Problem Set #3

Nicole Wood (in group NJ with Jade Levandofsky and Jessika Viveros)

October 14, 2023

The aim of this problem set 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 the data can also be done using `base R`'s subsetting operators.

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.

Note: For those who know how to use the `%>%` (pipe) operator, please **do not** use it for this assignment. We will go over pipe operators and how to use them later on in the quarter.

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/anyone-can-cook/rclass1/raw/master/data/recruiting/recruit_event_somevars.Rl`
        - Each row in `df_event` represents a recruiting visit

```
r = getOption("repos")
r["CRAN"] = "http://cran.us.r-project.org"
options(repos = r)
rm(list = ls())
R.Version()
#> $platform
#> [1] "x86_64-w64-mingw32"
#>
#> $arch
#> [1] "x86_64"
#>
#> $os
#> [1] "mingw32"
#>
#> $crt
#> [1] "ucrt"
#>
#> $system
#> [1] "x86_64, mingw32"
#>
```

```
#> $status
#> [1] ""
#>
#> $major
#> [1] "4"
#>
#> $minor
#> [1] "3.1"
#>
#> $year
#> [1] "2023"
#>
#> $month
#> [1] "06"
#>
#> $day
#> [1] "16"
#>
#> $`svn rev`
#> [1] "84548"
#>
#> $language
#> [1] "R"
#>
#> $version.string
#> [1] "R version 4.3.1 (2023-06-16 ucrt)"
#>
#> $nickname
#> [1] "Beagle Scouts"
install.packages("tinytex")
#> Installing package into 'C:/Users/there/AppData/Local/R/win-library/4.3'
#> (as 'lib' is unspecified)
#> package 'tinytex' successfully unpacked and MD5 sums checked
#>
#> The downloaded binary packages are in
#>  C:\Users\there\AppData\Local\Temp\RtmpOOJkDQ\downloaded_packages
install.packages("tidyverse")
#> Installing package into 'C:/Users/there/AppData/Local/R/win-library/4.3'
#> (as 'lib' is unspecified)
#> package 'tidyverse' successfully unpacked and MD5 sums checked
#>
#> The downloaded binary packages are in
#>  C:\Users\there\AppData\Local\Temp\RtmpOOJkDQ\downloaded_packages
install.packages("Rtools")
#> Installing package into 'C:/Users/there/AppData/Local/R/win-library/4.3'
#> (as 'lib' is unspecified)
#> Warning: package 'Rtools' is not available for this version of R
#>
#> A version of this package for your version of R might be available elsewhere,
#> see the ideas at
#> https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
library(tidyverse)
#> -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
```

```
#> v dplyr     1.1.3     v readr     2.1.4
#> v forcats   1.0.0     v stringr   1.5.0
#> v ggplot2   3.4.4     v tibble    3.2.1
#> v lubridate 1.9.3     v tidyr     1.3.0
#> v purrr     1.0.2
#> -- Conflicts ---------------------------------------- tidyverse_conflicts() --
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
#> i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
load(url("https://github.com/anyone-can-cook/rclass1/raw/master/data/recruiting/recruit_event_somevars.
str(df_event)
#> tibble [18,680 x 32] (S3: tbl_df/tbl/data.frame)
#>  $ instnm              : chr [1:18680] "UM Amherst" "UM Amherst" "UM Amherst" "UM Amherst" ...
#>  $ univ_id             : int [1:18680] 166629 166629 166629 166629 196097 218663 166629 166629 16662
#>  $ instst              : chr [1:18680] "MA" "MA" "MA" "MA" ...
#>  $ pid                 : int [1:18680] 57570 56984 57105 57118 16281 8608 56898 56933 56940 57030 ..
#>  $ event_date          : Date[1:18680], format: "2017-10-12" "2017-10-04" ...
#>  $ event_type          : chr [1:18680] "public hs" "public hs" "public hs" "public hs" ...
#>  $ zip                 : chr [1:18680] "01002" "01007" "01020" "01020" ...
#>  $ school_id           : chr [1:18680] "250192000042" "250243000134" "250366000496" "250366000495" .
#>  $ ipeds_id            : int [1:18680] NA NA NA NA NA NA NA NA NA NA ...
#>  $ event_state         : chr [1:18680] "MA" "MA" "MA" "MA" ...
#>  $ med_inc             : num [1:18680] 71714 89122 70137 70137 71024 ...
#>  $ pop_total           : num [1:18680] 29970 14888 30629 30629 17872 ...
#>  $ pct_white_zip       : num [1:18680] 73.7 91.4 79.4 79.4 88.7 ...
#>  $ pct_black_zip       : num [1:18680] 5.27 0.84 3.03 3.03 1.76 ...
#>  $ pct_asian_zip       : num [1:18680] 11.69 2.98 1.26 1.26 1.34 ...
#>  $ pct_hispanic_zip    : num [1:18680] 6.24 2.05 14.64 14.64 6.59 ...
#>  $ pct_amerindian_zip  : num [1:18680] 0.2469 0 0.1339 0.1339 0.0168 ...
#>  $ pct_nativehawaii_zip: num [1:18680] 0.0567 0 0 0 0 0 0 0 0 0 ...
#>  $ pct_tworaces_zip    : num [1:18680] 2.59 2.55 1.51 1.51 1.64 ...
#>  $ pct_otherrace_zip   : num [1:18680] 0.24 0.228 0 0 0 ...
#>  $ fr_lunch            : num [1:18680] 269 120 632 857 152 NA NA 117 NA 35 ...
#>  $ titlei_status_pub   : Factor w/ 7 levels "1","2","3","4",..: 6 6 6 6 6 NA NA 6 NA 6 ...
#>  $ total_12            : num [1:18680] NA NA NA NA NA 136 136 NA 71 NA ...
#>  $ school_type_pri     : int [1:18680] NA NA NA NA NA 1 1 NA 1 NA ...
#>  $ school_type_pub     : int [1:18680] 1 1 1 1 1 NA NA 1 NA 1 ...
#>  $ g12offered          : num [1:18680] 1 1 1 1 1 NA NA 1 NA 1 ...
#>  $ g12                 : num [1:18680] 263 172 249 329 117 NA NA 77 NA 41 ...
#>  $ total_students_pub  : num [1:18680] 980 708 1035 1453 451 ...
#>  $ total_students_pri  : num [1:18680] NA NA NA NA NA 509 509 NA 297 NA ...
#>  $ event_name          : chr [1:18680] "Amherst-Pelham Regional High School Visit" "Hampshire County
#>  $ event_location_name : chr [1:18680] "Amherst-Pelham Regional High School" "Belchertown High Schoo
#>  $ event_datetime_start: POSIXct[1:18680], format: "2017-10-12 11:30:00" "2017-10-04 18:00:00" ...
```

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

```r
names(df_event)
#>  [1] "instnm"               "univ_id"              "instst"
#>  [4] "pid"                  "event_date"           "event_type"
#>  [7] "zip"                  "school_id"            "ipeds_id"
#> [10] "event_state"          "med_inc"              "pop_total"
#> [13] "pct_white_zip"        "pct_black_zip"        "pct_asian_zip"
#> [16] "pct_hispanic_zip"     "pct_amerindian_zip"   "pct_nativehawaii_zip"
#> [19] "pct_tworaces_zip"     "pct_otherrace_zip"    "fr_lunch"
#> [22] "titlei_status_pub"    "total_12"             "school_type_pri"
#> [25] "school_type_pub"      "g12offered"           "g12"
#> [28] "total_students_pub"   "total_students_pri"   "event_name"
#> [31] "event_location_name"  "event_datetime_start"
typeof(df_event$event_state)
#> [1] "character"
str(df_event$med_inc)
#>  num [1:18680] 71714 89122 70137 70137 71024 ...
table(df_event$event_type)
#>
#> 2yr college 4yr college       other  private hs   public hs
#>         951         531        2001        3774       11423
```

Question 2: Filtering/isolating observations

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

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

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

1. Your turn now! 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 2 ways below. You should get the same answer.

tidyverse using `filter()`:

```r
nrow(filter(df_event, univ_id == "166629"))
#> [1] 908
nrow(filter(df_event, event_type == "public hs", event_state != instst))
#> [1] 7896
nrow(filter(df_event, med_inc >= 100000))
#> [1] 6346
```

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

```
nrow(df_event[which(df_event$univ_id == "166629"),])
#> [1] 908
nrow(df_event[which(df_event$event_type == "public hs" & df_event$event_state != df_event$instst),])
#> [1] 7896
nrow(df_event[which(df_event$med_inc >= 100000),])
#> [1] 6346
```

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()`:

```
nrow(filter(df_event, univ_id == "218663" | univ_id == "100751"))
#> [1] 5725
nrow(filter(df_event, univ_id == "218663" | univ_id == "100751", event_type == "2yr college", event_sta
#> [1] 68
nrow(filter(df_event, univ_id == "218663" | univ_id == "100751", pop_total < 10000))
#> [1] 484
nrow(filter(df_event, univ_id == "218663" | univ_id == "100751", ((event_type == "2yr college" & event_s
#> [1] 543

#9 events are for in-state 2 year college visits AND a zip code with a total population under 10,000 (r
```

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

```
nrow(df_event[which(df_event$univ_id == "218663" | df_event$univ_id == "100751"),])
#> [1] 5725
nrow(df_event[which((df_event$univ_id == "218663" | df_event$univ_id == "100751") & df_event$event_type
#> [1] 68
nrow(df_event[which((df_event$univ_id == "218663" | df_event$univ_id == "100751") & df_event$pop_total
#> [1] 484
nrow(df_event[which((df_event$univ_id == "218663" | df_event$univ_id == "100751") & ((df_event$event_ty
#> [1] 543
```

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 using `tidyverse`'s `arrange()` function.

tidyverse using `arrange()`:

```
df_event <- arrange(df_event, univ_id, event_date, event_state, desc(pct_white_zip), desc(med_inc))
head(df_event, n = 10)
#> # A tibble: 10 x 32
#>    instnm univ_id instst   pid event_date event_type  zip   school_id   ipeds_id
#>    <chr>    <int> <chr>  <int> <date>     <chr>       <chr> <chr>          <int>
#>  1 Bama    100751 AL      2667 2017-01-10 private hs  75001 X1328481          NA
#>  2 Bama    100751 AL      2674 2017-01-11 2yr college 35010 <NA>           100760
#>  3 Bama    100751 AL      2675 2017-01-11 other       35044 <NA>               NA
#>  4 Bama    100751 AL      2691 2017-01-12 private hs  75244 A0303150          NA
#>  5 Bama    100751 AL      2676 2017-01-17 2yr college 36350 <NA>           101286
#>  6 Bama    100751 AL      2851 2017-01-17 public hs   21769 2400330006~       NA
#>  7 Bama    100751 AL      2733 2017-01-17 public hs   75002 4807890001~       NA
#>  8 Bama    100751 AL      2677 2017-01-18 2yr college 36330 <NA>           101143
#>  9 Bama    100751 AL      2645 2017-01-18 public hs   30277 1301500020~       NA
#> 10 Bama    100751 AL      2736 2017-01-18 public hs   30281 1302820012~       NA
#> # i 23 more variables: event_state <chr>, med_inc <dbl>, pop_total <dbl>,
#> #   pct_white_zip <dbl>, pct_black_zip <dbl>, pct_asian_zip <dbl>,
#> #   pct_hispanic_zip <dbl>, pct_amerindian_zip <dbl>,
#> #   pct_nativehawaii_zip <dbl>, pct_tworaces_zip <dbl>,
#> #   pct_otherrace_zip <dbl>, fr_lunch <dbl>, titlei_status_pub <fct>,
#> #   total_12 <dbl>, school_type_pri <int>, school_type_pub <int>,
#> #   g12offered <dbl>, g12 <dbl>, total_students_pub <dbl>, ...
```

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 2 ways: using `tidyverse`'s `select()` and `base R`'s subsetting operators.

tidyverse using `select()`:

```
df_event_tidy <- select(df_event, univ_id, event_date, event_type, zip, med_inc)
names(df_event_tidy)
#> [1] "univ_id"    "event_date" "event_type" "zip"        "med_inc"
```

base R using subsetting operators:

```
df_event_R <- df_event[c(2,5,6,7,11)]
names(df_event_R)
#> [1] "univ_id"    "event_date" "event_type" "zip"        "med_inc"
```

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/anyone-can-cook/rclass1/raw/master/data/recruiting/recruit_school_allvars.R`
     - 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

```
load(url("https://github.com/anyone-can-cook/rclass1/raw/master/data/recruiting/recruit_school_allvars.
str(df_school_all)
#> tibble [21,301 x 55] (S3: tbl_df/tbl/data.frame)
#>  $ state_code      : chr [1:21301] "AK" "AK" "AK" "AK" ...
#>  $ school_type     : chr [1:21301] "public" "public" "public" "public" ...
#>  $ ncessch         : chr [1:21301] "020000100208" "020000100211" "020000100212" "020000100213" ...
#>  $ name            : chr [1:21301] "Bethel Regional High School" "Ayagina'ar Elitnaurvik" "Kwigillin
#>  $ address         : chr [1:21301] "1006 Ron Edwards Memorial Dr" "106 Village Road" "108 Village Ro
#>  $ city            : chr [1:21301] "Bethel" "Kongiganak" "Kwigillingok" "Toksook Bay" ...
#>  $ zip_code        : chr [1:21301] "99559" "99559" "99622" "99637" ...
#>  $ pct_white       : num [1:21301] 11.78 0 0 0 2.52 ...
#>  $ pct_black       : num [1:21301] 0.599 0 0 0 0 ...
#>  $ pct_hispanic    : num [1:21301] 1.6 0 0 0 0 ...
#>  $ pct_asian       : num [1:21301] 0.998 0 0 0 0 ...
#>  $ pct_amerindian  : num [1:21301] 84.6 99.5 100 100 97.5 ...
#>  $ pct_other       : num [1:21301] 0.399 0.549 0 0 0 ...
#>  $ num_fr_lunch    : num [1:21301] 362 182 116 187 238 180 418 185 179 186 ...
#>  $ total_students  : num [1:21301] 501 182 120 201 238 231 428 262 179 186 ...
#>  $ num_took_math   : num [1:21301] 146 17 14 30 28 25 62 21 23 19 ...
#>  $ num_prof_math   : num [1:21301] 24.8 1.7 3.5 3 2.8 ...
#>  $ num_took_rla    : num [1:21301] 147 17 14 30 28 24 62 22 23 19 ...
#>  $ num_prof_rla    : num [1:21301] 25 1.7 3.5 3 2.8 ...
#>  $ med_inc         : num [1:21301] 76160 76160 NA 57657 37553 ...
#>  $ latitude        : num [1:21301] 60.8 60 59.9 60.5 62.7 ...
#>  $ longitude       : num [1:21301] -162 -163 -163 -165 -165 ...
#>  $ visits_by_196097: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_186380: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_215293: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_201885: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_181464: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_139959: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_218663: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_100751: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_199193: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_110635: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_110653: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_126614: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_155317: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_106397: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
```

```
#>  $ visits_by_149222: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ visits_by_166629: int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ total_visits    : int [1:21301] 0 0 0 0 0 0 0 0 0 0 ...
#>  $ inst_196097     : chr [1:21301] "NY" "NY" "NY" "NY" ...
#>  $ inst_186380     : chr [1:21301] "NJ" "NJ" "NJ" "NJ" ...
#>  $ inst_215293     : chr [1:21301] "PA" "PA" "PA" "PA" ...
#>  $ inst_201885     : chr [1:21301] "OH" "OH" "OH" "OH" ...
#>  $ inst_181464     : chr [1:21301] "NE" "NE" "NE" "NE" ...
#>  $ inst_139959     : chr [1:21301] "GA" "GA" "GA" "GA" ...
#>  $ inst_218663     : chr [1:21301] "SC" "SC" "SC" "SC" ...
#>  $ inst_100751     : chr [1:21301] "AL" "AL" "AL" "AL" ...
#>  $ inst_199193     : chr [1:21301] "NC" "NC" "NC" "NC" ...
#>  $ inst_110635     : chr [1:21301] "CA" "CA" "CA" "CA" ...
#>  $ inst_110653     : chr [1:21301] "CA" "CA" "CA" "CA" ...
#>  $ inst_126614     : chr [1:21301] "CO" "CO" "CO" "CO" ...
#>  $ inst_155317     : chr [1:21301] "KS" "KS" "KS" "KS" ...
#>  $ inst_106397     : chr [1:21301] "AR" "AR" "AR" "AR" ...
#>  $ inst_149222     : chr [1:21301] "IL" "IL" "IL" "IL" ...
#>  $ inst_166629     : chr [1:21301] "MA" "MA" "MA" "MA" ...
names(df_school_all)
#>  [1] "state_code"       "school_type"      "ncessch"          "name"
#>  [5] "address"          "city"             "zip_code"         "pct_white"
#>  [9] "pct_black"        "pct_hispanic"     "pct_asian"        "pct_amerindian"
#> [13] "pct_other"        "num_fr_lunch"     "total_students"   "num_took_math"
#> [17] "num_prof_math"    "num_took_rla"     "num_prof_rla"     "med_inc"
#> [21] "latitude"         "longitude"        "visits_by_196097" "visits_by_186380"
#> [25] "visits_by_215293" "visits_by_201885" "visits_by_181464" "visits_by_139959"
#> [29] "visits_by_218663" "visits_by_100751" "visits_by_199193" "visits_by_110635"
#> [33] "visits_by_110653" "visits_by_126614" "visits_by_155317" "visits_by_106397"
#> [37] "visits_by_149222" "visits_by_166629" "total_visits"     "inst_196097"
#> [41] "inst_186380"      "inst_215293"      "inst_201885"      "inst_181464"
#> [45] "inst_139959"      "inst_218663"      "inst_100751"      "inst_199193"
#> [49] "inst_110635"      "inst_110653"      "inst_126614"      "inst_155317"
#> [53] "inst_106397"      "inst_149222"      "inst_166629"
table(df_school_all$school_type)
#>
#> private  public
#>    3822   17479
```

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.

```
df_school_all <- select(arrange(df_school_all, desc(total_visits)), name, state_code, city, school_type
head(df_school_all, n = 10)
#> # A tibble: 10 x 11
```

```
#>    name    state_code city  school_type total_visits med_inc pct_white pct_black
#>    <chr>   <chr>      <chr> <chr>              <int>   <dbl>     <dbl>     <dbl>
#>  1 EPISCO~ VA         ALEX~ private               26 109558.     77.8      12.1
#>  2 Lyons ~ IL         La G~ public                23  94306.     74.1       3.71
#>  3 ALLEN ~ TX         ALLEN public                23 100809      57.2      11.8
#>  4 COPPEL~ TX         COPP~ public                23 123382.     49.9       4.97
#>  5 FLOWER~ TX         FLOW~ public                22 157234.     74         3.06
#>  6 NOLAN ~ TX         FORT~ private               21  39490.     55.8       3.47
#>  7 FORT W~ TX         FORT~ private               20  89470.      4.09      2.82
#>  8 LOVEJO~ TX         LUCAS public                19 100809      81.9       1.91
#>  9 STRAKE~ TX         HOUS~ private               18  29630.     56.7       7.76
#> 10 TRINIT~ TX         ADDI~ private               18  77380      83.5       1.60
#> # i 3 more variables: pct_hispanic <dbl>, pct_asian <dbl>, pct_amerindian <dbl>
```

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

```
#the arrangement desc(total_visits)) was assigned to the dataframe df_school_all when completing questi

filter(df_school_all, school_type == "public", state_code == "CA") %>% head(df_school_all, n = 10)
#> # A tibble: 10 x 11
#>    name    state_code city  school_type total_visits med_inc pct_white pct_black
#>    <chr>   <chr>      <chr> <chr>              <int>   <dbl>     <dbl>     <dbl>
#>  1 Corona~ CA         Newp~ public                12 133966       82.6     0.900
#>  2 Trabuc~ CA         Miss~ public                12 112446.      57.2     1.69
#>  3 Monte ~ CA         Danv~ public                10 168605       67.9     0.931
#>  4 Santa ~ CA         Sant~ public                10  93942       41.4     9.28
#>  5 Tustin~ CA         Tust~ public                10  70780.      13.3     3.26
#>  6 Calaba~ CA         Cala~ public                 9 123449       78.7     4.17
#>  7 Palos ~ CA         Palo~ public                 9 211304.      69.5     2.28
#>  8 Mira C~ CA         Manh~ public                 8 168271       58.8     5.36
#>  9 Burrou~ CA         Burb~ public                 8  87288       37.2     2.34
#> 10 Aliso ~ CA         Alis~ public                 8 110660.      59.2     1.28
#> # i 3 more variables: pct_hispanic <dbl>, pct_asian <dbl>, pct_amerindian <dbl>
filter(df_school_all, school_type == "private", state_code == "CA") %>% head(df_school_all, n = 10)
#> # A tibble: 10 x 11
#>    name    state_code city  school_type total_visits med_inc pct_white pct_black
#>    <chr>   <chr>      <chr> <chr>              <int>   <dbl>     <dbl>     <dbl>
#>  1 SANTA ~ CA         RANC~ private               15 105576.      66.6      1.27
#>  2 JSERRA~ CA         SAN ~ private               14  88324       60.1      1.93
#>  3 MATER ~ CA         SANT~ private               12  64052.      38.3      3.72
#>  4 SERVIT~ CA         ANAH~ private               11  55142       41.0      4.53
#>  5 ST FRA~ CA         LA C~ private                9 177146.      48.0      1.66
#>  6 CHAMIN~ CA         WEST~ private                8  64568.      49.1      6.77
#>  7 NOTRE ~ CA         SHER~ private                8  91428.      62.6      7.40
#>  8 JUNIPE~ CA         SAN ~ private                8 123328       61.7      2.97
#>  9 CATHED~ CA         SAN ~ private                8 143160       87.1      2.12
#> 10 ST IGN~ CA         SAN ~ private                6 121018.      60.1      3.18
#> # i 3 more variables: pct_hispanic <dbl>, pct_asian <dbl>, pct_amerindian <dbl>
```

Create a GitHub issue

- Go to the class repository and create a new issue.
- Refer to rclass1 student issues readme for instructions on how to post questions or reflections.
- You are also required to respond to at least one issue posted by another student.
- Paste the url to your issue here: https://github.com/anyone-can-cook/rclass1_student_issues_f23/issues/399
- Paste the url to the issue you responded to here: https://github.com/anyone-can-cook/rclass1_student_issues_f23/issues/398

Knit to pdf and submit problem set

getwd() setwd("C:/Users/there/OneDrive/Documents)

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

- Go to the class website and under the "Readings & Assignments" » "Week 3" tab, click on the "Problem set 3 submission link"
- Submit both .Rmd and pdf files
- Use this naming convention "lastname_firstname_ps#" for your .Rmd and pdf files (e.g. jaquette_ozan_ps3.Rmd & jaquette_ozan_ps3.pdf)