Exercise 1: Exploration of the data

- Exercise 1A: How many subdirectories are present within the directory called "E_coli"?
  2 subdirectories.
  Either cd into E_coli and use ls (1), or use ls directly (2):
  1) cd E_coli
     ls
  2) ls E_coli/

- Exercise 1B: In which subdirectory can the file "E_coli_CP127297.fasta" be found?
  E_coli/assemblies

- Exercise 1C: How many files are in the "Virus" subdirectory?
  If you are in a subfolder, use cd .. to go back to a main folder
  3

- Exercise 1D: What is the full path for the file named "R1_E_coli_02.fq"?
  readlink -f R2_E_coli_02.fq (to find the answer)
  /home/gebt/BTG/Day1/BacterialData/E_coli/reads/R2_E_coli_02.fq

Exercise 2: Cleaning it up!

- Exercise 2A: Move the assembly file "E_coli_WWcol315.fasta" into the "assemblies" sub-directory within the "E_coli" directory.
  In BacterialData:
  mv E_coli_WWcol315.fasta E_coli/assemblies

- Exercise 2B: Move the read files "R1_E_coli_01.fq" and "R2_E_coli_01.fq" into the "reads" sub-directory within the "E_coli" directory.
  There are two ways, with or without wildcard
  In BacterialData:

  1)
  mv R1_E_coli_01.fq E_coli/reads
  mv R2_E_coli_01.fq E_coli/reads

2)
mv R*_E* E_coli/reads/

- Exercise 2C: In the "BacteriaData" directory, create a subdirectory named "P_aeruginosa".
  mkdir P_aeruginosa

- Exercise 2D: In the "P_aeruginosa" directory, make a subdirectory called "reads".
  If you're in BacterialData:
  mkdir P_aeruginosa/reads

  If you're in P_aeruginosa:
  mkdir reads

- Exercise 2E: In the "P_aeruginosa" directory, make another subdirectory called "assemblies".
  mkdir assemblies

- Exercise 2F: Move all read files into P_aeruginosa/reads using only a single command.
  The only read files that are left are those from P_aeruginosa. We can move them all using a wildcard.
  mv *.fq P_aeruginosa/reads

- Exercise 2G: Rename "P_oeruginosa_PPF1.fasta" to "P_aeruginosa_PPF1.fasta"
  mv P_oeruginosa_PPF1.fasta P_aeruginosa_PPF1.fasta

- Exercise 2H: Move all assembly files into P_aeruginosa/assemblies. Try to do it with a single command.
  mv *.fasta P_aeruginosa/assemblies

Exercise 3: Removing redundant content

- Exercise 3A: Remove the "Credit_cards.txt" file from "BacteriaData" (nothing to see here, we promise, delete!).
  rm Credit_cards.txt
- Exercise 3B: Remove the "Virus" directory and everything in it. As the folder insinuates, we are only working with bacteria.
  rm -rf Virus/

Exercise 4: Inspection

- Exercise 4A: Inspect the assembly file "P_aeruginosa_TOprJ3-positive_part1.fasta" using the **cat** command.
  cat P_aeruginosa_TOprJ3-positive_part1.fasta

- Exercise 4B: Inspect the assembly file "P_aeruginosa_TOprJ3-positive_part2.fasta" using the **less** command. Observe the difference between **cat** and **less**. Note: press "q" to exit **less**.
  less P_aeruginosa_TOprJ3-positive_part2.fasta

- Exercise 4C: When would you use cat and when would you use less? Discuss with your partner
  Cat for smaller files, less for larger. It will clutter the terminal a lot less.

- Exercise 4D: Get the first three lines of "P_aeruginosa_TOprJ3-positive_part2.fasta" using the **head** command.
  head -n 3 P_aeruginosa_TOprJ3-positive_part2.fasta

- Exercise 4E: Get the last five lines of "P_aeruginosa_TOprJ3-positive_part2.fasta" using the **tail** command.
  tail -n 5 P_aeruginosa_TOprJ3-positive_part2.fasta

Exercise 5: Combining files

- Exercise 5A: Using **cat,** concatenate the two files called "P_aeruginosa_TOprJ3-positive_**part1**.fasta" and "P_aeruginosa_TOprJ3-positive_**part2**.fasta" into a new file called "P_aeruginosa_TOprJ3-positive.fasta" in the P_aeruginosa/assembly

folder.
cat P_aeruginosa_TOprJ3-positive_part1.fasta P_aeruginosa_TOprJ3-positive_part2.fasta > P_aeruginosa_TOprJ3-positive.fasta

- Exercise 5B: Using **cat**, append the file "P_aeruginosa_TOprJ3-positive_**part3**.fasta" to the newly created "P_aeruginosa_TOprJ3-positive.fasta".
  cat P_aeruginosa_TOprJ3-positive_part3.fasta >> P_aeruginosa_TOprJ3-positive.fasta

Exercise 6: Searching within files

- Exercise 2A: Use **grep** to extract any line containing "terA".
  grep "terA" P_aeruginosa_TOprJ3-positive.fasta
- Exercise 2B: Count the total number of sequences; keep in mind that FASTA headers begin with the ">" symbol. Hint: remember proper quotations around ">".
  grep -c ">" P_aeruginosa_TOprJ3-positive.fasta
  106 sequences

Exercise 7: Word count and piping results between programs

- Exercise 3A: Begin by counting the number of lines in "P_aeruginosa_TOprJ3-positive.fasta".
  wc -l P_aeruginosa_TOprJ3-positive.fasta
  1674 lines
- Exercise 3B: Determine how many words are in the "README.md" file.
  wc -w README.md
  131 words
- Exercise 3C: Combine the **ls** and **wc** commands using a pipe to count the files in the P_aeruginosa/assemblies directory.
  ls P_aeruginosa/assemblies/ | wc -l
  8 files

Extra exercises

Extra Exercises 1: More fun with grep!

- Extra 1A: Use **grep** to save the header names in a file called "header_names.txt". View the file using **less**.
  grep ">" P_aeruginosa_TOprJ3-positive.fasta > header_names.txt
- Extra 1B: Let's discover what happens when you don't use proper quotations with **grep**. Run the following command:
  grep > header_names.txt

  Then display the content of header_names.txt using less.
  What happened and why? Discuss with your partner.

  Using the command above, you save the empty grep result into the file "header_names.txt".
  As > is not in quotations, the command is understood as saving the result and overwriting "header_names.txt", instead of looking for '>' in the file.
  Remember you overwrite using only one >, and >> would be appending.

  This is why it is recommended always to use quotation marks.
  If it's a habit, you won't accidentally empty a file.

  Don't leave trash! Delete the file after use:
  rm header_names.txt

Extra exercise 2: Options to ls!

- Extra 2A: What is the largest file in P_aeruginosa/assemblies?
  ls -lh
  P_aeruginosa_PPF1.fasta

Extra Exercises 3: Save space – use symbolic links

- Extra 3A: Make a new subdirectory called "Resfinder" inside "BacteriaData".

- Extra 3B: Copy the "E_coli_ASM584v2_reference.fasta" assembly file from E_coli/assemblies and place the copy in the "Resfinder" directory.
  cp E_coli/assemblies/E_coli_ASM584v2_reference.fasta Resfinder

- Extra 3C: Create a symbolic link to "E_coli_ASM584v2_reference.fasta" from the E_coli/assemblies directory in the "Resfinder" directory.
  ln -s cannot give the same name to a file already used in the same folder.
  An idea would be to name the symbolic link something different or delete the prior file.
  We will delete the prior file here, as keeping the same name will prevent future confusion.

  rm E_coli_ASM584v2_reference.fasta
  ln -s E_coli/assemblies/E_coli_ASM584v2_reference.fasta Resfinder/

- Extra 3D: Create symbolic links to all assembly files in the "E_coli" and "P_aeruginosa" directories within the "Resfinder" directory.
  ln -sr E_coli/assemblies/* Resfinder/
  ln -sr P_aeruginosa/assemblies/* Resfinder/

  You will get an error that says "E_coli_ASM584v2_reference.fasta: File exists"
  There is no worry, as we already have a symbolic link to that file,
  and the rest will still be granted symbolic links

- Extra 3E: List the contents of the "Resfinder" directory and observe the visual differences between symbolic links and actual file copies.
  ls -h should show a vast difference in size and colors

Extra Exercises 4: Files for a Colleague – Compressing files

- Extra 4A: Compress the "Resfinder" directory using the **tar** command and name the resulting gzipped file "ResFinder.tar.gz". Using the tar command, ensure that tar zips the real files through the symbolic links, not just the symbolic links themselves. Make sure you're not inside the "Resfinder" directory when you do this.
  tar -czvhf ResFinder.tar.gz Resfinder/

- Extra 4B: Retrieve the path for the "ResFinder.tar.gz" file so you can share it with your colleague.
  readlink -f ResFinder.tar.gz