Problem Set #1

INSERT YOUR NAME HERE

Invalid Date

Part 1 (Week 1)

Welcome to Data Cleaning and Management using R! This problem set is intended to give you some practice becoming familiar with using R. In part 1, I'm asking you to: create an R project; render to pdf; load and investigate an R data frame that is stored on the web; and apply some basic functions to atomic vectors.

- Note: Change the values of the YAML header above to your name and the date.

Question 1: Creating an R project

Create an R project

- Create a folder where you want to save files associated with problem set 1. Let's call that folder "problemset1", but you can name it whatever you want.
  - For instance, it could be psc290-fq23 » problem-sets » problemset1.
- In RStudio, click on "File" » "New Project" » "Existing Directory" » "Browse".
- Browse to find and select your problem set 1 folder.
- Click on "Create Project".
  - An R project file has the extension ".Rproj".
  - The name of the file should be "problemset1.Rproj", or whatever you named the folder.

Save this problemset1.Rmd file anywhere in the folder named problemset1.

- Use this naming convention "lastname-firstname-ps#" for your .qmd files (e.g. beck_emorie_ps1.qmd).
  - If you want, you can change the name of this file to include your first and last name.
- Run the `getwd()` function and the `list.files()` function in the code chunk below.

- What is the output? Why?

ANSWER:

Question 2: Knit to pdf

- At the top of this .qmd file, type in your first and last name in the appropriate place in the YAML header (e.g. "Hadley Wickham").
- In the date field of the YAML header, insert the date within quotations (any date format is fine).
- Now click the "Render" button near the top of your RStudio window (icon with blue yarn ball).
  - Alternatively you can use the shortcut: **Cmd/Ctrl + Shift + k**.
  - *Note*: One goal of this assignment is to make sure you are able to render to a PDF without running into errors.

Question 3: Load .Rdata directly with url and then investigate the data frame

1. This question asks you to load a dataframe by specifying the `url()` function within the `load()` function.

- Url link for data frame: https://github.com/emoriebeck/psc290-data-FQ23/raw/main/05-assignments/01-ps1/pwe-ps1-small.RData
  - Hint: to load .Rdata use the `load()` and `url()` functions because you are using a link. follow this approach: `load(url("url_link"))`.
    - Note: the `url_link` is put within quotes

Load the dataframe within this code chunk below.

```
#?load
```

2. Print the data frame `df_pwe` by typing its name.

3. Use the `typeof()` function to investigate the type of data frame `df_pwe`.

4. Apply the `length()` function to the data frame `df_pwe`. What does this output mean in your own words?

ANSWER:

5. Use the `str()` function to investigate the structure of the data frame `df_pwe`.

6. Use the `names` function to list the names of the elements (variables) within `df_pwe`.

7. Wrap your answer above — `names(data_frame_name)` — within the `typeof()` function. Do the same for the `length()` function, and the `str()` function as well. Interpret what the output means in your own words.

ANSWER:

Question 4: Applying basic functions to atomic vectors

1. Create an atomic vector object named age with the following values: 3, 6, 41, 43.

2. Apply the `typeof()`, `length()`, and `str()` functions to the object `age`.

3. Apply the `sum()` function to `age`.

4. Apply the `sum()` function to `age` but this time include the argument `na.rm = FALSE`.

5. In general, what is a function "argument name" and what is an "argument value"? What does the argument `na.rm` do?

ANSWER:

6. Create a new object `age2` with the following values: 3, 6, 41, 43, NA. Now calculate the sum of `age2` using the argument `na.rm = FALSE` and then calculate the sum using the argument `na.rm = TRUE`. Explain why the outputs of these two `sum()` functions differ.

ANSWER:

7. Create a vector `tf` using the following code: `tf <- c(TRUE,FALSE,TRUE,FALSE,TRUE)`. Next apply the `typeof()`, `length()`, and `str()` functions to the object `tf`. Based on this output, briefly describe the object `tf` in your own words (one sentence is fine).

ANSWER:

8. Apply the `sum()` function to the object, using the option to remove `NA` values prior to calculation. What numeric value do mathematical calculations in `R` assign to `TRUE` values and what do they assign to `FALSE` values?

ANSWER:

9. This is the syntax of the `mean()` function that includes both argument names and the default values for arguments: `mean(x, trim = 0, na.rm = FALSE)`.

When using a function, `R` requires you to type the values you assign to each argument, but typing in the argument names is usually optional. Even though it takes a bit more time, I usually like typing in both argument names and argument values, because it forces me to be more conscious about what value I am assigning to which argument, especially when a function is new to me.

Use the `mean()` function to calculate the mean of object `tf` (removing `NA` values prior to calculation). In your function call, include both the argument name and the argument value for each argument (argument value for the `trim` argument can be `0`). Then run the same function, but without typing any argument names.

Part 2 (Week 2)

The aim of part 2 is to give you practice completing data management tasks associated with filtering/isolating observations, sorting observations, and selecting variables. This can be done using the `filter()`, `arrange()`, and `select()` functions from the `tidyverse` package. Filtering/sorting the data can also be done using `base R`'s subsetting operators and `subset()`/`order()` functions (not covered in class but examples provided below; **note: just do you best with these that we didn't discuss in class!**).

For the following questions, you'll be asked to complete the same task multiple ways based on the `tidyverse` and `base R` approaches. We want you to understand that there are several ways to complete the same task and we want you to practice completing the same task in different ways.

Question 1: Load and inspect `df_event` dataset

1. In the code chunk below, complete the following:

    - Load the `tidyverse` library
    - Use the `load()` and `url()` functions to download the `df_event` dataframe from the url: `https://github.com/emoriebeck/psc290-data-FQ23/raw/main/05-assignments/02-ps2/ps`
        - Each row in `df_event` represents a recruiting visit

```
rm(list = ls())
```

2. Inspect the `df_event` dataframe:

    - Use `names()` to identify the column names in the dataframe
    - Use `typeof()` to show the data type of the `event_state` column
    - Use `str()` to show the structure of the `med_inc` column
    - Use `table()` to show the categorical values of the `event_type` column

Question 2: Filtering/isolating observations

Filtering can be done using multiple approaches: `tidyverse`'s `filter()` function, `base R`'s subsetting operators, and `base R`'s `subset()` function. Here is an example of using each method to obtain the total number of recruiting visits to California from the `df_event` dataframe:

```
# tidyverse using filter()
nrow(filter(df_event, event_state == 'CA'))

# base R using subsetting operators
nrow(df_event[df_event$event_state == 'CA', ])

# base R using subset()
nrow(subset(df_event, event_state == 'CA'))
```

1. Your turn! Count the number of recruiting events that satisfy all the following criteria:

   - By the University of Massachusetts-Amherst (`univ_id`: `166629`)
   - An out-of-state public high school (use `event_type`, `event_state`, and `instst`, which is the visiting university's home state)
   - Average median household income is greater than or equal to $100,000 (`med_inc`)
   - Make sure to drop any `NA` values

   Use `nrow()` to obtain the count. Do the filtering in the 3 ways below. You should get the same answer.

tidyverse using `filter()`:

base R using subsetting operators (hint: use `which()` to drop `NA`s):

base R using `subset()`:

2. Count the number of recruiting events that satisfy all the following criteria:

   - By the University of South Carolina-Columbia (`univ_id`: `218663`) or by the University of Alabama (`univ_id`: `100751`)
   - And either:
     - An in-state 2-year college visit (use `event_type`, `event_state`, and `instst`, which is the visiting university's home state) OR
     - A zip code with population under 10,000 (use `pop_total`)
   - Make sure to drop any `NA` values
   - Note the order of precedence: `&` is higher in priority than `|`

tidyverse using `filter()`:

base R using subsetting operators (hint: use `which()` to drop `NA`s):

base R using `subset()`:

Question 3: Sorting observations

1. Create a new dataframe that contains the events in `df_events` sorted by:

   - Ascending `univ_id`
   - Ascending `event_date`
   - Ascending `event_state`
   - Descending `pct_white_zip`
   - Descending `med_inc`

   Then preview the first 10 rows using `head()`. Do this in 2 ways: using `tidyverse`'s `arrange()` and `base R`'s `order()`.

tidyverse using `arrange()`:

base R using `order()`:

Question 4: Selecting variables

1. Create a new dataframe by selecting the columns `univ_id`, `event_date`, `event_type`, `zip`, and `med_inc` from `df_event`. Use the `names()` function to show what columns (variables) are in the newly created dataframe.

   Do this in 3 ways: using `tidyverse`'s `select()`, `base R`'s subsetting operators, and `base R`'s `subset()`. (Note for the latter 2, do your best attempt since you may not be familiar with it)

tidyverse using `select()`:

base R using subsetting operators:

base R using `subset()`:

Question 5: Additional practice with `df_school_all` dataframe

1. In the code chunk below, complete the following:

   - Use the `load()` and `url()` functions to download the `df_school_all` dataframe from the url: `https://github.com/emoriebeck/psc290-data-FQ23/raw/main/05-assignments/0`
     - Each row in `df_school_all` represents a high school (includes both public and private)
     - There are columns (e.g., `visit_by_100751`) indicating the number of times a university visited that high school
     - The variable `total_visits` identifies the number of visits the high school received from all (16) public research universities in this data collection sample
   - Use `names()` to identify the column names in the dataframe
   - Use `table()` to show the categorical values of the `school_type` column

2. Use the tidyverse functions `arrange()` and `select()` to do the following:

   - Sort `df_school_all` descending by `total_visits`
   - Select the following variables: `name`, `state_code`, `city`, `school_type`,`total_visits`, `med_inc`, `pct_white`, `pct_black`, `pct_hispanic`, `pct_asian`, `pct_amerindian`
     - Note: You can do this in one step by wrapping the `select()` function around `arrange()`, or you can do this in two steps by creating an intermediate dataframe.

   Print the first 10 rows of the final dataframe using `head()`, which represents the top 10 most visited schools by the 16 universities.

3. Building upon the previous question, print the following (select same variables as above):

   (A) Top 10 most visited public high schools in California
   (B) Top 10 most visited private high schoools in California

Render to pdf and submit problem set

Render to pdf by clicking the "Render" button near the top of your RStudio window (icon with blue yarn ball) or drop down and select "Knit to PDF".

- Go to the Canvas and under the "Assignments", submit to the Problem Set 1 Assignment.
- Submit both .qmd and .pdf files.
- Use this naming convention "lastname_firstname_ps#" for your .qmd and pdf files (e.g. beck-emorie-ps1.rmd & beck-emorie-ps1.pdf).