**Data Science Specialization – Coursera**

**The Data Scientist’s Toolbox -** 02-22-2016

**Command Line Interface:** 02-26-2016

* ‘Root directory’ represented by a slash: /
* ‘Home directory’ represented by a tilde: ~
* To navigate directories, you need a command line interface (CLI)
  + In this case, using Git Bash (<https://git-scm.com/downloads>)
* ‘Working directory’ represents whatever directory you are currently in
* CLI Commands:
  + Command flags arguments
    - Flags are options given to command. Preceded by a –
    - Arguments can be what the command is going to modify, or other options
    - Some commands have no flags or arguments
  + **pwd**: prints path to the working directory
  + **ls**: lists all files and folders in working directory
    - **ls –a**: lists hidden and unhidden files and folders
    - **ls –al**: lists details for hidden and unhidden files and folders
      * Note that **–a** and **–l** are flags. They can be combined into the flag **–al**
  + **cd**:changes the working directory.
    - Takes as an argument the directory you want to visit
    - cd Chemi-Doc
    - **cd ..**: changes to directory on level up from working directory
    - Use “ “ to indicate a file name with spaces included
  + **mkdir**: makes a new directory
    - Takes as an argument the name of the new directory you’re creating
    - Use “ “ to make a file name with spaces included
  + **touch**:Creates an empty file
    - Takes as an argument the name of the file
  + **cp**: Copies a file to a new location
    - Takes file name as first argument, path of new directory as second
    - Can also be used for copying contents of directories using the flag -r
  + **rm**: Remove a file or directory
    - Takes the name of the file you wish to remove as its argument
    - Cannot be undone!
    - Can also be used for removing entire directories using the flag –r
  + **mv**: Move a file between directories
    - Takes the file name as the first argument and the destination as the second
    - Can also be used to rename a file (name of file is first argument, new name is second)
  + **echo**: prints whatever argument you provide
  + **date**: prints the date

**cd Users/MA25302/Documents/R**

MA25302@2UA41115B0 MINGW64 /c/Users/MA25302/Documents/R

**#See a list of all files/folders in a directory**

**ls**

Agilent ID conversion/ GEO/ NCOR1\_prostate/ R-3.1.0/ VDR ChIP/

bimiR/ GO analysis/ Pipeline/ RARG/

CODES/ Microarray/ R-2.15.1/ RStudio/

Coursera/ MSKCC/ R-3.0.0/ TCGA/

ENCODE/ NCI60/ R-3.0.3/ TMA/

**#Change the current directory**

**cd Coursera**

MA25302@2UA41115B0 MINGW64 /c/Users/MA25302/Documents/R/Coursera

**#See the current directory**

**pwd**

/c/Users/MA25302/Documents/R/Coursera

**#Move up one directory**

**cd ..**

**pwd**

/c/Users/MA25302/Documents/R

**cd Coursera**

**#Create a new directory**

**mkdir “Data Science Specialization\_022216”**

**cd “Data Science Specialization\_022216”**

**pwd**

/c/Users/MA25302/Documents/R/Coursera/Data Science Specialization\_022216

**#Create an empty file**

**touch test\_file**

**ls**

test\_file

**#Copy a file to a new location**

**cd ..**

**touch test\_file2**

**cp test\_file2 “Data Science Specialization\_022216”**

**cd “Data Science Specialization\_022216”**

**ls**

test\_file test\_file2

**#Copy a directory to a new location**

**cd ..**

**mkdir Copy\_Directory**

**cp –r “Data Science Specialization\_022216” Copy\_Directory**

**cd Copy\_Directory**

**ls**

Data Science Specialization\_022216/

**touch test\_file3**

**cp test\_file3 /c/Users/MA25302/Documents/R/Coursera**

**cd ..**

**ls**

**#Move a file to a new location**

**mv test\_file3 “Data Science Specialization\_022216”**

**cd “Data Science Specialization\_022216”**

**ls**

test\_file test\_file2 test\_file3

**#Rename a file**

**mv test\_file3 renamed\_file3**

**ls**

test\_file test\_file2 renamed\_file3

**#Print an argument**

**echo Hello World!**

Hello world

**#Print the date**

**date**

Fri, Feb 26, 2016 11:27:59 AM

**Introduction to Git:** 02-26-2016

* **What is Version Control (or revision control)?**
  + A system that records changes to a file or set of files over time so that you can recall specific versions later
  + A means to managing the process of creating something, saving it, changing it, then saving it again in a reliable and efficient way
  + Important when collaborating with others
* **What is Git?**
  + A free and open source distributed version control system
  + Designed to handle everything from small to very large projects with speed and efficiency
  + Created by same people who developed Linux
  + Most commonly used implementation of version control today
  + Operated from the Command Line
  + Can download at <https://git-scm.com/downloads>
* **What is Git Bash**
  + The command line environment for interacting with Git
* **Configure Username and Email**
  + Each commit to a Git repository will be “tagged” with the username of the person who made the commit
  + Enter the following commands in Git Bash to set your username and email

**git config –global user.name mark14226**

**git config –global user.email** [**mark14226@gmail.com**](mailto:mark14226@gmail.com)

**git config –list**

$ git config --list

core.symlinks=false

core.autocrlf=true

color.diff=auto

color.status=auto

color.branch=auto

color.interactive=true

help.format=html

http.sslcainfo=C:/Users/MA25302/AppData/Local/Programs/Git/mingw64/ssl/certs/ca-bundle.crt

diff.astextplain.textconv=astextplain

rebase.autosquash=true

user.name=mark14226

user.email=mark14226@gmail.com

**#Close Git Bash**

**exit**

**Introduction to GitHub:** 02-26-2016

* **What is GitHub?**
  + Web-based hosting service for software development projects that use the Giy revision control system https://github.com/.
  + When setting up an account, use the same username and password used to configure Git
  + Repositories are created containing everything required for a project (files, folder, images, data sets, spreadsheets, readme files, etc)
  + These repositories can be shared with others, and you can access others repositories
  + Remote copies of your repositories are stored on GitHub’s server in case something happens to your local copies
  + Git = Local (on your computer); GitHub = Remote (on the web)
* **Creating a GitHub Repository (aka “Repo”)**
  + Method 1: Starting a repository from scratch
  + Method 2: “Fork” another user’s repository