

# **Exercises: Introduction to Unix**

## Licence

This manual is © 2018-2021, Simon Andrews.

This manual is distributed under the creative commons Attribution-Non-Commercial-Share Alike 2.0 licence. This means that you are free:

- to copy, distribute, display, and perform the work
- to make derivative works

Under the following conditions:

- Attribution. You must give the original author credit.
- Non-Commercial. You may not use this work for commercial purposes.
- Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under a licence identical to this one.

Please note that:

- For any reuse or distribution, you must make clear to others the licence terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.
- Nothing in this license impairs or restricts the author's moral rights.

Full details of this licence can be found at

<http://creativecommons.org/licenses/by-nc-sa/2.0/uk/legalcode>



- Using `cp` copy `Mito.dat` into the `compare` directory in your home directory
  - Use `nano` to edit the file
  - Change `Mito` to `Mitochondrion` in the `ID` and `AC` header lines
  - Save the file and exit `nano`
  - Using `mv` rename the file from `Mito.dat` to `Mitochondrion.txt`
- Using `ln -s` create a symlink from the original `Mito.dat` file (the one at `~/seqmonk_genomes/Saccharomyces\ cerevisiae/EF4/Mito.dat`) to the same filename in your `compare` directory
- Run `diff` on `Mitochondrion.txt` and `Mito.dat` to see what differences it can find

## Exercise 4: Redirection and Bash Loops

- Go into the `FastQ_Data` directory and look at one of the fastq files using `less`
  - `Less` is clever enough to realise that the file needs to be decompressed so you can just pass the file to `less` directly
  - Now validate that one of the files can be successfully decompressed
    - Run `zcat` on the file, but...
    - Throw away the `STDOUT` output (using `> /dev/null`) so that you just see errors or warnings
- Calculate the signatures of all of the fastq files using the `shasum` program
  - Start by running `shasum` on one fastq file to see how it works
  - Now run it on the entire contents of `FastQ_Data` using a wildcard (rather than a loop)
    - Write the results (`STDOUT`) to a file in your home directory using `>~/signatures.txt`
    - Write any errors to a different file in your home directory (`2>~/errors.txt`)
- Use `nohup` to run the `fastqc` program on all of the fastq.gz files (`*fastq.gz`)
  - Check the `nohup.out` file to see that it has finished.
- Once the `fastqc` jobs have finished, run `multiqc .` (note the dot to specify it should run in the current directory) to assemble the `fastqc` output into a single report.

## If you have time

- Write a bash loop which will go through every `.dat` file in `seqmonk_genomes` and will count the number of lines containing `rrna` (case insensitive). The process will be:
  - Move to the `seqmonk_genomes/Saccharomyces cerevisiae` folder
  - Use a shell wildcard which will find all of the `.dat` files (`*/*.dat`)
  - Write a loop to iterate over these. For each one
    - Use `echo` to write out the name of the file plus a space (check for how to not include a newline at the end)
    - Use `grep` to get the lines containing "rrna" (check for case insensitive)
    - Use `wc` to get and print the number of lines of hit (check how to just get the line count)
    - Run the loop and save the results to a file called `rrna_count.txt`

- Convert every fastq.gz file in `FastQ_Data` into a `fastq.bz2` file
  - Read the file with `zcat`
  - Pipe it to `bzip2` (with the option to write to stdout)
  - Redirect the output to a new file with `.bz2` on the end