Assignment 3

Due at 11:59pm on October 14.

You may work in pairs or individually for this assignment. Make sure you join a group in Canvas if you are working in pairs. Turn in this assignment as an HTML or PDF file to ELMS. Make sure to include the R Markdown or Quarto file that was used to generate it. Include the GitHub link for the repository containing these files.

```r
library(xml2)
library(rvest)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.1     v stringr   1.5.2
v ggplot2   4.0.0     v tibble    3.3.0
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.1.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter()         masks stats::filter()
x readr::guess_encoding() masks rvest::guess_encoding()
x dplyr::lag()            masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

- Working Repo could be found at: https://github.com/zzeng05/Zeng1-Liu2-a3.git

Web Scraping

In this assignment, your task is to scrape some information from Wikipedia. We start with the following page about Grand Boulevard, a Chicago Community Area.

https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago

The ultimate goal is to gather the table “Historical population” and convert it to a `data.frame`.

As a first step, read in the html page as an R object. Extract the tables from this object (using the `rvest` package) and save the result as a new object. Follow the instructions if there is an error. Use `str()` on this new object – it should be a list. Try to find the position of the “Historical population” in this list since we need it in the next step.

```r
# read in the html page
GB <- read_html("https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago")

GB_tables <- GB %>%
  html_elements("table") %>%
  html_table(fill = TRUE)

str(GB_tables, max.level = 1)
```

```
List of 7
 $ : tibble [27 x 2] (S3: tbl_df/tbl/data.frame)
 $ : tibble [11 x 4] (S3: tbl_df/tbl/data.frame)
 $ : tibble [6 x 17] (S3: tbl_df/tbl/data.frame)
 $ : tibble [5 x 3] (S3: tbl_df/tbl/data.frame)
 $ : tibble [9 x 2] (S3: tbl_df/tbl/data.frame)
 $ : tibble [2 x 2] (S3: tbl_df/tbl/data.frame)
 $ : tibble [2 x 2] (S3: tbl_df/tbl/data.frame)
```

```r
# Find historical population
GB_tables[[1]]
```

```
# A tibble: 27 x 2
   `Grand Boulevard`                                           `Grand Boulevard`
   <chr>                                                       <chr>
 1 Community area                                              Community area
 2 Community Area 38 - Grand Boulevard                         Community Area 3~
 3 The Harold Washington Cultural Center                       The Harold Washi~
 4 Location within the city of Chicago                         Location within ~
 5 Coordinates: .mw-parser-output .geo-default,.mw-parser-out~ Coordinates: .mw~
 6 Country                                                     United States
 7 State                                                       Illinois
 8 County                                                      Cook
 9 City                                                        Chicago
10 Named after                                                 Grand Boulevard ~
# i 17 more rows
```

```r
GB_tables[[2]]
```

```
# A tibble: 11 x 4
   Census Pop.    .mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0)~1 `%±`
   <chr>  <chr>   <chr>                                                    <chr>
 1 1930   87,005  ""                                                       -
 2 1940   103,256 ""                                                       18.7%
 3 1950   114,557 ""                                                       10.9%
 4 1960   80,036  ""                                                       -30.~
 5 1970   80,166  ""                                                       0.2%
 6 1980   53,741  ""                                                       -33.~
 7 1990   35,897  ""                                                       -33.~
 8 2000   28,006  ""                                                       -22.~
 9 2010   21,929  ""                                                       -21.~
10 2020   24,589  ""                                                       12.1%
11 [3][1] [3][1]  "[3][1]"                                                 [3][~
# i abbreviated name:
#   1: `.mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0);clip-path:polygon(0px 0px,0px
```

```r
GB_tables[[3]]
```

```
# A tibble: 6 x 17
  Places adjacent to Gran~1 Places adjacent to G~2 ``    ``    ``    ``    ``
  <chr>                     <chr>                  <chr> <chr> <chr> <lgl> <lgl>
1 "Armour Square, Chicago\~ "Armour Square, Chica~ "Arm~ Doug~ Oakl~ NA    NA
2 "Armour Square, Chicago"  "Douglas, Chicago"     "Oak~ <NA>  <NA>  NA    NA
3 ""                        ""                     ""    <NA>  <NA>  NA    NA
4 "Fuller Park, Chicago"    "Grand Boulevard, Chi~ "Ken~ <NA>  <NA>  NA    NA
5 ""                        ""                     ""    <NA>  <NA>  NA    NA
6 "New City, Chicago"       "Washington Park, Chi~ "Hyd~ <NA>  <NA>  NA    NA
# i abbreviated names: 1: `Places adjacent to Grand Boulevard, Chicago`,
#   2: `Places adjacent to Grand Boulevard, Chicago`
# i 10 more variables: `` <chr>, `` <chr>, `` <chr>, `` <chr>, `` <chr>,
#   `` <chr>, `` <chr>, `` <chr>, `` <chr>, `` <chr>
```

```r
GB_tables[[4]]
```

```
# A tibble: 5 x 3
  X1                        X2                          X3
  <chr>                     <chr>                       <chr>
```

```
1 "Armour Square, Chicago" "Douglas, Chicago"         "Oakland, Chicago"
2 ""                       ""                         ""
3 "Fuller Park, Chicago"   "Grand Boulevard, Chicago" "Kenwood, Chicago"
4 ""                       ""                         ""
5 "New City, Chicago"      "Washington Park, Chicago" "Hyde Park, Chicago"
```

```
GB_tables[[5]]
```

```
# A tibble: 9 x 2
  .mw-parser-output .navbar{display:inline;font-size:88~1 .mw-parser-output .n~2
  <chr>                                                   <chr>
1 Far North                                               "Rogers Park\nWest Ri~
2 Northwest                                               "Portage Park\nIrving~
3 North                                                   "North Center\nLake V~
4 Central                                                 "Near North Side\nThe~
5 West                                                    "Humboldt Park\nWest ~
6 South                                                   "Armour Square\nDougl~
7 Southwest                                               "Garfield Ridge\nArch~
8 Far Southwest                                           "Ashburn\nAuburn Gres~
9 Far Southeast                                           "Chatham\nAvalon Park~
# i abbreviated names:
#   1: `.mw-parser-output .navbar{display:inline;font-size:88%;font-weight:normal}.mw-parser
#   2: `.mw-parser-output .navbar{display:inline;font-size:88%;font-weight:normal}.mw-parser
```

```
GB_tables[[6]]
```

```
# A tibble: 2 x 2
  `vteNeighborhoods in Chicago`                        vteNeighborhoods in Ch~1
  <chr>                                                <chr>
1 Recognized by the city                               "Albany Park\nAndersonv~
2 Other districts and areas recognized by the community "Altgeld Gardens\nArmou~
# i abbreviated name: 1: `vteNeighborhoods in Chicago`
```

```
GB_tables[[7]]
```

```
# A tibble: 2 x 2
  `vte Chicago`                                                    `vte Chicago`
  <chr>                                                            <chr>
1 "Architecture\nBeaches\nClimate\ntornadoes\nColleges and univer~ "Architectur~
2 "Portal\n Category"                                              "Portal\n Ca~
```

- table 2 is the historical population

Extract the "Historical population" table from the list and save it as another object. You can use subsetting via `[[…]]` to extract pieces from a list. Print the result.

```
hist_pop <- GB_tables[[2]]
print(hist_pop)
```

```
# A tibble: 11 x 4
   Census Pop.    .mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0)~1 `%±`
   <chr>  <chr>   <chr>                                                  <chr>
 1 1930   87,005  ""                                                     -
 2 1940   103,256 ""                                                     18.7%
 3 1950   114,557 ""                                                     10.9%
 4 1960   80,036  ""                                                     -30.~
 5 1970   80,166  ""                                                     0.2%
 6 1980   53,741  ""                                                     -33.~
 7 1990   35,897  ""                                                     -33.~
 8 2000   28,006  ""                                                     -22.~
 9 2010   21,929  ""                                                     -21.~
10 2020   24,589  ""                                                     12.1%
11 [3][1] [3][1]  "[3][1]"                                               [3][~
# i abbreviated name:
#   1: `.mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0);clip-path:polygon(0px 0px,0px
```

You will see that the table needs some additional formatting. Keep only want rows and columns with actual values.

```
colnames(hist_pop)
```

```
[1] "Census"
[2] "Pop."
[3] ".mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0);clip-path:polygon(0px 0px,0px 0p
[4] "%±"
```

```
hist_pop_cleaned <- hist_pop %>%
  select(Census, Pop., `%±`) %>%
  slice(-n())

print(hist_pop_cleaned)
```

```
# A tibble: 10 x 3
   Census Pop.    `%±`  
   <chr>  <chr>   <chr> 
 1 1930   87,005  -     
 2 1940   103,256 18.7% 
 3 1950   114,557 10.9% 
 4 1960   80,036  -30.1%
 5 1970   80,166  0.2%  
 6 1980   53,741  -33.0%
 7 1990   35,897  -33.2%
 8 2000   28,006  -22.0%
 9 2010   21,929  -21.7%
10 2020   24,589  12.1% 
```

Expanding to More Pages

That's it for this page. However, we may want to repeat this process for other community areas. The Wikipedia page https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago has a section on "Places adjacent to Grand Boulevard, Chicago" at the bottom. Can you find the corresponding table in the list of tables that you created earlier? Extract this table as a new object.

```
# table for places adjacent
GB_tables[[3]]
```

```
# A tibble: 6 x 17
  Places adjacent to Gran~1 Places adjacent to G~2 ``    ``    ``    ``    ``   
  <chr>                     <chr>                  <chr> <chr> <chr> <lgl> <lgl>
1 "Armour Square, Chicago\~ "Armour Square, Chica~ "Arm~ Doug~ Oakl~ NA    NA   
2 "Armour Square, Chicago"  "Douglas, Chicago"     "Oak~ <NA>  <NA>  NA    NA   
3 ""                        ""                     ""    <NA>  <NA>  NA    NA   
4 "Fuller Park, Chicago"    "Grand Boulevard, Chi~ "Ken~ <NA>  <NA>  NA    NA   
5 ""                        ""                     ""    <NA>  <NA>  NA    NA   
6 "New City, Chicago"       "Washington Park, Chi~ "Hyd~ <NA>  <NA>  NA    NA   
# i abbreviated names: 1: `Places adjacent to Grand Boulevard, Chicago`,
#   2: `Places adjacent to Grand Boulevard, Chicago`
# i 10 more variables: `` <chr>, `` <chr>, `` <chr>, `` <chr>, `` <chr>,
#   `` <chr>, `` <chr>, `` <chr>, `` <chr>, `` <chr>
```

```
GB_adjacent <- GB_tables[[3]]
```

Then, grab the community areas east of Grand Boulevard and save them as a character vector. Print the result.

```
GB_east <- GB_adjacent[[3]] [-1] %>% # Third column is for east except in first row
  discard(~ .x == "")          # drop empty strings

print(GB_east)
```

```
[1] "Oakland, Chicago"   "Kenwood, Chicago"   "Hyde Park, Chicago"
```

We want to use this list to create a loop that extracts the population tables from the Wikipedia pages of these places. To make this work and build valid urls, we need to replace empty spaces in the character vector with underscores. The resulting vector should look like this: "Oakland,_Chicago" "Kenwood,_Chicago" "Hyde_Park,_Chicago"

```
GB_east_urls <- GB_east %>%
str_replace_all(" ", "_")

print(GB_east_urls)
```

```
[1] "Oakland,_Chicago"   "Kenwood,_Chicago"   "Hyde_Park,_Chicago"
```

Build a loop to grab the population tables from each page. Add columns to the original table using `cbind()`.

```
base_url <- "https://en.wikipedia.org/wiki/"
east_pop_list <- list()

for (i in GB_east_urls) {
  url <- paste0(base_url, i)

  page <- read_html(url)
  tables <- page %>% html_elements("table") %>% html_table(fill = TRUE)

  hist_table <- tables %>%
    keep(~ all(c("Census", "Pop.") %in% names(.x))) %>%
    first()

  if (!is.null(hist_table)) {
    clean_table <- hist_table %>%
      select(Census, Pop.) %>%
      rename(!!i := Pop.)

    east_pop_list[[i]] <- clean_table
  } else {
    print("False")
  }
}

east_pop_df <- reduce(east_pop_list, full_join, by = "Census")
final_pop_table <- full_join(hist_pop_cleaned, east_pop_df, by = "Census") %>%
  filter(
    !str_detect(Census, "\\["),
    !Census %in% c("1910", "1920")
  )

print(final_pop_table)
```

```
# A tibble: 10 x 6
   Census Pop.  `%±`  `Oakland,_Chicago` `Kenwood,_Chicago` `Hyde_Park,_Chicago`
   <chr>  <chr> <chr> <chr>              <chr>              <chr>
 1 1930   87,0~ -     14,962             26,942             48,017
 2 1940   103,~ 18.7% 14,500             29,611             50,550
 3 1950   114,~ 10.9% 24,464             35,705             55,206
 4 1960   80,0~ -30.~ 24,378             41,533             45,577
 5 1970   80,1~ 0.2%  18,291             26,890             33,531
 6 1980   53,7~ -33.~ 16,748             21,974             31,198
 7 1990   35,8~ -33.~ 8,197              18,178             28,630
 8 2000   28,0~ -22.~ 6,110              18,363             29,920
 9 2010   21,9~ -21.~ 5,918              17,841             25,681
10 2020   24,5~ 12.1% 6,799              19,116             29,456
```

Scraping and Analyzing Text Data

Suppose we wanted to take the actual text from the Wikipedia pages instead of just the information in the table. Our goal in this section is to extract the text from the body of the pages, then do some basic text cleaning and analysis.

First, scrape just the text without any of the information in the margins or headers. For example, for “Grand Boulevard”, the text should start with, “**Grand Boulevard** on the South Side of Chicago, Illinois, is one of the …”. Make sure all of the text is in one block by using something like the code below (I called my object `description`).

```
# description <- description %>% paste(collapse = ' ')
```

```
# Grand Boulevard page text
GB_url <- "https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago"

description <- read_html(GB_url) %>%
  html_element("#mw-content-text") %>%      # main content
  html_elements("p") %>%                    # paragraphs only
  html_text2() %>%                          # clean text
  paste(collapse = " ")                     # in one block

description
```

```
[1] " Grand Boulevard on the South Side of Chicago, Illinois, is one of the city's Community
King College in Englewood. A high school diploma had been earned by 85.5% of Grand Boulevard
```

Using a similar loop as in the last section, grab the descriptions of the various communities areas. Make a tibble with two columns: the name of the location and the text describing the location.

```
# function to fetch body text from a given Wikipedia slug
get_description <- function(title_slug) {
  url <- paste0("https://en.wikipedia.org/wiki/", title_slug)
  txt <- read_html(url) %>%
    html_element("#mw-content-text") %>%
    html_elements("p") %>%
    html_text2() %>%
    paste(collapse = " ")
  tibble(
    place = gsub(",_Chicago", "", gsub("_", " ", title_slug)),
    text  = txt
  )
}

# list of pages to fetch
pages <- unique(c("Grand_Boulevard,_Chicago", GB_east_urls))

descriptions <- bind_rows(lapply(pages, get_description))
descriptions
```

```
# A tibble: 4 x 2
```

```
  place                      text
  <chr>                      <chr>
1 Grand Boulevard, Chicago " Grand Boulevard on the South Side of Chicago, Illi~
2 Oakland, Chicago         " Oakland, located on the South Side of Chicago, Ill~
3 Kenwood, Chicago         " Kenwood, one of Chicago's 77 community areas, is o~
4 Hyde Park, Chicago       " Hyde Park is a neighborhood on the South Side of C~
```

Let's clean the data using `tidytext`. If you have trouble with this section, see the example shown in https://www.tidytextmining.com/tidytext.html

```
library(tidytext)
```

Create tokens using `unnest_tokens`. Make sure the data is in one-token-per-row format. Remove any stop words within the data. What are the most common words used overall?

```
tokens <- descriptions %>%
  unnest_tokens(word, text) %>%             # sort into one token per row
  anti_join(stop_words, by = "word") %>%    # remove stop words
  filter(!grepl("^[0-9]+$", word))          # drop pure numbers

# overall top 20 words
tokens %>%
  count(word, sort = TRUE) %>%
  slice_head(n = 20)
```

```
# A tibble: 20 x 2
   word             n
   <chr>        <int>
 1 park            85
 2 hyde            75
 3 chicago         58
 4 kenwood         40
 5 street          38
 6 south           29
 7 community       28
 8 neighborhood    26
 9 oakland         25
10 lake            23
11 university      19
12 african         18
13 boulevard       17
```

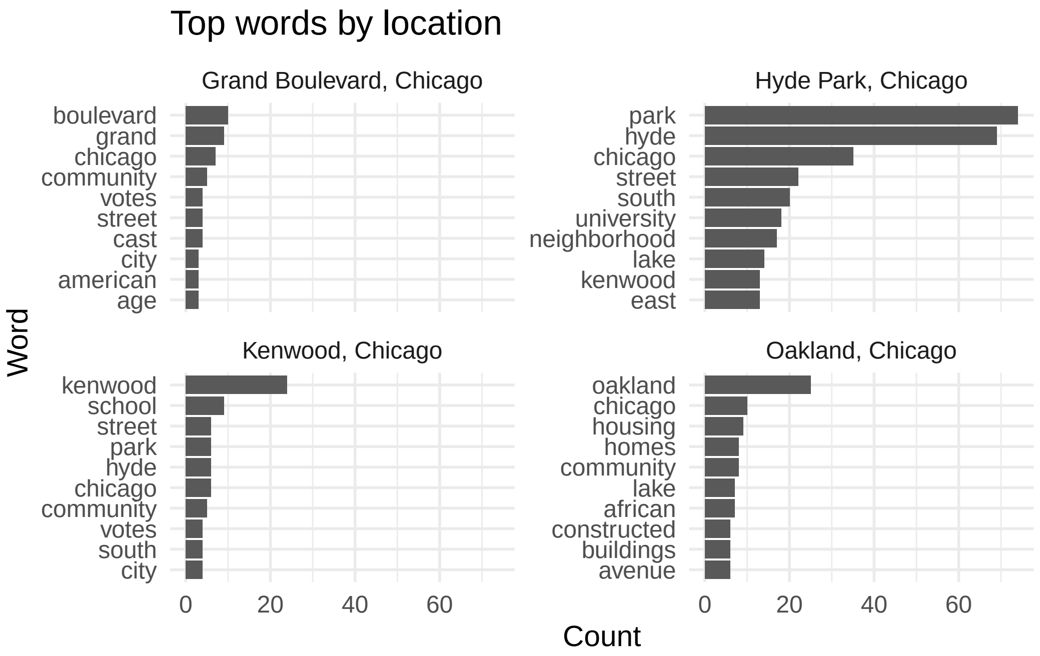
```
14 city            17
15 house           16
16 illinois        16
17 school          16
18 votes           16
19 east            15
20 located         15
```

Plot the most common words within each location. What are some of the similarities between the locations? What are some of the differences?

```r
# counts by place
per_place <- tokens |>
  count(place, word, sort = TRUE)

# top 10 words by place
top10_per_place <- per_place |>
  group_by(place) |>
  slice_max(n, n = 10, with_ties = FALSE) |>
  ungroup()

ggplot(top10_per_place,
       aes(x = n, y = reorder_within(word, n, place))) +
  geom_col() +
  facet_wrap(~ place, scales = "free_y") +
  tidytext::scale_y_reordered() +
  labs(title = "Top words by location",
       x = "Count", y = "Word") +
  theme_minimal()
```


Top words by location
Grand Boulevard, Chicago
boulevard
grand
chicago
community
votes
street
cast
city
american
age
Hyde Park, Chicago
park
hyde
chicago
street
south
university
neighborhood
lake
kenwood
east
Kenwood, Chicago
kenwood
school
street
park
hyde
chicago
community
votes
south
city
Oakland, Chicago
oakland
chicago
housing
homes
community
lake
african
constructed
buildings
avenue
Word
0
20
40
60
Count