# Project 1

## Semester One 2023



This project draws on lecture and lab content from Module 1: Creating HTML by combining modern technologies. The knowledge and skills assessed by the project will be covered in lectures/labs by **the end of Module 1**. Consequently, the order of the project instructions below does not necessarily reflect the order the material is covered in lectures/labs.

# Prep work

- You will also need access to a computer that has **R (https://cran.r-project.org/)** installed and **RStudio (https://www.rstudio.com/products/rstudio/download/)** installed.
- **If installing R and RStudio onto your computer is a problem, you can use the free level of Posit Cloud (https://posit.cloud/)**
- Install the R package {tidyverse} first (this will take a while)
- Then install the R package {magick}

*Remember, you can use the lab sessions to get help with completing your project, and with any technical difficulties you face, like installing R or RStudio, or any packages.*

# Part A: Setting up a GitHub account and repo

- If you do not already have a **GitHub (https://github.com)** account, you will need to set up a free account for use during this course (and beyond!). **Think wisely about what to name your account - remember future employers might look at it!**
- In your GitHub account, create a new repository called “stats220”. Choose the option to create a *README.md* file at the same time as creating your new repository.

- Edit the *README.md* file so that it contains some information about the purpose of the repo, so that other people who view your repo on GitHub know why you made it. Your *README.md* file needs to demonstrate *at least* the following Markdown syntax (you can do more if you want!):
    - use of two different levels of headers
    - use of two different types of bullet points (ordered or unordered)
    - use of **bold** and *italics*
    - use of links to other websites

*Remember, that your repo is public, so only write and share content that you are comfortable with others viewing. You can **view the "demo" repo Anna set up on GitHub (https://github.com/annafergusson/stats220_demo)** for some ideas to get started, although you should not copy this exactly!*

# Part B: Meme creation

- Create a folder **on your computer (or within RStudio Cloud)** called stats220. Within this folder, create a new project using RStudio that is called "Project1"
    - Create all R script files and Rmd files in this folder, and save any data and image files (e.g. plots, animated GIFs) to this folder
- Find an existing meme online that you will recreate for this part of the project
    - Analyse the design of the meme and identify the key components of the image you will need to create
- Create a new R file within your *Project1* project called "meme.R".
    - Edit the first line of the file so that it reads `library(magick)`
- Develop R code within the *meme.R* file to create a new/original meme. Your R code needs to demonstrate the following:
    - use of *at least* the following functions: `c()`, `image_blank()`, `image_read()`, `image_annotate()`, `image_append()`
    - use of pipes e.g. `%>%`
    - use of named objects e.g. `meme <- image_read(url)`
    - use of comments e.g. `# note to self`
    - use of indenting and "white space"
- Include R code that saves your meme as an image file (e.g. *.png*), using the `image_write()` function.
    - For example, to save the {magick} object called `meme` as an image file called *my_meme.png*, use the code `image_write(meme, "my_meme.png")` or the code `meme %>% image_write("my_meme.png")`

# Part C: Animated GIF creation

- Create a new R file within your *Project1* project called "animation.R".
  - Edit the first line of the file so that it reads `library(magick)`
- Develop R code within the *animation.R* file to create a new/original animated GIF. Your R code needs to demonstrate the following:
  - use of at least the `image_animate()` function
  - creation of at least 4 different frames for the animation using a combination of different functions from the {magick} package
  - use of named objects e.g. `meme <- image_read(url)`
  - use of comments e.g. `# note to self`
  - use of indenting and "white space"
- Include code that saves your animated GIF as an image file (e.g. *.gif*), using the `image_write()` function.
  - For example, to save the {magick} object called `animation` as an image file called *my_animation.gif*, use the code `image_write(animation, "my_animation.gif")` or the code `animation %>% image_write("my_animation.gif")`

# Part D: HTML creation

- Create a new Rmd file within the *Project1* project called "index.Rmd", with the title "Creating images with R" and with your name as the author. **Note: RStudio will automatically install the packages needed to write and knit .Rmd files the first time you use RMarkdown.**
- Edit the YAML of the *index.Rmd* file so that the subtitle says "with a little bit of magick".
- Edit the `knitr::opts_chunk$set()` so that messages are not displayed but keep `echo = TRUE` so that your code can be viewed in the web page.
- Structure the rest of the page using **second-level** headings as follows:
  - My meme
  - My animated GIF
- Under the *My meme* section of your page:
  - Write one sentence that introduces your meme and provides some context for its creation (e.g. what the motivation was and how your meme is new/original)
  - **use markdown** to include your meme i.e. use `![]()`
  - copy the code chunk below to include the code you used to create your meme (you do not need to add your code to it, this will be

done automatically when the Rmd is knitted)

```
```{r file='meme.R', eval=FALSE}

```
```

- Under the *My animated GIF* section of your page:
    - Write one sentence that introduces your animated GIF and provides some context for its creation (e.g. what the motivation was and how your meme is new/original)
    - **use markdown** to include your animated GIF i.e. use `![]()`
    - copy the code chunk below to include the code you used to create your meme (you do not need to add your code to it, this will be done automatically when the Rmd is knitted)

```
```{r file='animation.R', eval=FALSE}

```
```

- Add at least one CSS chunk to your *index.Rmd* file that changes the visual appearance of your report in at least two different ways.
- Knit your *index.Rmd* file to create a self-contained *index.html* file. **Check the *index.html* carefully that all code used, included CSS code, is visible otherwise you will lose marks.**

# Part E: Setting up Github Pages

- Go to your *stats220* repo on Github
    - Add a new empty file called ".nojekyll" to the root of your *stats220* repository (repo)
    - Upload your `index.html` file into your *stats220* repo (as your HTML file is 'self-contained' your images have been saved within the file)
- Set up GitHub Pages for your *stats220* repository (repo). **Note this is not the same as setting up an account, this is an additional step**.
    - Record the URL (web address) for your published stats220 website.
- Click on the URL to check it works (it will take a few minutes for the website to become "alive"!)

# Part F: Writing the project report

*The main purposes of the project report are provide you with an opportunity to: (1) demonstrate or explain how you have met some of the requirements for the project, and, (2) reflect on your learning for this project and across Module 1. Make sure to refer to the marking guide at the end of this page when completing your project and writing this report.*

- Create a new Rmd file within the *Project1* project called "report.Rmd"
- Create a second level heading called *Project requirements*
    - Briefly summarise how you have met the requirements related to working with GitHub and GitHub pages. Include the links to your repo and to your website (these are different links).
    - Include a screenshot of the project folder you have created on your computer. It should be obvious in the screenshot that you have set up and used `.Rproject` within RStudio, and all files and images used for this project should be visible.
    - Describe and justify how your project demonstrates creativity.

*What does "demonstrate creativity" mean? It means that you have gone beyond what was asked in terms of either your explanations, creations, presentation or use of data technologies. For this assignment, that could mean using additional functions from the {magick} package that were not covered in the lectures/labs, using more CSS than what was required to change the appearance of your HTML, adding more content to your GitHub website (e.g. creating another webpage using Rmd that is linked to from your index page), or incorporating additional context when creating the meme or animated GIF.* ***You can check with Anna or the lab tutors to confirm your plans for creativity are sufficient!***

- Create a second level heading called *Learning reflection*
    - Describe in your own words at least ONE important idea you learned from Module 1 *Creating HTML by combining modern technologies*.
    - Discuss what things related to data technologies that you are more curious about exploring further.
    - *Write at least 100 words for this section*
- Knit your *report.Rmd* file to create a self-contained *report.html* file.

# Marking guide

*For this assignment, you will submit the following files:*

- *index.html*
- *report.html*

**The assignment will be marked out of 10.** The criteria given below are based on the six learning objectives of Module 1 and the three focuses for this course:

| | Focus or objective | Criteria |
|---|---|---|
| | Develop creativity with data and technology | The project demonstrates creativity (1 mark) |
| | Use Markdown to create HTML for web pages | The Markdown syntax used meets the stated requirements (1 mark) |
| | Use R Markdown to structure and create web pages | The visual appearance of the HTML created has been modified in at least two different ways using CSS code (1 mark) |
| | Develop communication with data and technology | The project report meets the stated requirements (1 mark) |
| | Share work using GitHub & GitHub Pages | The GitHub repo contains the required files and GitHub Pages has been set up (1 mark) |
| | Create and use projects in RStudio | A RStudio project folder was created and used for the project (1 mark) |
| | Develop R-coding related knowledge | The R code features use of comments, indenting, and “white space” |
| | Identify key syntax features of R code | A new *meme* has been created with R code that meets the stated requirements (1 mark) |
| | Re-use and combine R functions to create a desired product | A new *animated GIF* has been created with R code that meets the stated requirements (1 mark) |
| | Submission requirements | All correct files were submitted for the project (1 mark) |