Problem Set 2

YOUR NAME HERE

DATE

- Due date: Monday, September 14
- Submission process: Please submit your assignment directly to Gradescope. You can do this by knitting your file and downloading the PDF to your computer. Then navigate to Gradescope.com or via the link on BCourses to submit your assignment.

Helpful hints:

- Knit your file early and often to minimize knitting errors! If you copy and paste code from the slides, you are bound to get an error that is hard to diagnose. Typing out the code is the way to smooth knitting. We recommend knitting your file each time after you write a few sentences/add a new code chunk, so yu can detect the source of the knitting error more easily. This will save you and the teaching team time!
- Please make sure that your code does not run off the page of the knitted PDF. If it does, we can't see your work. To avoid this, have a look at your knitted PDF and ensure all the code fits in the file. When it doesn't, go back to your .Rmd file and add spaces (new lines) using hte return or enter key so that the code runs onto the next line.

Question 1

Create a data frame and a tibble that matches the image below:

```
# by the way, you can load images into rmarkdown! Cool, right?!
# here we use the knitr library (though there are multiple ways to load images)
library(knitr)

# notice that we specify the path to look within the current directory
# by using the period: .
# followed by a slash: / to pull the image file
knitr::include_graphics('./table_replicate.png')
```

data_id	gender	temperature
101	female	98
102	male	97.3
103	non-binary	101.1
104	male	97.5
105	NA	99.6

Hint: You may need to load a library for tibbles.

```
# your code here
```

What are the key differences between data frames and tibbles?

Why are tibbles preferable?

Question 2

We just found out results for COVID testing and want to add it to our data. Using the tibble you just created, add the following test results to a new column called "results".

- 101 = NEGATIVE
- 102 = POSITIVE
- 103 = NEGATIVE
- 104 = NEGATIVE
- 105 = NEGATIVE

```
# your code here
```

Question 3

You find out there was an error in data collection and subject 102's temperature is actually 98.3, not 97.3. Correct the value in your data frame.

```
# your code here
```

Question 4

Load the "stds-by-disease-county-year-sex.csv" data set, which is in the data folder.

You can find more information about this data set from the California Open Data Portal:

https://data.ca.gov/dataset/stds-in-california-by-disease-county-year-and-sex

```
library(readr)

# your code here
```

You may have noticed that there are empty cells in the first three rows. Modify your code above (if you haven't already) to remove these rows.

Question 5

Let's explore this data set. Insert R chunks as needed. Find the following values:

```
# your code here
```

How many rows?

How many columns?

What are the column names?

What are the column types?

Question 6

You want to dig deeper into the data and focus on the years 2015 - 2018. Use the which() function to index which rows fit this year range and assign the results to a new data frame. To check whether this was done correctly you should expect the following dimensions: 2124 rows x 6 columns

```
# your code here
```

Question 7

Your colleague is interested in this data set but hasn't setup their git repository. They ask you to help them out by exporting this new data set as a .csv file. Place your output in the /data folder.

As a test, you can try to read in the .csv you created to make sure everything looks correct.

```
# your code here
```

Challenge

Look up how to use the unique() function and run it on the County column. You should see a total of 59 counties.

```
# your code here
```

You decide to focus on one county. Subset your data for one of your choice.

```
# your code here
```

You're very interested in finding the rate of cases per 100,000 population. Create a new column called "rate" with the calculated values.

Rate = (Cases / Population) * 100,000

Hint: R allows you to use manipulate variables within a data frame to calculate new values so long as the rows and data types match up. For example: df$var3 <- df$var1 + df$var2

```
# your code here
```

You're done! Please knit to pdf and upload to gradescope.