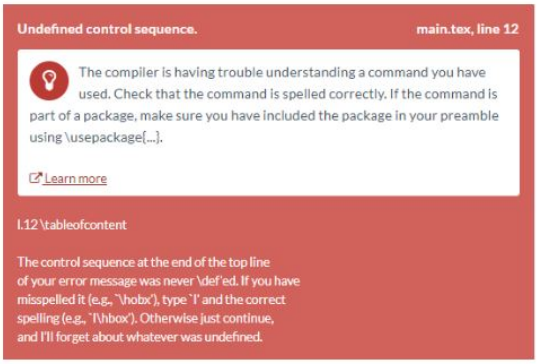
- I logged into my Projects on Overleaf
- I noticed a copy symbol at the end of the individual project rows and I pressed this button
- A new project populated called "Scoping Exercise (Copy)"
- I couldn't see any buttons which indicated that I could rename a document
- I tried checking the tick box next to the copied file and noticed that extra buttons/options showed up on the top right of the page.
- I selected the More drop down option and Rename showed up
- I clicked on rename and renamed the project Scoping Exercise II
- I also renamed the original Scoping Exercise, Scoping Exercise I
- There were no errors throughout this process
- Result: Success

24/08/2019 11:35am

Objective: To add a Contents list to my scoping document What I did:

- searched for Overleaf learn library for Table of contents
- the first link that showed up was "Table of contents - Introduction"
- I clicked on this link

- At the top of the page the code suggested was forward-slash tableofcontents, which seemed straight forward and easy
- I copy and pasted this command after the title area
- I pressed recompile to test it out and it showed up as an error (see below) which was a little confusing.

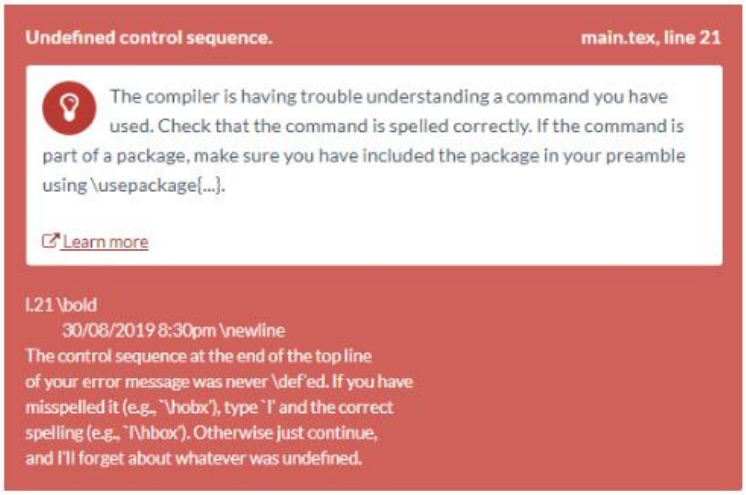


- I checked that what I had copy and pasted was correct
- It was not correct as I had missed the 's' on the end of the command
- I fixed the code and pressed recompile
- and success it worked!
- Reflection: double check when copy and pasting code.

01/09/2018 10:49am

Today I am reattempting to create my learning journal file in LaTex. I have successfully set up my title page with contents list and have added some Titles and subtitles. I am beginning with the content and want to make the date stamp bold. I tried to guess the code before looking it up.

- the code that I entered before the text that I want bold is forward-slash bold
- I then pressed the Recompile button, however, it showed an error (see below)



I went to the Overleaf help library and typed in "bold" in search bar. A page came up in the search which was Bold, italics and underline which I went to and save in my bookmarks for use at a later time: https://www.overleaf.com/learn/latex/Bold,_italics_and_underlining#Bold_text

- the code to make the date bold is:

```
\textbf{30/08/2019 8:30pm}
```

- I pressed the Recompile button again and the error was gone and the text was bold. Success!
- Commands for Italics is:

```
\textfit{add words}
```

- Command for Underline is:

```
\underline{add words}
```

- Command for Emphasis is:

```
\emph{add words}
```

06/09/2019 8:10pm

LaTex - Learning journal

I have previously presented my learning journal for assessment in word format as I have used alot of images throughout to show examples. However, I am now going to attempt to create this document in LaTex.The first thing that I am going to find out is how to add images.

Adding Images:

- I looked up "adding images" in search bar in Overleaf libary and found a page Learn LaTex in 30 minutes:
  https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes#Adding_images. I have saved this in my bookmarks as I believe this page will come in handy throughout this semester.
- I went to section 8 Adding images of this document which says to use the graphics package. See code below:

```
\documentclass{article}

\usepackage{graphicx}

\graphicspath{ {images/} }
```

- Which I copied and pasted over what was there previously at the top of the page
- It then said to upload the image that I want
- before I upload the image I will add all the text I want before the first image and format it how I want it to be
- Just added the text and I can't get anything to sit right and everything I try and do makes an error and I dont have time to look up every single thing so I will submit in word again. I hate this program!!!
- What I learnt - nothing!
- Now that I have calmed down. I am just going to try and add a picture since that is what I started out to do.
- I downloaded a random picture to my laptop desktop
- uploaded it to my learning journal overleaf project by
  - clicking the upload button
  - selecting the file and uploading

- I then typed the command:

\includegraphics{nameofimage.jpg}

- pressed recompile
- and the image was there
- HOWEVER, it is now showing a horrible little warning symbol. see below.

- sort of success I guess. Can I just ignore these horrible little annoying symbols if it all looks ok on the page?? I need to find out.
- I think I am getting somewhere now. I have added the command forward-slash, curly brackets, flushleft inside brackets and an end flushleft command at the end of the document, and all of the little annoying warning signs have dissapeared. Yay!
- I have also found out how to get rid of the dollar sign errors by adding a forward-slash sign in front - well it worked for a hyphen so I assuming it will work for other if not I will come to the bridge when it happens
- I will now continue calmly to add in text and images for learning journal and will document any problemos.

Next task: I need to find out how to make my typed commands text so that the computer doesn't recognise as commands instead of text eg forward-slash.

5 Proof of Concept

5.1 Elaboration stage

5.1.1 Applying for an API from Twitter

30/08/2019 8:30pm

As part of my proof of concept I am trying to discover how to collect a specific set of data from Twitter. I have been advised that to be able to collect that data, if I am unable to find a publicly available tool that is free, is to create some code to be able to get the data from Twitter directly. To do this I will need and Application Program Interface (API) keyfrom Twitter which I need to apply for. The application process will be documented below:

6 The Unix Shell

6.1 Introducing the Shell

28/08/2019 8:30pm

Objective: To successfully complete module 1 of the course with the intention of learning what a shell is and what it does - http://swcarpentry.github.io/shell-novice/01-intro/index.html

- GUI - graphical user interface - most common way that general computer users interact with computers - mouse, keyboard etc - good for general and easy stuff - not good for larger more complicated tasks - would take too long
- CLI - command-line interface - faster and good for automation of repetitive tasks -uses new language, combinations of commands and parameters.
- REPL - read-evaluate-print loop - "The heart of a command-line interface" - process:
    1. type command and press Enter
    2. shell reads command
    3. evaluates/executes
    4. output of command printed
    5. loops around and waits for next command

6.1.1 The Shell

- The Boss:
    - tells programs (simple [stand alone tasks AKA commands] and complicated [eg. modelling software]) what to do
    - it doesn't do the work itself.
- The most popular shell is Bash - "default shell on most modern implementations of Unix [whatever that is? better find out] and in most packages that provide Unix-like tools for Windows"

Activity:

- I opened the GitHub Bash shell that was downloaded in class because I am curious and need to know if the instructions work.
- There was dollar sign the prompt as the instructions indicated - most commonly the prompt is dollar sign but can be different
- the lesson says that the prompt "ls" will list the contents of the current directory that I am in
- it listed a whole bunch of things in different coulours - Do the colours mean anything? looks like they could be different types of files, some show what is on my PC desktop and others (different colour) show eg. application data, which seem like background files
- A new prompt is showing at the end, and example of REPL, waiting for a new command from me (the Big Boss, lol).

- The next instruction says if you accidentally type in the wrong command eg. “ks” it will tell you command not found
- I tested this and this was correct
- Question? - so what happens if the command that you incorrectly types was an actual command that might stuff up what you’re doing?
  - Ask in lesson on Friday (turns out I needed to find out earlier see below)

6.1.2 Flexibility and automation

- Script = Sequences of commands
- Shell language:
  - enables us to combine tools to make pipelines to automate and handle large amounts of data
  - makes it easier to interact with remote and super computers
  - is necessary to interact with scientific data, clusters and cloud computing systems

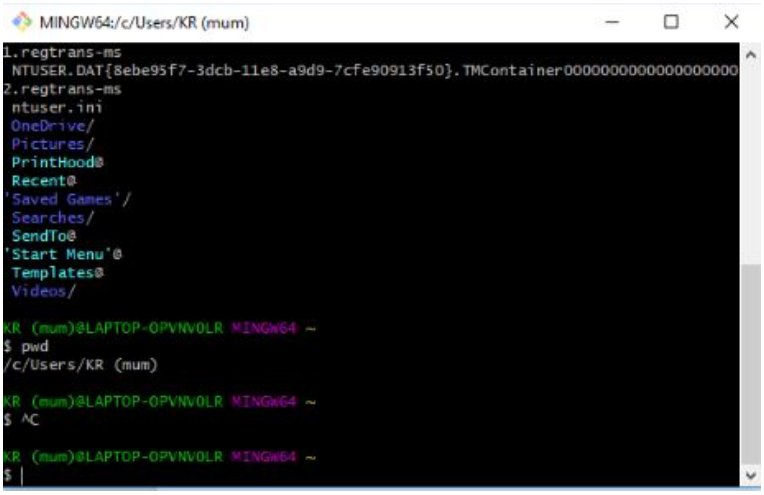
6.2 Navigating Files and Directories

28/08/2019 10:30pm ish

Objective: To successfully complete the Navigating files and directories module and to learn about files and directories, absolute paths and relative paths, how to navigate my computer using the shell, and tab completion.

- File system - the operating system responsible for managing the files and directories (AKA folders)
- Commands to manage file system:
  - create
  - inspect - pwd (print working directory) (where am I?) current working directory
  - rename
  - delete

- Crap! - tried to copy what the response was from Bash and it did a weird command and I am not sure what it means and if it will be bad for my computer - ahh help! see below:



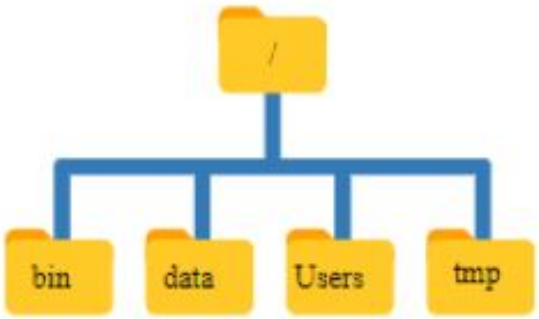
I am too scared to use my computer now that I have made some sort of command (Ĉ) on GitHub Bash, I don't know what I commanded it to do, so I have put a note up on Slack and Googled to see if I can find an answer to what I should do next. Just in case I need to do something to fix things. Will have to finish tomorrow instead. I shut down Bash - nothing seems to be going wrong at the moment - will wait and see tomorrow.

29/08/2019 10:26am

- Outcome of enquiry: Sheri commented on Slack that there doesn't seem to be an imminent problem and Brian confirmed this, this morning. Phew! Back to the exercise then.
- Resolution: Turns out Ctrl C (Ĉ) means clear which is great because I didn't stuff anything up and because I now know how to clear a command. we are going to discuss in class.
- Note to self: don't use GUI shortcuts (eg. Ctrl C to copy) in the Shell, the commands mean something different.

Lesson continued:

- root directory - holds everything else
- leading / command refers to the root directory
- /bin - root directory "/" + "bin" (other directories) - some built in programs are stored here
- /data - root directory "/" + "data" (misc. data files)
- /users - root directory "/" + "users - (user's personal directories)
- /tmp - temporary files



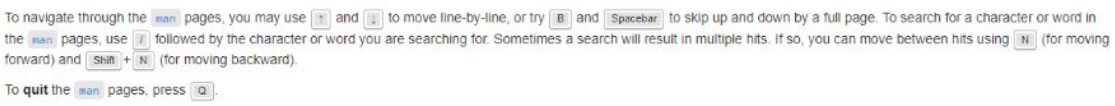
useful image to demonstrate above

- 2 meaning for /
  - leading / - root directory
  - if used inside a name it is a separator
- commands, options, arguments and parameters
  - ls - eg. of command - lists (prints) names of files in directory I am in -F - eg. of option [aka Switch or Flag],
    - starts with "-" or "- -" changes behavior of command -F adds a marker to output of file/directory to tell you what type of file it is:
      - / - directory
      - @ - link
      - * - executable
  - / - Arguments - tells command what to operate on (which file/directory)
  - Parameters - options/arguments sometimes referred to as parameters
  - Commands can have more than one option/argument
  - Command doesnt always require argument/option
  - each part of the parameter must have a space in between (ls -F)
  - Capitals are different from lowercase instructions
- I tried the / -F command in Bash and the marker showed up at the end of the file name(/ at end is subdirectory) - eg. Desktop/ and Cookies@
- no classification symbol means - plain files
- I tried typing in ls -F / and a list of files populated which was smaller than the "ls" only command

6.2.1 Getting Help

- ls - -help - more info on how to use help command
- man ls - (man = manual)
- I used the "ls – help" which worked in bringing up the list of options - VERY HANDY
- "man ls" - did not work for me
  - error message reads - "bash: man: command not found"

  Help page navigation - helpful image



- conducted a google search for "unix man page" and came up with a user guide - https://acadix.biz/Unix-guide/unix-guide.pdf which might come in handy

Exercise: Exploring ls flags:

- what happens when using ls -1
  - I am pretty sure that it did the same as ls -F - it listed the files/directories with the markers eg. /, @) at the end.
- what happens when using ls -h - it did the same thing as ls -1 and ls -F. I cant see a difference
- Problem - the solution says that they should have made the list human readable and showed file sizes but mine did not do that
- Action: follow-up Friday.

omg this is a long lesson - break time 29/08/2019 - 12:32pm

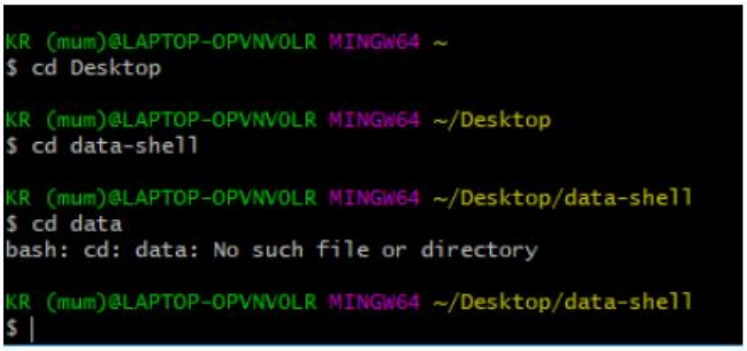
29/08/2019

Exercise: Listing Recursively and By Time

Question: In what order does ls -R -t display things?

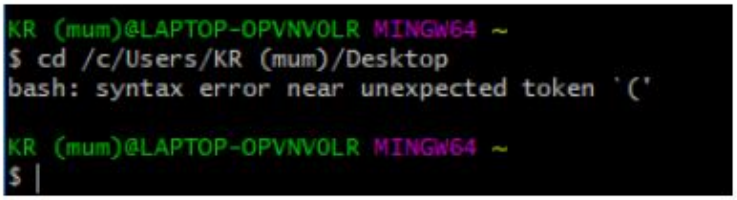
- I typed in ls -R at $ prompt and a whole bunch of data that means nothing to me started being listed, it is taking a very long time and I think it might be listing every single file on my laptop - it finished after about 3 minutes - the files where listed in Alphabetical order i think
- I typed in ls -t at $ prompt and a short list populated similar to when i typed in ls but in a different order - the Unix shell lesson says they are in time of last update order but I am unable to tell if that is true because there are no time stamps in the list
- $ ls F Desktop - this worked showing files on my desktop
- $ ls -F Desktop/data-shell - this did not show up anything for me (see below)
- why?

- cd - change directory
- i cant continue with this exercise because last one didnt work not sure if I should try it until i figure what happened in last step. wait I think it did work it is just there is only one file and doesn't look the example in lesson but that is probably because the sample probably had more files connected to data-shell file.
- $ cd Desktop - shows that we are in the Desktop (in yellow) directory on line above prompt
- $ cd data-shell - shows that we are in the Desktop/data-shell (in yellow) directory on line above prompt
- $ cd data - PROBLEM - says that no such file or directory exists (see image below)



- typed in pwd after prompt - shows that I am still in Desktop/data-shell directory
- Problem: why? this is different from what the lesson says I am going to go back to the Unix shell set up page to make sure I downloaded file properly.
- Solution: Instead of going back to set up page I first checked the file on my desktop, turns out I have an additional data-shell directory within my data-shell directory so I had to do an extra cd data-shell and then cd data
- ls -F - shows files as they are meant to be showing - SUCCESS!
- To go backwards up the directory tree type cd .. - Success apparently a special directory is missing (how would you know that if you didn't know what files are in the directory) - to get to special directory use show all -a

- $ ls -F -a - shows special hidden files (which begin with . or .. - "The prefix "." is used to prevent these configuration files from cluttering the terminal when a standard ls command is used".)
- Orthogonality -
- tried "cd" on its own and it took me back to my user directory (the home directory)
- to go to specific path (absolute path) enter in full directory name from root directory eg. /blah/blah2/blah3
- I entered pwd to find out where I was and it showed up as - /c/Users/KR (mum)/Desktop but it showed an error (see below)



- Problem: Not doing what lesson says it should - possible cause the brackets in my KR (mum) directory - does it not like brackets in directory names?? need to find out and change this somehow on my laptop files
- Solution: unsure for now. Slack??

6.2.2 Shortcuts

- (tilde) - at beginning of the path it means the current user's home directory
- cd - dash after cd takes you to back to last directory - different from cd .. which takes you up the directory tree.

Exercise: Absolute vs Relative Paths

Q1. Starting from /Users/amanda/data, which of the following commands could Amanda use to navigate to her home directory, which is /Users/amanda?

- Answer: cd .. CORRECT

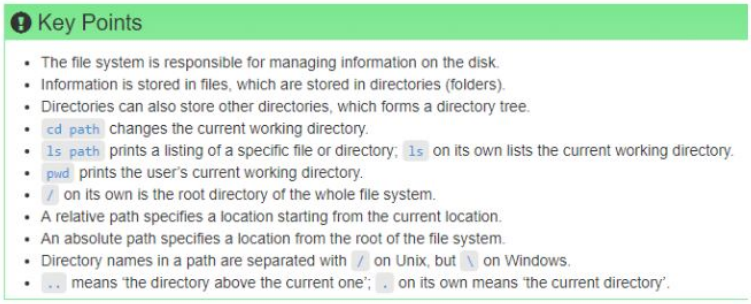
Q2. Using the file system diagram below, if pwd displays /Users/thing, what will ls - F ../backup display?

- Answer: ../backup: No such file or directory INCORRECT
- Correct answer: Yes: ../backup/ refers to /Users/backup/.

Q3. Using the filesystem diagram below, if pwd displays /Users/backup, and -r tells ls to display things in reverse order, what command(s) will result in the following output:

- pnas_sub/ pnas_final/ original/

- Answer: ls -r -F /Users/backup CORRECT
- tab completion - To auto populate file/directory names
- handy image to remember shortcuts and commands



Finish - 29/08/2019 2:57pm

6.3 Working with files and directories

01/09/2019 11:20am

Objective: To successfully complete the Working with files and directories module and to learn about how to create a directory hierarchy, files in that hierarchy using an editor or by copying and renaming existing files, and to delete, copy and move files and/or directories.

6.3.1 Creating directories (folders):

- see where we are
  - For this lesson I was instructed to go to the Data-shell directory on my desktop in shell (Bash)
  - I entered cd Desktop and Enter to change directories from user directory
  - I entered cd data-shell and Enter to go to Data-shell directory
  - type “ls -F” so I can see what is in the directory
- create a directory called thesis
  - mkdir (make directory) thesis (relative path - no / before so directory is located in current directory)
  - typed ls -F to check that directory was created
  - and thesis/ shows up in directory and checked windows file explorer and thesis folder is there also
  - Success!

- Note: Good naming conventions for files
  - NO spaces, NO dashes at beginning or end (saw end bit on a different site) because computer read - (dash) as a command for options.
  - DO use letters, numbers, dash (surrounded by other text), fullstops, and underscores (_).
  - if referring to filenames with spaces surround with quotation marks eg. "KR MUM" directory on my laptop.
- Create a text file in thesis directory
  - changed into thesis directory by typing cd thesis
  - typed in nano draft.txt as instructed - on windows can use notepad as well
  - had to try it so typed $\hat{\text{X}}$ as is instructed at the bottom of the window.
  - it just typed as text though - will have to ask in Slack if I dont find out in modules - nevermind just found out $\hat{=}$ Ctrl on keyboard
  - it populated what looks to be a window to type text (see below)

- I wrote out some text and pressed Ctrl and O as instructed and it populated a File name to write to option which was the draft.txt file that we created earlier
- I pressed Enter
- It populated a note at the bottom telling me that it had written two lines - that mean saved
- I pressed Ctrl X as instructed, however, it didnt send me back to the shell as it said that it would, instead it asked about a buffer, options were Yes No or Cancel.
- I said No and it took me back to the shell.
- I wrote ls to see if the file was in the directory and it was
- Success!

Exercise: Creating files in a different way

- typed command touch my_file.txt as instructed
- the files are listed with what looks like a character count
- Question: When might you want to create a file this way?
  - Answer: to see which files are the most recent and that have content ??
  - Solution: Some programs do not generate output files themselves, but instead require that empty files have already been generated. When the program is run, it searches for an existing file to populate with its output. The touch command allows you to efficiently generate a blank text file to be used by such programs.
- typed cd - before ending session - having a break

01/09/2019 8:37pm

6.3.2 What's in a name?

filename extension - eg. txt, pdf, png etc. - to help identify the type of file

6.3.3 Moving files and directories:

- went to data-shell direcoty by typing in cd /Desktop/data-shell
- to rename file mv thesis/draft.txt thesis/quotes.txt rename - mv means move but because it is in the same file it just renames
- I pressed enter and it went to prompt
- I typed in ls thesis and it listed the quotes.txt file
- however there was also another file in there my_file.txt also and I dont know where that came from??

6.3.4 Moving to the current folder

- I was going to attempt this exercise, however, I could not find any of these directory or file names in the directory we downloaded for these exercises so I am assuming that it is a test question of what to do.
- It asks how to move the files to another directory and to swap the file names over
- the question wasnt clear to me and the instructions were a little confusing so I went to the solution which is reminding me that you can use .. (parent directory) to go back a level in the directory heirarchy and . for the current directory. In the answer it says to move the files to the current directory which was the raw directory however in the solution it wants the files in the analyzed directory as different names
- I don't understand what it is trying to tell me to do - it is very unclear

6.3.5 Copying files and directories

- cp - copy file
  - this exercise doesn't give instructions about the directory we are supposed to be starting from - I am making an assumption that it is the one were in from the last exercise
  - this exercise did not work for me as the command "cp quotes.txt thesis/quotations.txt" did not work for me
  - Copying the directory to thesis_backup worked by following the commands suggested from the data-shell directory. the command was - $ cp -r thesis thesis_backup
  - I checkecd to see if the last step worked by typing $ ls thesis thesis_backup as suggested and this worked The file had been created

6.3.6 Renaming files exercise:

- Q: Suppose that you created a plain-text file in your current directory to contain a list of the statistical tests you will need to do to analyze your data, and named it: statstics.txt
- After creating and saving this file you realize you misspelled the filename! You want to correct the mistake, which of the following commands could you use to do so?
  - cp statstics.txt statistics.txt
  - mv statstics.txt statistics.txt
  - mv statstics.txt
  - cp statstics.txt
- A: 2 mv statstics.txt statistics.tx
- Result: Yes, this would work to rename the file. Success!

Exercise - Moving and copying

Answer is 4 - proteins-saved.dat bummer got it wrong it was the directory name recombine

6.3.7 Removing files and directories

- rm - to remove files
- I removed the quotes.txt from the directory by typing in rm quotes.txt
- I then tested to see if it was gone by listing the files in the directory by typing ls
- the quotes.txt file was gone
- Note: deleted files cannot be recovered - there is no trash/recycle bin

Exercise: What happens when we execute rm -i thesis_backup/quotations.txt? Why would we want this protection when using rm?

- Answer: It says no such file/directory exists

- this is not what was supposed to happen - it says that using -i is supposed to make it prompt you before going ahead and deleting.
- I am not sure why my file has disappeared.
- I made sure I was in the right directory which I wasn't
- I changed directory
- Did the command again and it worked
- I tried to delete the thesis file by typing rm thesis but it did not work
- I tried rm -r (recursive) thesis
- I typed ls to list files and it worked

6.3.8 Operations with multiple files and directories

Note: copy or move several files at once - done by providing a list of individual filenames, or specifying a naming pattern using wildcards. eg.

- I entered cd data to change directory
- I typed mkdir backup as directed to create a new directory
- It went to prompt which means everything went ok
- I typed cp amino-acids.txt animals.txt backup/
- the outcome was that two new files were created in the new backup directory

what does cp do when given 3+ files names?

- I typed ls -F as directed and the tow files as listed above showed up
- cp amino-acids.txt animals.txt morse.txt and pressed enter
- it came back with an error saying that morse.txt is not a directory

6.3.9 Using wildcards for accessing multiple files at once

- * - wildcard - matches 0 or more characters
- ? - wildcard - matches one character
- When the shell sees a wildcard, it expands the wildcard to create a list of matching filenames before running the command

List filenames matching a pattern:

- Answer: 3 correct
- Problem: I am starting to forget what some of the commands do
- Solution: I have started a new spreadsheet in my OneDrive so that I can keep track of the code and of what they do.

6.3.10 Organising Directories and Files

Exercise: move two files from one file to another

- Answer: mv *.dat analyzed/
- Solution: answer is correct!

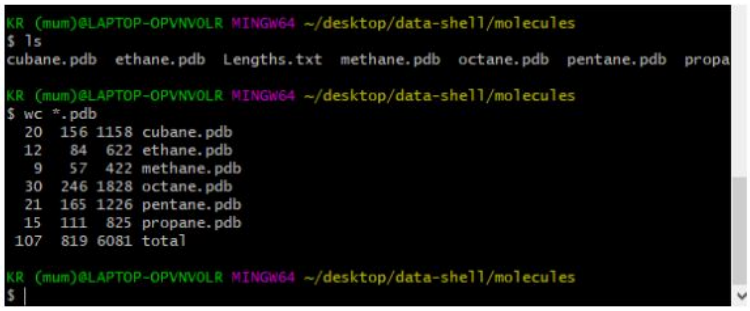
6.4 Pipes and Filters

Objective: to successfully complete the Unix Shell: Pipes and Filters mudule of this lesson. See here: http://swcarpentry.github.io/shell-novice/04-pipefilter/index.html

In this module I will learn to redirect a command's output to a file, process a file instead of keyboard input using redirection, construct command pipelines with two or more stages, explain what usually happens if a program or pipeline isn't given any input to process, explain Unix's 'small pieces, loosely joined' philosophy.

WC - Counts the number of lines, words, and characters in a file

- I went into the molecules file and typed wc *.pdb to see the word count of all the files with pdb as the file extension and it listed all of the files in the directory (they all ended with .pdb) with their counts as stated above.



- I typed wc -l as directed and it was supposed to list the line numbers for the files but this did not happen and it did not go to the prompt. I am not sure what happened.
- I realised that I had forgotten the file extension and pressed Ctrl C to escape as we were instructed in our last lesson at uni.
- Turns out this mistake is the next exercise so bonus I just saved some time. Yay!
- I will now do the correct command and list the items with their line numbers by typing
- wc -l *.pdb
- and success!
- replace wc -l with wc -w for number of words and wc -c for characters

- $ wc -l *.pdb ¿ lengths.txt - this comman creates a txt file in the molecules directory -
- after typing the command I checked to see if it had been created by typing ls to list the files and
- it was there.
- to see what is in the file I typed cat lengths.txt
- It showed the contents
- I typed this in and pressed enter and it showed me the page
- the lesson indicated that I could type b to go back but this did not work for me and it came up with the symbol instead - I am not sure why
- I pressed q for quit - this worked
- Reflection - I am assuming b did not work because there were no other pages

6.4.1 Sort -n

- Sort - sorts - n (numerically) doesn't work on its own have to specify file
- it shows up in numerical order
- to put sorted data in a file I typed sort -n lengths.txt ¿ sorted-lengths.txt
- I typed ls to see if it had worked
- then typed head -n 1 sorted-lengths.txt to see the first line in the file
- to see if changing the number showed 2 lines I typed in head -n 2 sorted-lengths.txt
- and it did show the top 2 lines in numerical order
- I will try 4 to see if it keeps working with higher numbers
- it did.

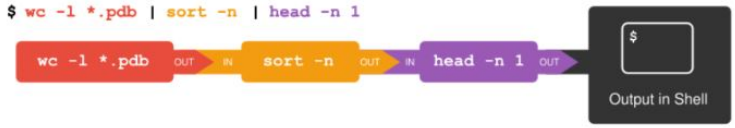
Note: do not redirect files to the same file name - could delete the original file

6.4.2 what does ¿¿ mean

- Echo - command - prints the text you type after it on the screen
- testing echo and opertors ¿ and ¿¿:
  - Echo hello ¿ testfile01.txt - it adds two hello's if you do the command twice
  - Echo hello ¿¿ testfile02.txt
  - the difference between the operators - one ¿ there was only one hello first hello is overwritten by the 2nd command - two ¿¿ there were two hellos in the file it adds more info, it appends the string to the file.

6.4.3 The pipeline

- shift + backslash = — (the pipe)
- The pipe filters through one command to the next (see image below)



- Piping commands together exercise - answer: 4 - Correct
- -d - is typed before the character which will be the delimiter (eg. -d , - the comma is now the delimiter, I think meaning whatever comes before it, is cut out of the equation) ?? not sure though
- uniq - filters out adjacent matching lines in a file

6.5 Loops

Objective: to successfully complete the Unix Shell: Loops module. See here: http://swcarpentry.github.io/shell-novice/05-loop/index.html
In this module I will learn to write a loop, learn about variables within a loop, identify executed commands, and re-run executed commands.

6.5.1 Follow the prompt

Loops are a "programming construct which allow us to repeat a command or set of commands for each item in a list"
Example of a loop:

```
for thing in list_of_things
do
    operation_using $thing    # Indentation within the loop is not required, but aids legibility
done
```

```
$ for filename in basilisk.dat minotaur.dat unicorn.dat
> do
>     head -n 2 $filename | tail -n 1
> done
```

```
CLASSIFICATION: basiliscus vulgaris
CLASSIFICATION: bos hominus
CLASSIFICATION: equus monoceros
```

- in creatures file to find the classification for each species (located on second line of each file) could execute head -n 2 — tail -n 1 for each individual file
- instead better to create a loop
- I typed the command as per image above and it printed the list below it successfully
- prompt $ changes to ¿ and back to $
- the command for means repeat command
- ; (semi-colon) can be used to separate commands on a single line
- the second $ marks the variable in the list - tells the shell interpreter to treat the variable as a variable name and substitute its value in its place, rather than treat it as text or an external command

6.5.2 Same symbols different meanings

- ¿ - overwrites OR can also be used as a prompt in a loop
- $ - shell prompt OR can also ask shell to get value of a variable
- if they show up on screen they are prompts
- if you type yourself they are instructions to ¿ redirect output or $ get value/variable
- $filename, $filename same, $filename different

Variables in Loops - Exercise

- $ for datafile in *.pdb
  ¿ do
  ¿ ls *.pdb
  ¿ done
  gives the same output on each loop - matches all files ending in .pdb and then lists (ls) them
- $ for datafile in *.pdb
  ¿ do
  ¿ ls $datafile
  ¿ done
  lists a different file on each loop. The value of the datafile variable ($datafile) is listed (ls)

Limiting Sets of Files - Exercise

- $ for filename in c*
  ¿ do
  ¿ ls $filename
  ¿ done Answer: 4 Only cubane.pdb is listed - Correct
- $ for filename in *c*
  ¿ do
  ¿ ls $filename
  ¿ done Answer: 4 The files cubane.pdb and octane.pdb will be listed - Correct

Saving to a File in a Loop - Part One - Exercise

- $ for alkanes in *.pdb
  ¿ do
  ¿ echo $alkanes
  ¿ cat $alkanes ¿ alkanes.pdb
  ¿ done
  Answer: none of the above - incorrect
  Correct answer: The text from each file in turn gets written to the alkanes.pdb file. However, the file gets overwritten on each loop, so the final content of alkanes.pdb is the text from the propane.pdb file.

Saving to a File in a Loop - Part Two - Exercise

- $ for datafile in *.pdb
  ¿ do
  ¿ cat $datafile ¿¿ all.pdb
  ¿ done
  Answer: 3 All of the text from cubane.pdb, ethane.pdb, methane.pdb, octane.pdb, pentane.pdb and propane.pdb would be concatenated and saved to a file called all.pdb - Correct

more complicated loop:

- $ for filename in *.dat
  ¿ do
  ¿ echo $filename
  ¿ head -n 100 $filename — tail -n 20
  ¿ done
- *.dat - all files with extension .dat listed for processing
- loop body - echo (print) $filename (prints the name of files)
- the head and tail part are a bit confusing - apparently it selects the 81-100 line of each file to print - not sure.

6.5.3 Spaces in Names

If an element eg. file name has a space in it we need to put quotation marks around it. eg. Kylie Reynolds.pdf would be "Kylie Reynolds.pdf" - this is because spaces are used to separate commands.

Try to avoid using spaces in filenames as it can cause problems'

```
$ for filename in "red dragon.dat" "purple unicorn.dat"
> do
>     head -n 100 "$filename" | tail -n 20
> done
```

- I removed the quotes around $filename in the loop above to see the effect - see below:

```
KR (mum)@LAPTOP-OPVNVOLR MINGW64 ~/desktop/data-shell/molecules
$ for filename in red dragon.dat purple unicorn.dat
> do
> head -n 100 $filename | tail -n 20
> done
head: cannot open 'red' for reading: No such file or directory
head: cannot open 'dragon.dat' for reading: No such file or directory
head: cannot open 'purple' for reading: No such file or directory
head: cannot open 'unicorn.dat' for reading: No such file or directory
```

Echo (prints) good for checking what is happening in the loop Can put whole loop on one line by separating with semi-colons (;) between the actions

Beginning and End

- Ctrl a - moves to beginning of line
- Ctrl e - moves to end

Those Who Know History Can Choose to Repeat It

- the history command to get a list of the last few hundred commands that have been executed - eg. - $ history — tail -n 5 - to look up last five commands
- to repeat a command you can type in !(and number of the command line on the list)

Other History Commands

- Ctrl R - 'reverse-i-search' - history search mode - most recent command in your history that matches the text you enter next
- !! - retrieves the last command entered - preceding command
- !! retrieves the immediately preceding command
- less !$ to look at the file

Doing a Dry Run - Exercise

- to preview the commands - Echo
- Answer: Version 1 - incorrect - version 2

There are two more modules to go to finish this unit. I have been unable to complete them in time for handing in of Learning Journal for Unix Shell submission. I will be catching up on these modules this week. Each module is taking me hours to get through and understand. I may need to catch up on break with all of my work.
Placemarkers for modules to be done are below:

Key Points

- A `for` loop repeats commands once for every thing in a list.
- Every `for` loop needs a variable to refer to the thing it is currently operating on.
- Use `$name` to expand a variable (i.e., get its value). `${name}` can also be used.
- Do not use spaces, quotes, or wildcard characters such as '*' or '?' in filenames, as it complicates variable expansion.
- Give files consistent names that are easy to match with wildcard patterns to make it easy to select them for looping.
- Use the up-arrow key to scroll up through previous commands to edit and repeat them.
- Use `Ctrl-R` to search through the previously entered commands.
- Use `history` to display recent commands, and `!number` to repeat a command by number.

6.6 Shell Scripts

To do.

6.7 Finding Things

To do.