Lab-06

Negar

2/24/2022

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Exercise 1: singer

```
songs <- read_csv("https://raw.githubusercontent.com/bwiernik/progdata-class/master/data/singer/songs.cs
```

```
## Rows: 22 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): title, artist_name
## dbl (1): year
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
locations <- read_csv("https://raw.githubusercontent.com/bwiernik/progdata-class/master/data/singer/loc
```

```
## Rows: 14 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): artist_name, city, release, title
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
(time <- as_tibble(songs) |>
   rename(song = title))
```

```
## # A tibble: 22 x 3
##    song                          artist_name      year
##    <chr>                         <chr>           <dbl>
##  1 Corduroy                      Pearl Jam        1994
##  2 Grievance                     Pearl Jam        2000
##  3 Stupidmop                     Pearl Jam        1994
##  4 Present Tense                 Pearl Jam        1996
```

```
##  5 MFC                                Pearl Jam        1998
##  6 Lukin                              Pearl Jam        1996
##  7 It's Lulu                          The Boo Radleys  1995
##  8 Sparrow                            The Boo Radleys  1992
##  9 Martin_ Doom! It's Seven O'Clock The Boo Radleys  1995
## 10 Leaves And Sand                    The Boo Radleys  1993
## # ... with 12 more rows
```

```
(album <- as_tibble(locations) |>
   select(title, everything()) |>
   rename(album = release,
          song  = title))
```

```
## # A tibble: 14 x 4
##    song                                                artist_name  city   album
##    <chr>                                               <chr>        <chr>  <chr>
##  1 "Grievance"                                         Pearl Jam    Seatt~ Binau~
##  2 "Stupidmop"                                         Pearl Jam    Seatt~ Vital~
##  3 "Present Tense"                                     Pearl Jam    Seatt~ No Co~
##  4 "MFC"                                               Pearl Jam    Seatt~ Live ~
##  5 "Lukin"                                             Pearl Jam    Seatt~ Seatt~
##  6 "Stuck On Amber"                                    The Boo Rad~ Liver~ Wake ~
##  7 "It's Lulu"                                         The Boo Rad~ Liver~ Best ~
##  8 "Sparrow"                                           The Boo Rad~ Liver~ Every~
##  9 "High as Monkeys"                                   The Boo Rad~ Liver~ Kings~
## 10 "Butterfly McQueen"                                 The Boo Rad~ Liver~ Giant~
## 11 "My One and Only Love"                              Carly Simon  New Y~ Moonl~
## 12 "It Was So Easy  (LP Version)"                      Carly Simon  New Y~ No Se~
## 13 "I've Got A Crush On You"                           Carly Simon  New Y~ Cloud~
## 14 "Manha De Carnaval (Theme from \"Black Orpheus\")" Carly Simon  New Y~ Into ~
```

1. We really care about the songs in time. But, for which of those songs do we know its corresponding album?

```
time |>
  left_join(album, by = c("song","artist_name"))
```

```
## # A tibble: 22 x 5
##    song                                artist_name      year city      album
##    <chr>                               <chr>           <dbl> <chr>     <chr>
##  1 Corduroy                            Pearl Jam        1994 <NA>      <NA>
##  2 Grievance                           Pearl Jam        2000 Seattle,~ Binaural
##  3 Stupidmop                           Pearl Jam        1994 Seattle,~ Vitalogy
##  4 Present Tense                       Pearl Jam        1996 Seattle,~ No Code
##  5 MFC                                 Pearl Jam        1998 Seattle,~ Live On Two~
##  6 Lukin                               Pearl Jam        1996 Seattle,~ Seattle Was~
##  7 It's Lulu                           The Boo Radleys  1995 Liverpoo~ Best Of
##  8 Sparrow                             The Boo Radleys  1992 Liverpoo~ Everything'~
##  9 Martin_ Doom! It's Seven O'Clock The Boo Radleys  1995 <NA>      <NA>
## 10 Leaves And Sand                     The Boo Radleys  1993 <NA>      <NA>
## # ... with 12 more rows
```

2. Go ahead and add the corresponding albums to the time tibble, being sure to preserve rows even if album info is not readily available.

```
time |>
  full_join(album, by = c("song","artist_name"))
```

```
## # A tibble: 23 x 5
##    song                              artist_name      year city      album
##    <chr>                             <chr>           <dbl> <chr>     <chr>
##  1 Corduroy                          Pearl Jam        1994 <NA>      <NA>
##  2 Grievance                         Pearl Jam        2000 Seattle,~ Binaural
##  3 Stupidmop                         Pearl Jam        1994 Seattle,~ Vitalogy
##  4 Present Tense                     Pearl Jam        1996 Seattle,~ No Code
##  5 MFC                               Pearl Jam        1998 Seattle,~ Live On Two~
##  6 Lukin                             Pearl Jam        1996 Seattle,~ Seattle Was~
##  7 It's Lulu                         The Boo Radleys  1995 Liverpoo~ Best Of
##  8 Sparrow                           The Boo Radleys  1992 Liverpoo~ Everything'~
##  9 Martin_ Doom! It's Seven O'Clock The Boo Radleys  1995 <NA>      <NA>
## 10 Leaves And Sand                   The Boo Radleys  1993 <NA>      <NA>
## # ... with 13 more rows
```

3. Which songs do we have "year", but not album info?

```
time |>
  anti_join(album, by = "song")
```

```
## # A tibble: 9 x 3
##   song                              artist_name      year
##   <chr>                             <chr>           <dbl>
## 1 Corduroy                          Pearl Jam        1994
## 2 Martin_ Doom! It's Seven O'Clock The Boo Radleys  1995
## 3 Leaves And Sand                   The Boo Radleys  1993
## 4 Comb Your Hair                    The Boo Radleys  1998
## 5 Mine Again                        Mariah Carey     2005
## 6 Don't Forget About Us             Mariah Carey     2005
## 7 Babydoll                          Mariah Carey     1997
## 8 Don't Forget About Us             Mariah Carey     2005
## 9 Vision Of Love                    Mariah Carey     1990
```

4. Which artists are in time, but not in album?

```
time |>
  anti_join(album, by = "artist_name")
```

```
## # A tibble: 5 x 3
##   song                  artist_name   year
##   <chr>                 <chr>        <dbl>
## 1 Mine Again            Mariah Carey  2005
## 2 Don't Forget About Us Mariah Carey  2005
## 3 Babydoll              Mariah Carey  1997
## 4 Don't Forget About Us Mariah Carey  2005
## 5 Vision Of Love        Mariah Carey  1990
```

5. You've come across these two tibbles, and just wish all the info was available in one tibble. What would you do?

```
time |>
  full_join(album, by = "song")
```

```
## # A tibble: 23 x 6
##    song            artist_name.x   year artist_name.y  city      album
##    <chr>           <chr>          <dbl> <chr>          <chr>     <chr>
##  1 Corduroy        Pearl Jam       1994 <NA>           <NA>      <NA>
##  2 Grievance       Pearl Jam       2000 Pearl Jam      Seattle,~ Binaural
```

```
##  3 Stupidmop       Pearl Jam       1994 Pearl Jam      Seattle,~ Vitalogy
##  4 Present Tense   Pearl Jam       1996 Pearl Jam      Seattle,~ No Code
##  5 MFC             Pearl Jam       1998 Pearl Jam      Seattle,~ Live On Two Le~
##  6 Lukin           Pearl Jam       1996 Pearl Jam      Seattle,~ Seattle Washin~
##  7 It's Lulu       The Boo Radle~  1995 The Boo Radle~ Liverpoo~ Best Of
##  8 Sparrow         The Boo Radle~  1992 The Boo Radle~ Liverpoo~ Everything's A~
##  9 Martin_ Doom! ~ The Boo Radle~  1995 <NA>           <NA>      <NA>
## 10 Leaves And Sand The Boo Radle~  1993 <NA>           <NA>      <NA>
## # ... with 13 more rows
```

Exercise 2: LOTR

```
fell <- read_csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Fellowship_Of_The
```

```
## Rows: 3 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Film, Race
## dbl (2): Female, Male
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
ttow <- read_csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Two_Towers.csv")
```

```
## Rows: 3 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Film, Race
## dbl (2): Female, Male
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
retk <- read_csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Return_Of_The_King
```

```
## Rows: 3 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Film, Race
## dbl (2): Female, Male
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

1. Stack these into a single tibble.

```
retk |>
  full_join(fell, by = c("Film", "Race", "Female", "Male")) |>
  full_join(ttow, by = c("Film", "Race", "Female", "Male"))
```

```
## # A tibble: 9 x 4
##   Film                      Race   Female  Male
##   <chr>                     <chr>   <dbl> <dbl>
## 1 The Return Of The King    Elf       183   510
## 2 The Return Of The King    Hobbit      2  2673
## 3 The Return Of The King    Man       268  2459
## 4 The Fellowship Of The Ring Elf     1229   971
```

```
## 5 The Fellowship Of The Ring Hobbit     14  3644
## 6 The Fellowship Of The Ring Man         0  1995
## 7 The Two Towers             Elf       331   513
## 8 The Two Towers             Hobbit      0  2463
## 9 The Two Towers             Man       401  3589
```

2. Which races are present in "The Fellowship of the Ring" (fell), but not in any of the other ones?

```
fell |>
  anti_join(retk, by = "Race") |>
  anti_join(ttow, by = "Race")
```

```
## # A tibble: 0 x 4
## # ... with 4 variables: Film <chr>, Race <chr>, Female <dbl>, Male <dbl>
```

Exercise 3: Set Operations

```
(y <-  tibble(x1 = LETTERS[1:3], x2 = 1:3))
```

```
## # A tibble: 3 x 2
##   x1       x2
##   <chr> <int>
## 1 A         1
## 2 B         2
## 3 C         3
```

```
(z <- tibble(x1 = c("B", "C", "D"), x2 = 2:4))
```

```
## # A tibble: 3 x 2
##   x1       x2
##   <chr> <int>
## 1 B         2
## 2 C         3
## 3 D         4
```

1. Rows that appear in both y and z.

```
intersect(y, z)
```

```
## # A tibble: 2 x 2
##   x1       x2
##   <chr> <int>
## 1 B         2
## 2 C         3
```

2. You collected the data in y on Day 1, and z in Day 2. Make a data set to reflect that.

```
union(
  mutate(y, day = "Day 1"),
  mutate(z, day = "Day 2")
)
```

```
## # A tibble: 6 x 3
##   x1       x2 day
##   <chr> <int> <chr>
## 1 A         1 Day 1
## 2 B         2 Day 1
## 3 C         3 Day 1
## 4 B         2 Day 2
## 5 C         3 Day 2
```

```
## 6 D         4 Day 2
```

3. The rows contained in z are bad! Remove those rows from y.

```
setdiff(y, z)
```

```
## # A tibble: 1 x 2
##   x1       x2
##   <chr> <int>
## 1 A         1
```