HW_6 Data 412

Ethan Pastel

2024-02-01

Identifying Table Keys in the NASA Weather Dataset

You might need to install the {nasaweather} package using the console

Read the description of the {nasaweather} dataset with the below

```
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   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     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 errors
```

```
library(dplyr)
library(ggplot2)
library(nasaweather)
```

```
##
## Attaching package: 'nasaweather'
##
## The following object is masked from 'package:dplyr':
##
##     storms
```

```
library(help = "nasaweather")
```

1. What are the data frames in this data set?

The data frames that are in the data set 'nasaweather' are atmos: Atmospheric Data, borders: Country Borders, elev: Elevation, glaciers: Glacier locations, storms: Storm track data.

2. What are the keys in each data frame?

atmos: year, month, lat, and long.

elev: lat and long

glaciers: id

storms: name, year, month, day, and hour

3. For "atoms", "elev", and "glaciers" demonstrate the keys generate unique rows.

```
unique_atmos <- atmos %>%
  distinct(year, month, lat, long, .keep_all = TRUE)

unique_elev <- elev %>%
  distinct(lat, long, .keep_all = TRUE)

unique_glaciers <- glaciers %>%
  distinct(id, .keep_all = TRUE)

head(unique_atmos, 20)
```

```
## # A tibble: 20 x 11
##       lat  long  year month surftemp  temp pressure ozone cloudlow cloudmid
##     <dbl> <dbl> <int> <int>    <dbl> <dbl>    <dbl> <dbl>    <dbl>    <dbl>
##  1  36.2  -114.  1995     1     273.  272.      835   304      7.5     34.5
##  2  33.7  -114.  1995     1     280.  282.      940   304     11.5     32.5
##  3  31.2  -114.  1995     1     285.  285.      960   298     16.5     26
##  4  28.7  -114.  1995     1     289.  291.      990   276     20.5     14.5
##  5  26.2  -114.  1995     1     292.  293.     1000   274     26       10.5
##  6  23.7  -114.  1995     1     294.  294.     1000   264     30        9.5
##  7  21.2  -114.  1995     1     295   295.     1000   258     29.5     11
##  8  18.7  -114.  1995     1     298.  297.     1000   252     26.5     17.5
##  9  16.2  -114.  1995     1     300.  298.     1000   250     27.5     18.5
## 10  13.7  -114.  1995     1     300.  299.     1000   250     26       16.5
## 11  11.2  -114.  1995     1     301   300.     1000   248     28.5     12.5
## 12   8.75 -114.  1995     1     301   300.     1000   248     28       13.5
## 13   6.25 -114.  1995     1     299.  300.     1000   250     33       18.5
## 14   3.76 -114.  1995     1     299.  300.     1000   248     44.5     13
## 15   1.26 -114.  1995     1     298.  298.     1000   248     43.5      4
## 16  -1.23 -114.  1995     1     298.  298.     1000   248     37        1
## 17  -3.73 -114.  1995     1     299.  298.     1000   248     29        1.5
## 18  -6.23 -114.  1995     1     299.  298.     1000   250     25.5      4
## 19  -8.72 -114.  1995     1     299.  298.     1000   252     27        4.5
## 20 -11.2  -114.  1995     1     298.  298.     1000   252     34.5      4.5
```

```
## # i 1 more variable: cloudhigh <dbl>
```

```
head(unique_elev, 20)
```

```
## # A tibble: 20 x 3
##     long    lat    elev
##    <dbl>  <dbl>   <dbl>
##  1 -114. -21.2     0
##  2 -114. -18.7     0
##  3 -114. -16.2     0
##  4 -114. -13.7     0
##  5 -114. -11.2     0
##  6 -114.  -8.72    0
##  7 -114.  -6.23    0
##  8 -114.  -3.73    0.19
##  9 -114.  -1.23    0.62
## 10 -114.   1.26  132.
## 11 -114.   3.76  306.
## 12 -114.   6.25  459.
## 13 -114.   8.75  325.
## 14 -114.  11.2   103.
## 15 -114.  13.7    82.4
## 16 -114.  16.2   334.
## 17 -114.  18.7   232.
## 18 -114.  21.2   472.
## 19 -114.  23.7   865.
## 20 -114.  26.2  1693.
```

```
head(unique_glaciers, 20)
```

```
## # A tibble: 20 x 6
##    id          name          lat  long area   country
##    <chr>       <chr>       <dbl> <dbl> <chr>  <chr>
##  1 CO1A0101001 RAMIREZ E 4  10.8 -73.6 " NA"  CO
##  2 CO1A0101002 RAMIREZ E 3  10.8 -73.6 " NA"  CO
##  3 CO1A0101003 RAMIREZ E 2  10.8 -73.6 " NA"  CO
##  4 CO1A0101004 RAMIREZ E 1  10.8 -73.6 "0.03" CO
##  5 CO1A0101005 RAMIREZ 5 N  10.8 -73.6 "0.1"  CO
##  6 CO1A0101007 RAMIREZ 3 N  10.8 -73.6 "0.03" CO
##  7 CO1A0101008 RAMIREZ 2 N  10.8 -73.6 "0.04" CO
##  8 CO1A0101009 RAMIREZ 1 N  10.8 -73.6 "0.03" CO
##  9 CO1A0101010 REINA NINA   10.8 -73.6 "0.01" CO
## 10 CO1A0101011 REINA 7      10.8 -73.6 " NA"  CO
## 11 CO1A0101012 REINA B      10.8 -73.6 "0.85" CO
## 12 CO1A0101014 OJEDA S 3    10.8 -73.6 "0.04" CO
## 13 CO1A0101015 OJEDA S 4    10.8 -73.6 "0.09" CO
## 14 CO1A0101016 OJEDA S 2    10.8 -73.6 "0.01" CO
## 15 CO1A0101017 NUEVO 1      10.8 -73.6 "0.06" CO
## 16 CO1A0101018 NUEVO 2      10.8 -73.6 "0.02" CO
## 17 CO1A0101019 NUEVO 3      10.8 -73.6 "0.04" CO
## 18 CO1A0101020 NUEVO 4      10.8 -73.6 "0.02" CO
## 19 CO1A0101021 CORAL 1      10.8 -73.6 "0.07" CO
## 20 CO1A0102001 NUEVO 9      10.8 -73.6 "0.01" CO
```

Lahman's Baseball Dataset

You may need to install the {Lahman} package using the console. You can read about it with:

```
library(Lahman)
```

```
## Warning: package 'Lahman' was built under R version 4.3.2
```

For this exercise, we'll use the People, Batting, Pitching, Fielding, Teams, and Salaries data frames

1. Load these six data frames into R and read about them.

```
data("People")
data("Batting")
data("Pitching")
data("Fielding")
data("Teams")
data("Salaries")

str(People)
```

```
## 'data.frame':    20676 obs. of  26 variables:
##  $ playerID    : chr  "aardsda01" "aaronha01" "aaronto01" "aasedo01" ...
##  $ birthYear   : int  1981 1934 1939 1954 1972 1985 1850 1877 1869 1866 ...
##  $ birthMonth  : int  12 2 8 9 8 12 11 4 11 10 ...
##  $ birthDay    : int  27 5 5 8 25 17 4 15 11 14 ...
##  $ birthCountry: chr  "USA" "USA" "USA" "USA" ...
##  $ birthState  : chr  "CO" "AL" "AL" "CA" ...
##  $ birthCity   : chr  "Denver" "Mobile" "Mobile" "Orange" ...
##  $ deathYear   : int  NA 2021 1984 NA NA NA 1905 1957 1962 1926 ...
##  $ deathMonth  : int  NA 1 8 NA NA NA 5 1 6 4 ...
##  $ deathDay    : int  NA 22 16 NA NA NA 17 6 11 27 ...
##  $ deathCountry: chr  NA "USA" "USA" NA ...
##  $ deathState  : chr  NA "GA" "GA" NA ...
##  $ deathCity   : chr  NA "Atlanta" "Atlanta" NA ...
##  $ nameFirst   : chr  "David" "Hank" "Tommie" "Don" ...
##  $ nameLast    : chr  "Aardsma" "Aaron" "Aaron" "Aase" ...
##  $ nameGiven   : chr  "David Allan" "Henry Louis" "Tommie Lee" "Donald William" ...
##  $ weight      : int  215 180 190 190 184 235 192 170 175 169 ...
##  $ height      : int  75 72 75 75 73 74 72 71 71 68 ...
##  $ bats        : Factor w/ 3 levels "B","L","R": 3 3 3 3 2 2 3 3 3 2 ...
##  $ throws      : Factor w/ 3 levels "L","R","S": 2 2 2 2 1 1 2 2 2 1 ...
##  $ debut       : chr  "2004-04-06" "1954-04-13" "1962-04-10" "1977-07-26" ...
##  $ finalGame   : chr  "2015-08-23" "1976-10-03" "1971-09-26" "1990-10-03" ...
##  $ retroID     : chr  "aardd001" "aaroh101" "aarot101" "aased001" ...
```

```
##  $ bbrefID     : chr  "aardsda01" "aaronha01" "aaronto01" "aasedo01" ...
##  $ deathDate   : Date, format: NA "2021-01-22" ...
##  $ birthDate   : Date, format: "1981-12-27" "1934-02-05" ...
```

```
str(Batting)
```

```
## 'data.frame':    112184 obs. of  22 variables:
##  $ playerID: chr  "abercda01" "addybo01" "allisar01" "allisdo01" ...
##  $ yearID  : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1871 ...
##  $ stint   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ teamID  : Factor w/ 149 levels "ALT","ANA","ARI",..: 136 111 39 142 111 56 111 24 56 24 ...
##  $ lgID    : Factor w/ 7 levels "AA","AL","FL",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ G       : int  1 25 29 27 25 12 1 31 1 18 ...
##  $ AB      : int  4 118 137 133 120 49 4 157 5 86 ...
##  $ R       : int  0 30 28 28 29 9 0 66 1 13 ...
##  $ H       : int  0 32 40 44 39 11 1 63 1 13 ...
##  $ X2B     : int  0 6 4 10 11 2 0 10 1 2 ...
##  $ X3B     : int  0 0 5 2 3 1 0 9 0 1 ...
##  $ HR      : int  0 0 0 2 0 0 0 0 0 0 ...
##  $ RBI     : int  0 13 19 27 16 5 2 34 1 11 ...
##  $ SB      : int  0 8 3 1 6 0 0 11 0 1 ...
##  $ CS      : int  0 1 1 1 2 1 0 6 0 0 ...
##  $ BB      : int  0 4 2 0 2 0 1 13 0 0 ...
##  $ SO      : int  0 0 5 2 1 1 0 1 0 0 ...
##  $ IBB     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ HBP     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ SH      : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ SF      : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ GIDP    : int  0 0 1 0 0 0 0 1 0 0 ...
```

```
str(Pitching)
```

```
## 'data.frame':    50402 obs. of  30 variables:
##  $ playerID: chr  "bechtge01" "brainas01" "fergubo01" "fishech01" ...
##  $ yearID  : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1871 ...
##  $ stint   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ teamID  : Factor w/ 149 levels "ALT","ANA","ARI",..: 97 142 90 111 90 136 111 56 97 136 ...
##  $ lgID    : Factor w/ 7 levels "AA","AL","FL",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ W       : int  1 12 0 4 0 0 0 6 18 12 ...
##  $ L       : int  2 15 0 16 1 0 1 11 5 15 ...
##  $ G       : int  3 30 1 24 1 1 3 19 25 29 ...
##  $ GS      : int  3 30 0 24 1 0 1 19 25 29 ...
##  $ CG      : int  2 30 0 22 1 0 1 19 25 28 ...
##  $ SHO     : int  0 0 0 1 0 0 0 1 0 0 ...
##  $ SV      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ IPouts  : int  78 792 3 639 27 3 39 507 666 747 ...
##  $ H       : int  43 361 8 295 20 1 20 261 285 430 ...
##  $ ER      : int  23 132 3 103 10 0 5 97 113 153 ...
##  $ HR      : int  0 4 0 3 0 0 0 5 3 4 ...
##  $ BB      : int  11 37 0 31 3 0 3 21 40 75 ...
##  $ SO      : int  1 13 0 15 0 0 1 17 15 12 ...
##  $ BAOpp   : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ ERA     : num  7.96 4.5 27 4.35 10 0 3.46 5.17 4.58 5.53 ...
```

```
##  $ IBB     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ WP      : int  7 7 2 20 0 0 1 15 3 44 ...
##  $ HBP     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ BK      : int  0 0 0 0 0 0 0 2 0 0 ...
##  $ BFP     : int  146 1291 14 1080 57 3 70 876 1059 1334 ...
##  $ GF      : int  0 0 0 1 0 1 1 0 0 0 ...
##  $ R       : int  42 292 9 257 21 0 30 243 223 362 ...
##  $ SH      : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ SF      : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ GIDP    : int  NA NA NA NA NA NA NA NA NA NA ...
```

```
str(Fielding)
```

```
## 'data.frame':    149365 obs. of  18 variables:
##  $ playerID: chr  "abercda01" "addybo01" "addybo01" "allisar01" ...
##  $ yearID  : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1871 ...
##  $ stint   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ teamID  : Factor w/ 149 levels "ALT","ANA","ARI",..: 136 111 111 39 39 142 111 111 111 111 ...
##  $ lgID    : Factor w/ 7 levels "AA","AL","FL",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ POS     : chr  "SS" "2B" "SS" "2B" ...
##  $ G       : int  1 22 3 2 29 27 1 2 20 5 ...
##  $ GS      : int  1 22 3 0 29 27 0 1 19 4 ...
##  $ InnOuts : int  24 606 96 18 729 681 15 30 555 93 ...
##  $ PO      : int  1 67 8 1 51 68 7 3 38 10 ...
##  $ A       : int  3 72 14 4 3 15 0 4 52 0 ...
##  $ E       : int  2 42 7 0 7 20 0 1 28 8 ...
##  $ DP      : int  0 5 0 0 1 4 0 0 2 0 ...
##  $ PB      : int  NA NA NA NA NA 18 NA NA NA 7 ...
##  $ WP      : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ SB      : int  NA NA NA NA NA 0 NA NA NA 0 ...
##  $ CS      : int  NA NA NA NA NA 0 NA NA NA 0 ...
##  $ ZR      : int  NA NA NA NA NA NA NA NA NA NA ...
```

```
str(Teams)
```

```
## 'data.frame':    3015 obs. of  48 variables:
##  $ yearID        : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1872 ...
##  $ lgID          : Factor w/ 7 levels "AA","AL","FL",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ teamID        : Factor w/ 149 levels "ALT","ANA","ARI",..: 24 31 39 56 90 97 111 136 142 8 ...
##  $ franchID      : Factor w/ 120 levels "ALT","ANA","ARI",..: 13 36 25 56 70 85 91 109 77 9 ...
##  $ divID         : chr  NA NA NA NA ...
##  $ Rank          : int  3 2 8 7 5 1 9 6 4 2 ...
##  $ G             : int  31 28 29 19 33 28 25 29 32 58 ...
##  $ Ghome         : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ W             : int  20 19 10 7 16 21 4 13 15 35 ...
##  $ L             : int  10 9 19 12 17 7 21 15 15 19 ...
##  $ DivWin        : chr  NA NA NA NA ...
##  $ WCWin         : chr  NA NA NA NA ...
##  $ LgWin         : chr  "N" "N" "N" "N" ...
##  $ WSWin         : chr  NA NA NA NA ...
##  $ R             : int  401 302 249 137 302 376 231 351 310 617 ...
##  $ AB            : int  1372 1196 1186 746 1404 1281 1036 1248 1353 2571 ...
##  $ H             : int  426 323 328 178 403 410 274 384 375 753 ...
```

```
##  $ X2B           : int  70 52 35 19 43 66 44 51 54 106 ...
##  $ X3B           : int  37 21 40 8 21 27 25 34 26 31 ...
##  $ HR            : int  3 10 7 2 1 9 3 6 6 14 ...
##  $ BB            : int  60 60 26 33 33 46 38 49 48 29 ...
##  $ SO            : int  19 22 25 9 15 23 30 19 13 28 ...
##  $ SB            : int  73 69 18 16 46 56 53 62 48 53 ...
##  $ CS            : int  16 21 8 4 15 12 10 24 13 18 ...
##  $ HBP           : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ SF            : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ RA            : int  303 241 341 243 313 266 287 362 303 434 ...
##  $ ER            : int  109 77 116 97 121 137 108 153 137 166 ...
##  $ ERA           : num  3.55 2.76 4.11 5.17 3.72 4.95 4.3 5.51 4.37 2.9 ...
##  $ CG            : int  22 25 23 19 32 27 23 28 32 48 ...
##  $ SHO           : int  1 0 0 1 1 0 1 0 0 1 ...
##  $ SV            : int  3 1 0 0 0 0 0 0 0 1 ...
##  $ IPouts        : int  828 753 762 507 879 747 678 750 846 1548 ...
##  $ HA            : int  367 308 346 261 373 329 315 431 371 573 ...
##  $ HRA           : int  2 6 13 5 7 3 3 4 4 3 ...
##  $ BBA           : int  42 28 53 21 42 53 34 75 45 63 ...
##  $ SOA           : int  23 22 34 17 22 16 16 12 13 77 ...
##  $ E             : int  243 229 234 163 235 194 220 198 218 432 ...
##  $ DP            : int  24 16 15 8 14 13 14 22 20 22 ...
##  $ FP            : num  0.834 0.829 0.818 0.803 0.84 0.845 0.821 0.845 0.85 0.83 ...
##  $ name          : chr  "Boston Red Stockings" "Chicago White Stockings" "Cleveland Forest Citys" "Fe
##  $ park          : chr  "South End Grounds I" "Union Base-Ball Grounds" "National Association Grounds
##  $ attendance    : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ BPF           : int  103 104 96 101 90 102 97 101 94 106 ...
##  $ PPF           : int  98 102 100 107 88 98 99 100 98 102 ...
##  $ teamIDBR      : chr  "BOS" "CHI" "CLE" "KEK" ...
##  $ teamIDlahman45: chr  "BS1" "CH1" "CL1" "FW1" ...
##  $ teamIDretro   : chr  "BS1" "CH1" "CL1" "FW1" ...
```

```
str(Salaries)
```

```
## 'data.frame':    26428 obs. of  5 variables:
##  $ yearID  : int  1985 1985 1985 1985 1985 1985 1985 1985 1985 1985 ...
##  $ teamID  : Factor w/ 35 levels "ANA","ARI","ATL",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ lgID    : Factor w/ 2 levels "AL","NL": 2 2 2 2 2 2 2 2 2 2 ...
##  $ playerID: chr  "barkele01" "bedrost01" "benedbr01" "campri01" ...
##  $ salary  : int  870000 550000 545000 633333 625000 800000 150000 483333 772000 250000 ...
```

```
head(People)
```

```
##    playerID birthYear birthMonth birthDay birthCountry birthState  birthCity
## 1 aardsda01      1981         12       27          USA         CO     Denver
## 2 aaronha01      1934          2        5          USA         AL     Mobile
## 3 aaronto01      1939          8        5          USA         AL     Mobile
## 4  aasedo01      1954          9        8          USA         CA     Orange
## 5  abadan01      1972          8       25          USA         FL Palm Beach
## 6  abadfe01      1985         12       17         D.R.  La Romana  La Romana
##   deathYear deathMonth deathDay deathCountry deathState deathCity nameFirst
## 1        NA         NA       NA         <NA>       <NA>      <NA>     David
## 2      2021          1       22          USA         GA   Atlanta      Hank
```

```
## 3      1984          8       16          USA         GA   Atlanta    Tommie
## 4        NA         NA       NA         <NA>       <NA>      <NA>       Don
## 5        NA         NA       NA         <NA>       <NA>      <NA>      Andy
## 6        NA         NA       NA         <NA>       <NA>      <NA>  Fernando
##   nameLast        nameGiven weight height bats throws      debut  finalGame
## 1  Aardsma      David Allan    215     75    R      R 2004-04-06 2015-08-23
## 2    Aaron      Henry Louis    180     72    R      R 1954-04-13 1976-10-03
## 3    Aaron       Tommie Lee    190     75    R      R 1962-04-10 1971-09-26
## 4     Aase   Donald William    190     75    R      R 1977-07-26 1990-10-03
## 5     Abad    Fausto Andres    184     73    L      L 2001-09-10 2006-04-13
## 6     Abad Fernando Antonio    235     74    L      L 2010-07-28 2021-10-01
##    retroID   bbrefID  deathDate  birthDate
## 1 aardd001 aardsda01       <NA> 1981-12-27
## 2 aaroh101 aaronha01 2021-01-22 1934-02-05
## 3 aarot101 aaronto01 1984-08-16 1939-08-05
## 4 aased001  aasedo01       <NA> 1954-09-08
## 5 abada001  abadan01       <NA> 1972-08-25
## 6 abadf001  abadfe01       <NA> 1985-12-17
```

```
head(Batting)
```

```
##    playerID yearID stint teamID lgID  G  AB  R  H X2B X3B HR RBI SB CS BB SO
## 1 abercda01   1871     1    TRO   NA  1   4  0  0   0   0  0   0  0  0  0  0
## 2  addybo01   1871     1    RC1   NA 25 118 30 32   6   0  0  13  8  1  4  0
## 3 allisar01   1871     1    CL1   NA 29 137 28 40   4   5  0  19  3  1  2  5
## 4 allisdo01   1871     1    WS3   NA 27 133 28 44  10   2  2  27  1  1  0  2
## 5 ansonca01   1871     1    RC1   NA 25 120 29 39  11   3  0  16  6  2  2  1
## 6 armstbo01   1871     1    FW1   NA 12  49  9 11   2   1  0   5  0  1  0  1
##   IBB HBP SH SF GIDP
## 1  NA  NA NA NA    0
## 2  NA  NA NA NA    0
## 3  NA  NA NA NA    1
## 4  NA  NA NA NA    0
## 5  NA  NA NA NA    0
## 6  NA  NA NA NA    0
```

```
head(Pitching)
```

```
##    playerID yearID stint teamID lgID  W  L  G GS CG SHO SV IPouts   H  ER HR BB
## 1 bechtge01   1871     1    PH1   NA  1  2  3  3  2   0  0     78  43  23  0 11
## 2 brainas01   1871     1    WS3   NA 12 15 30 30 30   0  0    792 361 132  4 37
## 3 fergubo01   1871     1    NY2   NA  0  0  1  0  0   0  0      3   8   3  0  0
## 4 fishech01   1871     1    RC1   NA  4 16 24 24 22   1  0    639 295 103  3 31
## 5 fleetfr01   1871     1    NY2   NA  0  1  1  1  1   0  0     27  20  10  0  3
## 6 flowedi01   1871     1    TRO   NA  0  0  1  0  0   0  0      3   1   0  0  0
##   SO BAOpp   ERA IBB WP HBP BK  BFP GF   R SH SF GIDP
## 1  1    NA  7.96  NA  7  NA  0  146  0  42 NA NA   NA
## 2 13    NA  4.50  NA  7  NA  0 1291  0 292 NA NA   NA
## 3  0    NA 27.00  NA  2  NA  0   14  0   9 NA NA   NA
## 4 15    NA  4.35  NA 20  NA  0 1080  1 257 NA NA   NA
## 5  0    NA 10.00  NA  0  NA  0   57  0  21 NA NA   NA
## 6  0    NA  0.00  NA  0  NA  0    3  1   0 NA NA   NA
```

```
head(Fielding)
```

```
##    playerID yearID stint teamID lgID POS  G GS InnOuts PO  A  E DP PB WP SB CS
## 1 abercda01   1871     1    TRO   NA  SS  1  1      24  1  3  2  0 NA NA NA NA
## 2  addybo01   1871     1    RC1   NA  2B 22 22     606 67 72 42  5 NA NA NA NA
## 3  addybo01   1871     1    RC1   NA  SS  3  3      96  8 14  7  0 NA NA NA NA
## 4 allisar01   1871     1    CL1   NA  2B  2  0      18  1  4  0  0 NA NA NA NA
## 5 allisar01   1871     1    CL1   NA  OF 29 29     729 51  3  7  1 NA NA NA NA
## 6 allisdo01   1871     1    WS3   NA   C 27 27     681 68 15 20  4 18 NA  0  0
##   ZR
## 1 NA
## 2 NA
## 3 NA
## 4 NA
## 5 NA
## 6 NA
```

```
head(Teams)
```

```
##   yearID lgID teamID franchID divID Rank  G Ghome  W  L DivWin WCWin LgWin
## 1   1871   NA    BS1      BNA  <NA>    3 31    NA 20 10   <NA>  <NA>     N
## 2   1871   NA    CH1      CNA  <NA>    2 28    NA 19  9   <NA>  <NA>     N
## 3   1871   NA    CL1      CFC  <NA>    8 29    NA 10 19   <NA>  <NA>     N
## 4   1871   NA    FW1      KEK  <NA>    7 19    NA  7 12   <NA>  <NA>     N
## 5   1871   NA    NY2      NNA  <NA>    5 33    NA 16 17   <NA>  <NA>     N
## 6   1871   NA    PH1      PNA  <NA>    1 28    NA 21  7   <NA>  <NA>     Y
##   WSWin   R   AB   H X2B X3B HR BB SO SB CS HBP SF  RA  ER  ERA CG SHO SV
## 1  <NA> 401 1372 426  70  37  3 60 19 73 16  NA NA 303 109 3.55 22   1  3
## 2  <NA> 302 1196 323  52  21 10 60 22 69 21  NA NA 241  77 2.76 25   0  1
## 3  <NA> 249 1186 328  35  40  7 26 25 18  8  NA NA 341 116 4.11 23   0  0
## 4  <NA> 137  746 178  19   8  2 33  9 16  4  NA NA 243  97 5.17 19   1  0
## 5  <NA> 302 1404 403  43  21  1 33 15 46 15  NA NA 313 121 3.72 32   1  0
## 6  <NA> 376 1281 410  66  27  9 46 23 56 12  NA NA 266 137 4.95 27   0  0
##   IPouts  HA HRA BBA SOA   E DP    FP                    name
## 1    828 367   2  42  23 243 24 0.834    Boston Red Stockings
## 2    753 308   6  28  22 229 16 0.829 Chicago White Stockings
## 3    762 346  13  53  34 234 15 0.818  Cleveland Forest Citys
## 4    507 261   5  21  17 163  8 0.803    Fort Wayne Kekiongas
## 5    879 373   7  42  22 235 14 0.840        New York Mutuals
## 6    747 329   3  53  16 194 13 0.845  Philadelphia Athletics
##                           park attendance BPF PPF teamIDBR teamIDlahman45
## 1          South End Grounds I         NA 103  98      BOS            BS1
## 2      Union Base-Ball Grounds         NA 104 102      CHI            CH1
## 3 National Association Grounds         NA  96 100      CLE            CL1
## 4               Hamilton Field         NA 101 107      KEK            FW1
## 5     Union Grounds (Brooklyn)         NA  90  88      NYU            NY2
## 6     Jefferson Street Grounds         NA 102  98      ATH            PH1
##   teamIDretro
## 1         BS1
## 2         CH1
## 3         CL1
## 4         FW1
## 5         NY2
```

```
## 6         PH1
```

```
head(Salaries)
```

```
##   yearID teamID lgID  playerID salary
## 1   1985    ATL   NL barkele01 870000
## 2   1985    ATL   NL bedrost01 550000
## 3   1985    ATL   NL benedbr01 545000
## 4   1985    ATL   NL  campri01 633333
## 5   1985    ATL   NL ceronri01 625000
## 6   1985    ATL   NL chambch01 800000
```

2. Find all the names of the players who have ever had a stint (from the Fielding data frame) in the Red Sox (or the Boston Americans) in years where the team made it to the World Series (so they won their leagues) There should be 13 years. Note the World Series was not played each year and began in 1903 and there should be two teams for

each year it was played.

Show the only the first ten names (arranged in alphabetical order of last name). Your output should look like this:

```
world_series_years <- Teams %>%
  filter(teamID == "BOS", WSWin == "Y") %>%
  pull(yearID)

filtered_fielding <- Fielding %>%
  filter(teamID %in% c("BOS", "BOS"), yearID %in% world_series_years)

joined_data <- filtered_fielding %>%
  left_join(People, by = "playerID")

sorted_data <- joined_data %>%
  arrange(nameLast, nameFirst)


head(select(sorted_data, nameFirst, nameLast, yearID), 10)
```

```
##    nameFirst nameLast yearID
## 1    Alfredo   Aceves   2013
## 2      Terry    Adams   2004
## 3        Sam    Agnew   1916
## 4        Sam    Agnew   1918
## 5       Nick  Altrock   1903
## 6        Abe  Alvarez   2004
## 7      Jimmy Anderson   2004
## 8    Bronson   Arroyo   2004
## 9      Pedro  Astacio   2004
## 10      Lore    Bader   1918
```

3. Some players play on multiple teams each year.

a. Construct a data frame containing the total salary for each player for each year. Show the number of rows - should be 26,323

```
total_salary_per_player_per_year <- Salaries %>%
  group_by(playerID, yearID) %>%
  summarise(total_salary = sum(salary, na.rm = TRUE), .groups = 'drop')
glimpse(total_salary_per_player_per_year)
```

```
## Rows: 26,323
## Columns: 3
## $ playerID     <chr> "aardsda01", "aardsda01", "aardsda01", "aardsda01", "aard~
## $ yearID       <int> 2004, 2007, 2008, 2009, 2010, 2011, 2012, 1986, 1987, 198~
## $ total_salary <int> 300000, 387500, 403250, 419000, 2750000, 4500000, 500000,~
```

b. Construct a second data frame containing columns with the total number of at bats and total number of hits for each player for each year. Show the number of rows - should be 100,690.

```
total_at_bats_and_hits <- Batting %>%
  group_by(playerID, yearID) %>%
  summarise(total_AB = sum(AB, na.rm = TRUE),
            total_hits = sum(H, na.rm = TRUE),
            .groups = ('drop'))

glimpse(total_at_bats_and_hits)
```

```
## Rows: 103,693
## Columns: 4
## $ playerID   <chr> "aardsda01", "aardsda01", "aardsda01", "aardsda01", "aardsd~
## $ yearID     <int> 2004, 2006, 2007, 2008, 2009, 2010, 2012, 2013, 2015, 1954,~
## $ total_AB   <int> 0, 2, 0, 1, 0, 0, 0, 0, 1, 468, 602, 609, 615, 601, 629, 59~
## $ total_hits <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 131, 189, 200, 198, 196, 223, 17~
```

4. The batting average of a player is the number of Hits divided by the number of at bats (a larger value is is better).

Using the data frames you created in part 3, create a new data frame with batting average and salary information for only players who had a minimum of 400 at bats in the years from 1985 on (when salary information started being collected). Eliminate any rows with no batting or no salary information. Show the number of rows. Should be 5,345.

```
batting_avg_data <- total_at_bats_and_hits %>%
  mutate(batting_average = total_hits / total_AB)
```

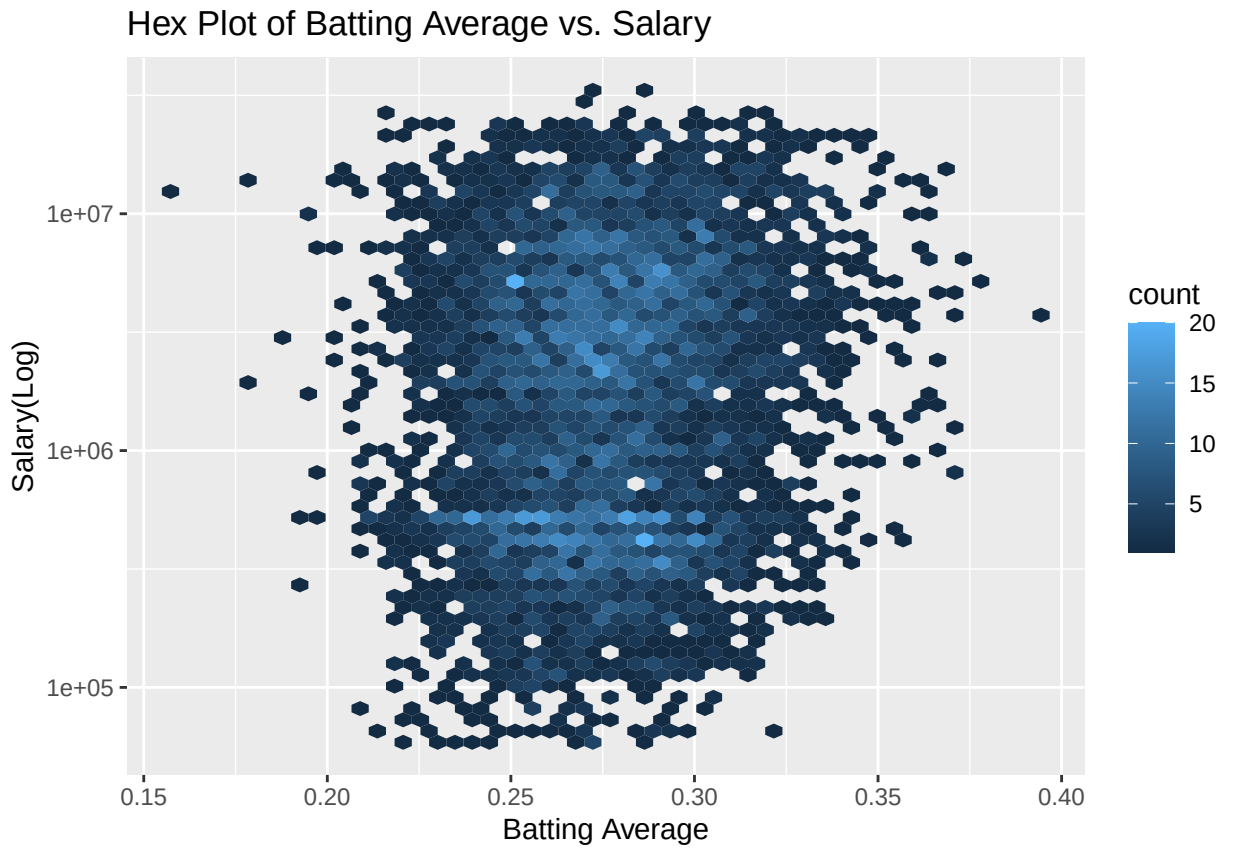
```r
merged_data <- left_join(batting_avg_data, total_salary_per_player_per_year, by = c("playerID", "yearID
```

```r
filtered_data <- merged_data %>%
  filter(total_AB >= 400)

filtered_data <- filtered_data %>%
  filter(!is.na(batting_average) & !is.na(total_salary))

glimpse(filtered_data)
```

```
## Rows: 5,345
## Columns: 6
## $ playerID        <chr> "abbotku01", "abernbr01", "abreubo01", "abreubo01", "a~
## $ yearID          <int> 1995, 2002, 1998, 1999, 2000, 2001, 2002, 2003, 2004, ~
## $ total_AB        <int> 420, 463, 497, 546, 576, 588, 572, 577, 574, 588, 548,~
## $ total_hits      <int> 107, 112, 155, 183, 182, 170, 176, 173, 173, 168, 163,~
## $ batting_average <dbl> 0.2547619, 0.2419006, 0.3118712, 0.3351648, 0.3159722,~
## $ total_salary    <int> 119000, 215000, 180000, 400000, 2933333, 4983000, 6333~
```

b. Use a hex plot to explore the association between a player's batting average (x axis) and their salary (y axis). Use a log scale for salary. Interpret the plot

```r
ggplot(filtered_data, aes(x = batting_average, y = total_salary)) + geom_hex(bins = 50) +
  scale_y_log10() + labs(title = "Hex Plot of Batting Average vs. Salary", x = "Batting Average", y = "
```




```
paste("According to the graph, it seems like there isnt much of a correlation between batting average an
```

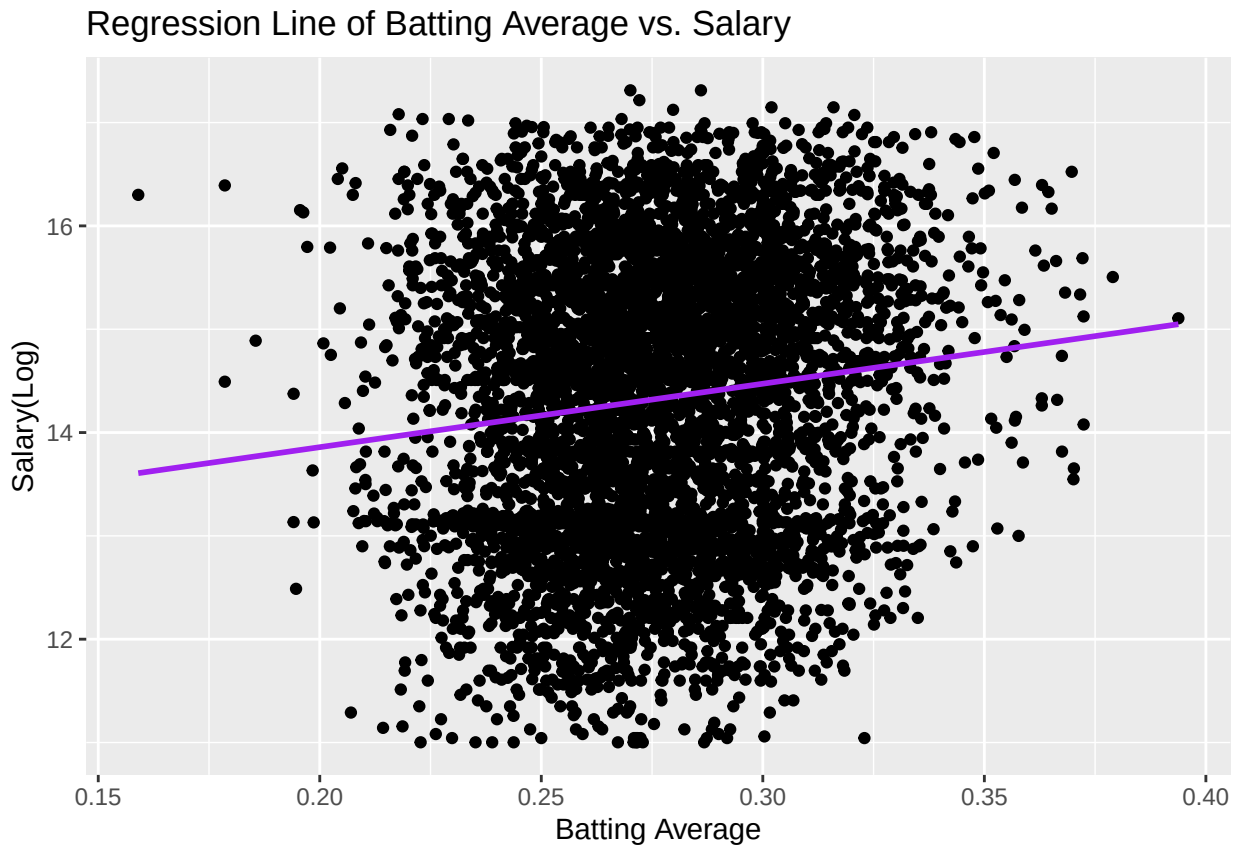
```
## [1] "According to the graph, it seems like there isnt much of a correlation between batting average a
```

c. Use a single (not faceted) plot of just the Ordinary Least Squares lines for batting average and salary to explore if this association has changed over time. Use a log scale for salary. Interpret the plot

```
lm_model <- lm(log(total_salary) ~ batting_average, data = filtered_data)
```

```
ggplot(filtered_data, aes(x = batting_average, y = log(total_salary))) + geom_point() + geom_smooth(meth
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
paste("Compared to the other graph that I made. The regression line shows that there is a slight positiv
```

```
## [1] "Compared to the other graph that I made. The regression line shows that there is a slight positi
```

5. Find the salary of all players with first name "John" in even numbered years after 1985. Show only the first ten values arranged in descending order of salary. Your output should look like this:

```
john_salary_data <- total_salary_per_player_per_year %>%
  inner_join(People, by = "playerID") %>%
  filter(nameFirst == "John" & yearID > 1985 & yearID %% 2 == 0) %>%
  select(yearID, nameFirst, nameLast, total_salary) %>%
  arrange(desc(total_salary))

head(john_salary_data, 10)
```

```
## # A tibble: 10 x 4
##    yearID nameFirst nameLast total_salary
##     <int> <chr>     <chr>           <int>
##  1   2010 John      Lackey       18700000
##  2   2016 John      Lackey       16000000
##  3   2012 John      Lackey       15950000
```

```
##  4   2016 John      Danks        15750000
##  5   2014 John      Lackey       15250000
##  6   2014 John      Danks        14250000
##  7   2008 John      Smoltz       14000000
##  8   2004 John      Smoltz       11666667
##  9   2006 John      Smoltz       11000000
## 10   2000 John      Smoltz        8500000
```