ds4

Matthew Robinson

library(ggplot2)  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ lubridate 1.9.3 ✔ tibble 3.2.1  
## ✔ purrr 1.0.2 ✔ tidyr 1.3.1  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggstance)

##   
## Attaching package: 'ggstance'  
##   
## The following objects are masked from 'package:ggplot2':  
##   
## geom\_errorbarh, GeomErrorbarh

library(lvplot)  
library(ggbeeswarm)  
library(lubridate)  
library(nycflights13)  
library(maps)

##   
## Attaching package: 'maps'  
##   
## The following object is masked from 'package:purrr':  
##   
## map

# Section 10.5

# Exercise 1

How can you tell if an object is a tibble?

flights

## # A tibble: 336,776 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ℹ 336,766 more rows  
## # ℹ 11 more variables: arr\_delay <dbl>, carrier <chr>, flight <int>,  
## # tailnum <chr>, origin <chr>, dest <chr>, air\_time <dbl>, distance <dbl>,  
## # hour <dbl>, minute <dbl>, time\_hour <dttm>

As seen above, a tibble will say that it is a tibble at the top when it is printed.

mtcars

## mpg cyl disp hp drat wt qsec vs am gear carb  
## Mazda RX4 21.0 6 160.0 110 3.90 2.620 16.46 0 1 4 4  
## Mazda RX4 Wag 21.0 6 160.0 110 3.90 2.875 17.02 0 1 4 4  
## Datsun 710 22.8 4 108.0 93 3.85 2.320 18.61 1 1 4 1  
## Hornet 4 Drive 21.4 6 258.0 110 3.08 3.215 19.44 1 0 3 1  
## Hornet Sportabout 18.7 8 360.0 175 3.15 3.440 17.02 0 0 3 2  
## Valiant 18.1 6 225.0 105 2.76 3.460 20.22 1 0 3 1  
## Duster 360 14.3 8 360.0 245 3.21 3.570 15.84 0 0 3 4  
## Merc 240D 24.4 4 146.7 62 3.69 3.190 20.00 1 0 4 2  
## Merc 230 22.8 4 140.8 95 3.92 3.150 22.90 1 0 4 2  
## Merc 280 19.2 6 167.6 123 3.92 3.440 18.30 1 0 4 4  
## Merc 280C 17.8 6 167.6 123 3.92 3.440 18.90 1 0 4 4  
## Merc 450SE 16.4 8 275.8 180 3.07 4.070 17.40 0 0 3 3  
## Merc 450SL 17.3 8 275.8 180 3.07 3.730 17.60 0 0 3 3  
## Merc 450SLC 15.2 8 275.8 180 3.07 3.780 18.00 0 0 3 3  
## Cadillac Fleetwood 10.4 8 472.0 205 2.93 5.250 17.98 0 0 3 4  
## Lincoln Continental 10.4 8 460.0 215 3.00 5.424 17.82 0 0 3 4  
## Chrysler Imperial 14.7 8 440.0 230 3.23 5.345 17.42 0 0 3 4  
## Fiat 128 32.4 4 78.7 66 4.08 2.200 19.47 1 1 4 1  
## Honda Civic 30.4 4 75.7 52 4.93 1.615 18.52 1 1 4 2  
## Toyota Corolla 33.9 4 71.1 65 4.22 1.835 19.90 1 1 4 1  
## Toyota Corona 21.5 4 120.1 97 3.70 2.465 20.01 1 0 3 1  
## Dodge Challenger 15.5 8 318.0 150 2.76 3.520 16.87 0 0 3 2  
## AMC Javelin 15.2 8 304.0 150 3.15 3.435 17.30 0 0 3 2  
## Camaro Z28 13.3 8 350.0 245 3.73 3.840 15.41 0 0 3 4  
## Pontiac Firebird 19.2 8 400.0 175 3.08 3.845 17.05 0 0 3 2  
## Fiat X1-9 27.3 4 79.0 66 4.08 1.935 18.90 1 1 4 1  
## Porsche 914-2 26.0 4 120.3 91 4.43 2.140 16.70 0 1 5 2  
## Lotus Europa 30.4 4 95.1 113 3.77 1.513 16.90 1 1 5 2  
## Ford Pantera L 15.8 8 351.0 264 4.22 3.170 14.50 0 1 5 4  
## Ferrari Dino 19.7 6 145.0 175 3.62 2.770 15.50 0 1 5 6  
## Maserati Bora 15.0 8 301.0 335 3.54 3.570 14.60 0 1 5 8  
## Volvo 142E 21.4 4 121.0 109 4.11 2.780 18.60 1 1 4 2

You can tell this is not a tibble because it does not say it is a tibble.

# Exercise 2

Compare and contrast the following operations on a data.frame and equivalent tibble. What is different? Why might the default data frame behaviours cause you frustration?

df <- data.frame(abc = 1, xyz = "a")  
df$x

## [1] "a"

df[, "xyz"]

## [1] "a"

df[, c("abc", "xyz")]

## abc xyz  
## 1 1 a

These operations are performed on a data frame. The first operation incorrectly returns something despite there being no column named x. The second operation returns the correct data but does not specify what row and column it is in. The third operation returns the correct data but does not specify what data type each column is.

df <- tibble(abc = 1, xyz = "a")  
df$x

## Warning: Unknown or uninitialised column: `x`.

## NULL

df[, "xyz"]

## # A tibble: 1 × 1  
## xyz   
## <chr>  
## 1 a

df[, c("abc", "xyz")]

## # A tibble: 1 × 2  
## abc xyz   
## <dbl> <chr>  
## 1 1 a

These operations are performed on a tibble. The first operation correctly points out that there is no column named x. The second operation returns the correct data with the relevant information. The third operation also returns the correct data with the relevant information.

# Exercise 5

What does tibble::enframe() do? When might you use it?

enframe(1:5)

## # A tibble: 5 × 2  
## name value  
## <int> <int>  
## 1 1 1  
## 2 2 2  
## 3 3 3  
## 4 4 4  
## 5 5 5

The enframe command converts a vector into a tibble. This can be used if you have data in a vector which you want in a tibble.

# Exercise 6

What option controls how many additional column names are printed at the footer of a tibble?

flights

## # A tibble: 336,776 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ℹ 336,766 more rows  
## # ℹ 11 more variables: arr\_delay <dbl>, carrier <chr>, flight <int>,  
## # tailnum <chr>, origin <chr>, dest <chr>, air\_time <dbl>, distance <dbl>,  
## # hour <dbl>, minute <dbl>, time\_hour <dttm>

By default, the flights tibble lists 11 columns in the footer which do not appear in the table.

flights %>% print(width = Inf)

## # A tibble: 336,776 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745

## arr\_delay carrier flight tailnum origin dest air\_time distance hour minute  
## <dbl> <chr> <int> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 11 UA 1545 N14228 EWR IAH 227 1400 5 15  
## 2 20 UA 1714 N24211 LGA IAH 227 1416 5 29  
## 3 33 AA 1141 N619AA JFK MIA 160 1089 5 40  
## 4 -18 B6 725 N804JB JFK BQN 183 1576 5 45  
## 5 -25 DL 461 N668DN LGA ATL 116 762 6 0  
## 6 12 UA 1696 N39463 EWR ORD 150 719 5 58  
## 7 19 B6 507 N516JB EWR FLL 158 1065 6 0  
## 8 -14 EV 5708 N829AS LGA IAD 53 229 6 0  
## 9 -8 B6 79 N593JB JFK MCO 140 944 6 0  
## 10 8 AA 301 N3ALAA LGA ORD 138 733 6 0  
## time\_hour   
## <dttm>   
## 1 2013-01-01 05:00:00  
## 2 2013-01-01 05:00:00  
## 3 2013-01-01 05:00:00  
## 4 2013-01-01 05:00:00  
## 5 2013-01-01 06:00:00  
## 6 2013-01-01 05:00:00  
## 7 2013-01-01 06:00:00  
## 8 2013-01-01 06:00:00  
## 9 2013-01-01 06:00:00  
## 10 2013-01-01 06:00:00  
## # ℹ 336,766 more rows

Changing the width parameter changes how many columns are included in the table and, by extension, how many columns are listed in the footer.

# Section 12.2.1

# Exercise 1

Using prose, describe how the variables and observations are organised in each of the sample tables.

table1

## # A tibble: 6 × 4  
## country year cases population  
## <chr> <dbl> <dbl> <dbl>  
## 1 Afghanistan 1999 745 19987071  
## 2 Afghanistan 2000 2666 20595360  
## 3 Brazil 1999 37737 172006362  
## 4 Brazil 2000 80488 174504898  
## 5 China 1999 212258 1272915272  
## 6 China 2000 213766 1280428583

Table 1 is the tidy table. Each variable, including country, year, cases, and population, has its own column. Each observation, which is a combination of a country and year, has its own row. Therefore, each value has its own cell.

table2

## # A tibble: 12 × 4  
## country year type count  
## <chr> <dbl> <chr> <dbl>  
## 1 Afghanistan 1999 cases 745  
## 2 Afghanistan 1999 population 19987071  
## 3 Afghanistan 2000 cases 2666  
## 4 Afghanistan 2000 population 20595360  
## 5 Brazil 1999 cases 37737  
## 6 Brazil 1999 population 172006362  
## 7 Brazil 2000 cases 80488  
## 8 Brazil 2000 population 174504898  
## 9 China 1999 cases 212258  
## 10 China 1999 population 1272915272  
## 11 China 2000 cases 213766  
## 12 China 2000 population 1280428583

Table 2 is not tidy. Cases and population form different rows for each combination of a country and year when they should be two separate columns.

table3

## # A tibble: 6 × 3  
## country year rate   
## <chr> <dbl> <chr>   
## 1 Afghanistan 1999 745/19987071   
## 2 Afghanistan 2000 2666/20595360   
## 3 Brazil 1999 37737/172006362   
## 4 Brazil 2000 80488/174504898   
## 5 China 1999 212258/1272915272  
## 6 China 2000 213766/1280428583

Table 3 is not tidy. The rate column includes both cases and population when they should be in separate columns.

table4a

## # A tibble: 3 × 3  
## country `1999` `2000`  
## <chr> <dbl> <dbl>  
## 1 Afghanistan 745 2666  
## 2 Brazil 37737 80488  
## 3 China 212258 213766

table4b

## # A tibble: 3 × 3  
## country `1999` `2000`  
## <chr> <dbl> <dbl>  
## 1 Afghanistan 19987071 20595360  
## 2 Brazil 172006362 174504898  
## 3 China 1272915272 1280428583

These tables are not tidy. Table 4a only includes cases while table 4b only includes population when they should be combined into one table.

# Section 12.3.3

# Exercise 2

Why does this code fail?

table4a %>% pivot\_longer(c(1999, 2000), names\_to = “year”, values\_to = “cases”)  
#> Error in pivot\_longer():  
#> ! Can’t subset columns past the end.  
#> ℹ Locations 1999 and 2000 don’t exist.  
#> ℹ There are only 3 columns.

This code fails because 1999 and 2000 are considered characters instead of integers in the table.

table4a %>% pivot\_longer(c("1999", "2000"), names\_to = "year", values\_to = "cases")

## # A tibble: 6 × 3  
## country year cases  
## <chr> <chr> <dbl>  
## 1 Afghanistan 1999 745  
## 2 Afghanistan 2000 2666  
## 3 Brazil 1999 37737  
## 4 Brazil 2000 80488  
## 5 China 1999 212258  
## 6 China 2000 213766

Putting quotation marks around 1999 and 2000 fixes the issue.

# Section 12.4.3

# Exercise 1

What do the extra and fill arguments do in separate()? Experiment with the various options for the following two toy datasets.

tibble(x = c("a,b,c", "d,e,f,g", "h,i,j")) %>% separate(x, c("one", "two", "three"))

## Warning: Expected 3 pieces. Additional pieces discarded in 1 rows [2].

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 d e f   
## 3 h i j

This function gives a warning because the second row includes four elements instead of three. Therefore, the last element of that row is dropped.

tibble(x = c("a,b,c", "d,e,f,g", "h,i,j")) %>% separate(x, c("one", "two", "three"), extra = "drop")

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 d e f   
## 3 h i j

Setting extra to “drop” gives the same results as the default settings. However, it removes the warning.

tibble(x = c("a,b,c", "d,e,f,g", "h,i,j")) %>% separate(x, c("one", "two", "three"), extra = "merge")

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 d e f,g   
## 3 h i j

Setting extra to “merge” puts the last two elements in the second row in the same column. This results in f and g sharing a cell.

tibble(x = c("a,b,c", "d,e", "f,g,i")) %>% separate(x, c("one", "two", "three"))

## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 1 rows [2].

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 d e <NA>   
## 3 f g i

This function gives a warning because the second row includes two elements instead of three. Therefore, the last element of that row is listed as NA.

tibble(x = c("a,b,c", "d,e", "f,g,i")) %>% separate(x, c("one", "two", "three"), fill = "right")

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 d e <NA>   
## 3 f g i

Setting fill to “right” gives the same results as the default settings. However, it removes the warning.

tibble(x = c("a,b,c", "d,e", "f,g,i")) %>% separate(x, c("one", "two", "three"), fill = "left")

## # A tibble: 3 × 3  
## one two three  
## <chr> <chr> <chr>  
## 1 a b c   
## 2 <NA> d e   
## 3 f g i

Setting fill to “left” skips the first element in the second row instead of the last one. This results in the element of that row being listed as NA.

# Section 12.6.1

# Exercise 2

What happens if you neglect the mutate() step? (mutate(names\_from = stringr::str\_replace(key, “newrel”, “new\_rel”)))

who %>% pivot\_longer(cols = new\_sp\_m014:newrel\_f65, names\_to = "key", values\_to = "cases", values\_drop\_na = TRUE) %>% separate(key, c("new", "var", "sexage")) %>% select(-new, -iso2, -iso3) %>% separate(sexage, c("sex", "age"), sep = 1)

## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 2580 rows [243, 244, 679,  
## 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 903, 904, 905,  
## 906, ...].

## # A tibble: 76,046 × 6  
## country year var sex age cases  
## <chr> <dbl> <chr> <chr> <chr> <dbl>  
## 1 Afghanistan 1997 sp m 014 0  
## 2 Afghanistan 1997 sp m 1524 10  
## 3 Afghanistan 1997 sp m 2534 6  
## 4 Afghanistan 1997 sp m 3544 3  
## 5 Afghanistan 1997 sp m 4554 5  
## 6 Afghanistan 1997 sp m 5564 2  
## 7 Afghanistan 1997 sp m 65 0  
## 8 Afghanistan 1997 sp f 014 5  
## 9 Afghanistan 1997 sp f 1524 38  
## 10 Afghanistan 1997 sp f 2534 36  
## # ℹ 76,036 more rows

Removing the mutate function gives a warning that several pieces are missing.

who %>% pivot\_longer(cols = new\_sp\_m014:newrel\_f65, names\_to = "key", values\_to = "cases", values\_drop\_na = TRUE) %>% mutate(key = stringr::str\_replace(key, "newrel", "new\_rel")) %>% separate(key, c("new", "var", "sexage")) %>% select(-new, -iso2, -iso3) %>% separate(sexage, c("sex", "age"), sep = 1)

## # A tibble: 76,046 × 6  
## country year var sex age cases  
## <chr> <dbl> <chr> <chr> <chr> <dbl>  
## 1 Afghanistan 1997 sp m 014 0  
## 2 Afghanistan 1997 sp m 1524 10  
## 3 Afghanistan 1997 sp m 2534 6  
## 4 Afghanistan 1997 sp m 3544 3  
## 5 Afghanistan 1997 sp m 4554 5  
## 6 Afghanistan 1997 sp m 5564 2  
## 7 Afghanistan 1997 sp m 65 0  
## 8 Afghanistan 1997 sp f 014 5  
## 9 Afghanistan 1997 sp f 1524 38  
## 10 Afghanistan 1997 sp f 2534 36  
## # ℹ 76,036 more rows

Adding the mutate function back in removes this warning.

# Section 13.4.6

# Exercise 2

Add the location of the origin and destination (i.e. the lat and lon) to flights.

flights %>% left\_join(airports, c("origin" = "faa")) %>% left\_join(airports, c("dest" = "faa"))

## # A tibble: 336,776 × 33  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 1 517 515 2 830 819  
## 2 2013 1 1 533 529 4 850 830  
## 3 2013 1 1 542 540 2 923 850  
## 4 2013 1 1 544 545 -1 1004 1022  
## 5 2013 1 1 554 600 -6 812 837  
## 6 2013 1 1 554 558 -4 740 728  
## 7 2013 1 1 555 600 -5 913 854  
## 8 2013 1 1 557 600 -3 709 723  
## 9 2013 1 1 557 600 -3 838 846  
## 10 2013 1 1 558 600 -2 753 745  
## # ℹ 336,766 more rows  
## # ℹ 25 more variables: arr\_delay <dbl>, carrier <chr>, flight <int>,  
## # tailnum <chr>, origin <chr>, dest <chr>, air\_time <dbl>, distance <dbl>,  
## # hour <dbl>, minute <dbl>, time\_hour <dttm>, name.x <chr>, lat.x <dbl>,  
## # lon.x <dbl>, alt.x <dbl>, tz.x <dbl>, dst.x <chr>, tzone.x <chr>,  
## # name.y <chr>, lat.y <dbl>, lon.y <dbl>, alt.y <dbl>, tz.y <dbl>,  
## # dst.y <chr>, tzone.y <chr>

Using two left joins to map both “origin” and “dest” to “faa” adds the location of both origin and destination to flights.

# Section 15.4.1

# Exercise 1

arrange(gss\_cat, desc(tvhours))

## # A tibble: 21,483 × 9  
## year marital age race rincome partyid relig denom tvhours  
## <int> <fct> <int> <fct> <fct> <fct> <fct> <fct> <int>  
## 1 2000 Never married 30 Black Not applicable Independe… Prot… Bapt… 24  
## 2 2000 Separated 45 Black Not applicable Ind,near … Prot… Other 24  
## 3 2002 Never married 33 White $6000 to 6999 Independe… Cath… Not … 24  
## 4 2006 Divorced 53 Black Not applicable Strong de… Prot… Sout… 24  
## 5 2008 Divorced 50 Black No answer Ind,near … Prot… Bapt… 24  
## 6 2008 Never married 44 White Not applicable Independe… Prot… Other 24  
## 7 2008 Never married 21 White Don't know Independe… Cath… Not … 24  
## 8 2008 Widowed 71 White Not applicable Strong de… Prot… Don'… 24  
## 9 2010 Widowed 62 Black Not applicable Strong de… Prot… Am b… 24  
## 10 2010 Widowed 52 Black Refused Strong de… Prot… Bapt… 24  
## # ℹ 21,473 more rows

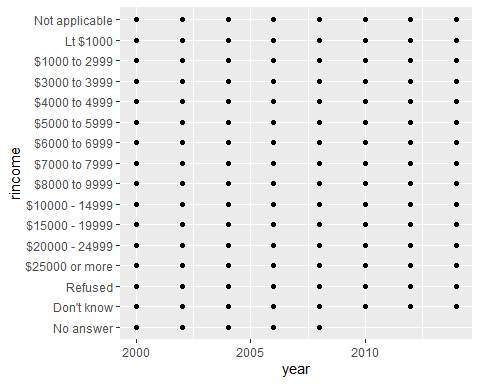
Several people said they watch television 24 hours a day which is clearly not realistic. It would probably be better to use median instead of mean because median mostly ignores these unrealistic numbers.

# Section 15.5.1

# Exercise 2

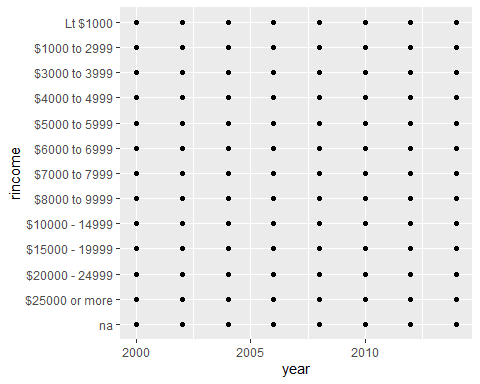
How could you collapse rincome into a small set of categories?

ggplot(data = gss\_cat) + geom\_point(mapping = aes(x = year, y = rincome))



The categories “Not applicable”, “Refused”, “Don’t know”, and “No answer” all basically mean not applicable.

ggplot(data = gss\_cat %>% mutate(rincome = fct\_collapse(rincome, na = c("Not applicable", "Refused", "Don't know", "No answer")))) + geom\_point(mapping = aes(x = year, y = rincome))



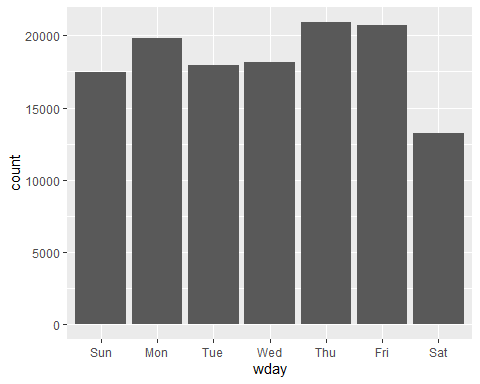
Combining these four categories into one category named “na” makes the data tidier.

# Section 16.3.4

# Exercise 5

On what day of the week should you leave if you want to minimise the chance of a delay?

make\_datetime\_100 <- function(year, month, day, time) {make\_datetime(year, month, day, time %/% 100, time %% 100)}  
flights\_dt <- flights %>% filter(!is.na(dep\_time), !is.na(arr\_time)) %>% mutate(dep\_time = make\_datetime\_100(year, month, day, dep\_time), arr\_time = make\_datetime\_100(year, month, day, arr\_time), sched\_dep\_time = make\_datetime\_100(year, month, day, sched\_dep\_time), sched\_arr\_time = make\_datetime\_100(year, month, day, sched\_arr\_time)) %>% select(origin, dest, ends\_with("delay"), ends\_with("time"))  
filter(flights\_dt, dep\_delay > 0) %>% mutate(wday = wday(dep\_time, label = TRUE)) %>% ggplot(aes(x = wday)) + geom\_bar()



If you want to avoid delays, you should leave on Saturday because it has the least amount of delays.

# Section 16.4.5

# Exercise 1

Why is there months() but no dmonths()?

dseconds(1)

## [1] "1s"

dminutes(1)

## [1] "60s (~1 minutes)"

dhours(1)

## [1] "3600s (~1 hours)"

ddays(1)

## [1] "86400s (~1 days)"

dweeks(1)

## [1] "604800s (~1 weeks)"

dyears(1)

## [1] "31557600s (~1 years)"

Unlike seconds, minutes, hours, days, weeks, and years, months do not have a set length. A month could be anywhere from 28 to 31 days.

# Exercise 2

Explain days(overnight \* 1) to someone who has just started learning R. How does it work?

flights\_dt %>% mutate(overnight = arr\_time < dep\_time, arr\_time = arr\_time + days(overnight \* 1), sched\_arr\_time = sched\_arr\_time + days(overnight \* 1))

## # A tibble: 328,063 × 10  
## origin dest dep\_delay arr\_delay dep\_time sched\_dep\_time   
## <chr> <chr> <dbl> <dbl> <dttm> <dttm>   
## 1 EWR IAH 2 11 2013-01-01 05:17:00 2013-01-01 05:15:00  
## 2 LGA IAH 4 20 2013-01-01 05:33:00 2013-01-01 05:29:00  
## 3 JFK MIA 2 33 2013-01-01 05:42:00 2013-01-01 05:40:00  
## 4 JFK BQN -1 -18 2013-01-01 05:44:00 2013-01-01 05:45:00  
## 5 LGA ATL -6 -25 2013-01-01 05:54:00 2013-01-01 06:00:00  
## 6 EWR ORD -4 12 2013-01-01 05:54:00 2013-01-01 05:58:00  
## 7 EWR FLL -5 19 2013-01-01 05:55:00 2013-01-01 06:00:00  
## 8 LGA IAD -3 -14 2013-01-01 05:57:00 2013-01-01 06:00:00  
## 9 JFK MCO -3 -8 2013-01-01 05:57:00 2013-01-01 06:00:00  
## 10 LGA ORD -2 8 2013-01-01 05:58:00 2013-01-01 06:00:00  
## # ℹ 328,053 more rows  
## # ℹ 4 more variables: arr\_time <dttm>, sched\_arr\_time <dttm>, air\_time <dbl>,  
## # overnight <lgl>

The overnight variable will equal 1 if it is an overnight flight and 0 if it is not an overnight flight. Therefore, the “days(overnight \* 1)” command will return 1 if it is an overnight flight and 0 if it is not an overnight flight. This ensures that a day is added only to overnight flights.

# Exercise 4

Write a function that given your birthday (as a date), returns how old you are in years.

as.duration((today() - dmy(08052002))) %/% dyears(1)

## [1] 22

This function correctly returns my age as 22.

# FBS Schools

This data set contains data on all NCAA Division 1 Football Bowl Subdivision (FBS) schools during the 2024 college football season. The data set was downloaded from collegefootballdata.com as a csv file. The read\_csv function is used to transfer this data set into a data frame. The following table shows the first ten FBS schools alphabetically.

fbs <- read\_csv("fbs1.csv", show\_col\_types = FALSE)  
fbs %>% print(width = Inf)

## # A tibble: 134 × 28  
## Id School Mascot Abbreviation `Alt Name1` `Alt Name2`  
## <dbl> <chr> <chr> <chr> <chr> <chr>   
## 1 2005 Air Force Falcons AFA <NA> AFA   
## 2 2006 Akron Zips AKR <NA> AKR   
## 3 333 Alabama Crimson Tide ALA <NA> ALA   
## 4 2026 App State Mountaineers APP Appalachian State APP   
## 5 12 Arizona Wildcats ARIZ <NA> ARIZ   
## 6 9 Arizona State Sun Devils ASU <NA> ASU   
## 7 8 Arkansas Razorbacks ARK <NA> ARK   
## 8 2032 Arkansas State Red Wolves ARST <NA> ARST   
## 9 349 Army Black Knights ARMY <NA> ARMY   
## 10 2 Auburn Tigers AUB <NA> AUB   
## `Alt Name3` Conference Division Color `Alt Color`  
## <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Mountain West <NA> #004a7b #ffffff   
## 2 Akron Mid-American <NA> #00285e #84754e   
## 3 Alabama SEC <NA> #9e1632 #ffffff   
## 4 App State Sun Belt East #ffcc00 #222222   
## 5 Arizona Big 12 <NA> #0c234b #ab0520   
## 6 Arizona St Big 12 <NA> #8e0c3a #ffc72c   
## 7 Arkansas SEC <NA> #a41f35 #ffffff   
## 8 Arkansas St Sun Belt West #e81018 #000000   
## 9 Army American Athletic <NA> #ce9c00 #231f20   
## 10 Auburn SEC <NA> #002b5c #f26522   
## `Logos[0]`   
## <chr>   
## 1 http://a.espncdn.com/i/teamlogos/ncaa/500/2005.png  
## 2 http://a.espncdn.com/i/teamlogos/ncaa/500/2006.png  
## 3 http://a.espncdn.com/i/teamlogos/ncaa/500/333.png   
## 4 http://a.espncdn.com/i/teamlogos/ncaa/500/2026.png  
## 5 http://a.espncdn.com/i/teamlogos/ncaa/500/12.png   
## 6 http://a.espncdn.com/i/teamlogos/ncaa/500/9.png   
## 7 http://a.espncdn.com/i/teamlogos/ncaa/500/8.png   
## 8 http://a.espncdn.com/i/teamlogos/ncaa/500/2032.png  
## 9 http://a.espncdn.com/i/teamlogos/ncaa/500/349.png   
## 10 http://a.espncdn.com/i/teamlogos/ncaa/500/2.png

## `Logos[1]` Twitter   
## <chr> <chr>   
## 1 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2005.png @AF\_Football   
## 2 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2006.png @ZipsFB   
## 3 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/333.png @AlabamaFTBL   
## 4 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2026.png @AppState\_FB   
## 5 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/12.png @ArizonaFBall   
## 6 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/9.png @ASUFootball   
## 7 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/8.png @RazorbackFB   
## 8 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2032.png @AStateFB   
## 9 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/349.png @ArmyWP\_Football  
## 10 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2.png @AuburnFootball   
## `Location Venue Id` `Location Name` `Location City`   
## <dbl> <chr> <chr>   
## 1 3713 Falcon Stadium Colorado Springs  
## 2 3768 InfoCision Stadium Akron   
## 3 3657 Bryant-Denny Stadium Tuscaloosa   
## 4 3792 Kidd Brewer Stadium Boone   
## 5 3619 Arizona Stadium Tucson   
## 6 3947 Mountain America Stadium Tempe   
## 7 3887 Razorback Stadium Fayetteville   
## 8 3625 Centennial Bank Stadium Jonesboro   
## 9 3841 Michie Stadium West Point   
## 10 3785 Jordan-Hare Stadium Auburn   
## `Location State` `Location Zip` `Location Country Code` `Location Timezone`  
## <chr> <dbl> <chr> <chr>   
## 1 CO 80840 US America/Denver   
## 2 OH 44399 US America/New\_York   
## 3 AL 35487 US America/Chicago   
## 4 NC 28608 US America/New\_York   
## 5 AZ 85721 US America/Phoenix   
## 6 AZ 85287 US America/Phoenix   
## 7 AR 72702 US America/Chicago   
## 8 AR 72404 US America/Chicago   
## 9 NY 10996 US America/New\_York   
## 10 AL 36849 US America/Chicago

## `Location Latitude` `Location Longitude` `Location Elevation`  
## <dbl> <dbl> <dbl>  
## 1 39.0 -105. 2025.   
## 2 41.1 -81.5 321.   
## 3 33.2 -87.6 70.1  
## 4 36.2 -81.7 991.   
## 5 32.2 -111. 742.   
## 6 33.4 -112. 360.   
## 7 36.1 -94.2 404.   
## 8 35.8 -90.7 89.5  
## 9 41.4 -74.0 99.2  
## 10 32.6 -85.5 201.   
## `Location Capacity` `Location Year Constructed` `Location Grass`  
## <dbl> <dbl> <lgl>   
## 1 46692 1962 FALSE   
## 2 30000 2009 FALSE   
## 3 101821 1929 TRUE   
## 4 30000 1962 FALSE   
## 5 50782 1928 FALSE   
## 6 65870 1958 TRUE   
## 7 80000 1938 FALSE   
## 8 30382 1974 FALSE   
## 9 38000 1924 FALSE   
## 10 87451 1939 TRUE   
## `Location Dome`  
## <lgl>   
## 1 FALSE   
## 2 FALSE   
## 3 FALSE   
## 4 FALSE   
## 5 FALSE   
## 6 FALSE   
## 7 FALSE   
## 8 FALSE   
## 9 FALSE   
## 10 FALSE   
## # ℹ 124 more rows

When looking at this data, it is clear that it is not tidy. A few things will need to be done to tidy up this data. First of all, you will notice that each school has a unique ID. This ID ties each school into a larger database. Each school’s stadium also has a unique ID for the same purpose. Since we are only looking at this data set, it is safe to remove these two columns.

fbs <- select(fbs, -c("Id", "Location Venue Id"))  
fbs %>% print(width = Inf)

## # A tibble: 134 × 26  
## School Mascot Abbreviation `Alt Name1` `Alt Name2`  
## <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA <NA> AFA   
## 2 Akron Zips AKR <NA> AKR   
## 3 Alabama Crimson Tide ALA <NA> ALA   
## 4 App State Mountaineers APP Appalachian State APP   
## 5 Arizona Wildcats ARIZ <NA> ARIZ   
## 6 Arizona State Sun Devils ASU <NA> ASU   
## 7 Arkansas Razorbacks ARK <NA> ARK   
## 8 Arkansas State Red Wolves ARST <NA> ARST   
## 9 Army Black Knights ARMY <NA> ARMY   
## 10 Auburn Tigers AUB <NA> AUB   
## `Alt Name3` Conference Division Color `Alt Color`  
## <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Mountain West <NA> #004a7b #ffffff   
## 2 Akron Mid-American <NA> #00285e #84754e   
## 3 Alabama SEC <NA> #9e1632 #ffffff   
## 4 App State Sun Belt East #ffcc00 #222222   
## 5 Arizona Big 12 <NA> #0c234b #ab0520   
## 6 Arizona St Big 12 <NA> #8e0c3a #ffc72c   
## 7 Arkansas SEC <NA> #a41f35 #ffffff   
## 8 Arkansas St Sun Belt West #e81018 #000000   
## 9 Army American Athletic <NA> #ce9c00 #231f20   
## 10 Auburn SEC <NA> #002b5c #f26522   
## `Logos[0]`   
## <chr>   
## 1 http://a.espncdn.com/i/teamlogos/ncaa/500/2005.png  
## 2 http://a.espncdn.com/i/teamlogos/ncaa/500/2006.png  
## 3 http://a.espncdn.com/i/teamlogos/ncaa/500/333.png   
## 4 http://a.espncdn.com/i/teamlogos/ncaa/500/2026.png  
## 5 http://a.espncdn.com/i/teamlogos/ncaa/500/12.png   
## 6 http://a.espncdn.com/i/teamlogos/ncaa/500/9.png   
## 7 http://a.espncdn.com/i/teamlogos/ncaa/500/8.png   
## 8 http://a.espncdn.com/i/teamlogos/ncaa/500/2032.png  
## 9 http://a.espncdn.com/i/teamlogos/ncaa/500/349.png   
## 10 http://a.espncdn.com/i/teamlogos/ncaa/500/2.png

## `Logos[1]` Twitter   
## <chr> <chr>   
## 1 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2005.png @AF\_Football   
## 2 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2006.png @ZipsFB   
## 3 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/333.png @AlabamaFTBL   
## 4 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2026.png @AppState\_FB   
## 5 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/12.png @ArizonaFBall   
## 6 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/9.png @ASUFootball   
## 7 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/8.png @RazorbackFB   
## 8 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2032.png @AStateFB   
## 9 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/349.png @ArmyWP\_Football  
## 10 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2.png @AuburnFootball   
## `Location Name` `Location City` `Location State` `Location Zip`  
## <chr> <chr> <chr> <dbl>  
## 1 Falcon Stadium Colorado Springs CO 80840  
## 2 InfoCision Stadium Akron OH 44399  
## 3 Bryant-Denny Stadium Tuscaloosa AL 35487  
## 4 Kidd Brewer Stadium Boone NC 28608  
## 5 Arizona Stadium Tucson AZ 85721  
## 6 Mountain America Stadium Tempe AZ 85287  
## 7 Razorback Stadium Fayetteville AR 72702  
## 8 Centennial Bank Stadium Jonesboro AR 72404  
## 9 Michie Stadium West Point NY 10996  
## 10 Jordan-Hare Stadium Auburn AL 36849  
## `Location Country Code` `Location Timezone` `Location Latitude`  
## <chr> <chr> <dbl>  
## 1 US America/Denver 39.0  
## 2 US America/New\_York 41.1  
## 3 US America/Chicago 33.2  
## 4 US America/New\_York 36.2  
## 5 US America/Phoenix 32.2  
## 6 US America/Phoenix 33.4  
## 7 US America/Chicago 36.1  
## 8 US America/Chicago 35.8  
## 9 US America/New\_York 41.4  
## 10 US America/Chicago 32.6  
## `Location Longitude` `Location Elevation` `Location Capacity`  
## <dbl> <dbl> <dbl>  
## 1 -105. 2025. 46692  
## 2 -81.5 321. 30000  
## 3 -87.6 70.1 101821  
## 4 -81.7 991. 30000  
## 5 -111. 742. 50782  
## 6 -112. 360. 65870  
## 7 -94.2 404. 80000  
## 8 -90.7 89.5 30382  
## 9 -74.0 99.2 38000  
## 10 -85.5 201. 87451  
## `Location Year Constructed` `Location Grass` `Location Dome`  
## <dbl> <lgl> <lgl>   
## 1 1962 FALSE FALSE   
## 2 2009 FALSE FALSE   
## 3 1929 TRUE FALSE   
## 4 1962 FALSE FALSE   
## 5 1928 FALSE FALSE   
## 6 1958 TRUE FALSE   
## 7 1938 FALSE FALSE   
## 8 1974 FALSE FALSE   
## 9 1924 FALSE FALSE   
## 10 1939 TRUE FALSE   
## # ℹ 124 more rows

Removing this unnecessary data is a good start, but there are still more columns which are not needed. As can be seen, there are three alternate name columns. Most of the cells in these columns are left empty, and most of the data that is in these columns is redundant anyway. Therefore, these columns can also be removed.

fbs <- select(fbs, -c("Alt Name1", "Alt Name2", "Alt Name3"))  
fbs %>% print(width = Inf)

## # A tibble: 134 × 23  
## School Mascot Abbreviation Conference Division Color   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West <NA> #004a7b  
## 2 Akron Zips AKR Mid-American <NA> #00285e  
## 3 Alabama Crimson Tide ALA SEC <NA> #9e1632  
## 4 App State Mountaineers APP Sun Belt East #ffcc00  
## 5 Arizona Wildcats ARIZ Big 12 <NA> #0c234b  
## 6 Arizona State Sun Devils ASU Big 12 <NA> #8e0c3a  
## 7 Arkansas Razorbacks ARK SEC <NA> #a41f35  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 9 Army Black Knights ARMY American Athletic <NA> #ce9c00  
## 10 Auburn Tigers AUB SEC <NA> #002b5c  
## `Alt Color` `Logos[0]`   
## <chr> <chr>   
## 1 #ffffff http://a.espncdn.com/i/teamlogos/ncaa/500/2005.png  
## 2 #84754e http://a.espncdn.com/i/teamlogos/ncaa/500/2006.png  
## 3 #ffffff http://a.espncdn.com/i/teamlogos/ncaa/500/333.png   
## 4 #222222 http://a.espncdn.com/i/teamlogos/ncaa/500/2026.png  
## 5 #ab0520 http://a.espncdn.com/i/teamlogos/ncaa/500/12.png   
## 6 #ffc72c http://a.espncdn.com/i/teamlogos/ncaa/500/9.png   
## 7 #ffffff http://a.espncdn.com/i/teamlogos/ncaa/500/8.png   
## 8 #000000 http://a.espncdn.com/i/teamlogos/ncaa/500/2032.png  
## 9 #231f20 http://a.espncdn.com/i/teamlogos/ncaa/500/349.png   
## 10 #f26522 http://a.espncdn.com/i/teamlogos/ncaa/500/2.png

## `Logos[1]` Twitter   
## <chr> <chr>   
## 1 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2005.png @AF\_Football   
## 2 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2006.png @ZipsFB   
## 3 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/333.png @AlabamaFTBL   
## 4 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2026.png @AppState\_FB   
## 5 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/12.png @ArizonaFBall   
## 6 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/9.png @ASUFootball   
## 7 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/8.png @RazorbackFB   
## 8 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2032.png @AStateFB   
## 9 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/349.png @ArmyWP\_Football  
## 10 http://a.espncdn.com/i/teamlogos/ncaa/500-dark/2.png @AuburnFootball   
## `Location Name` `Location City` `Location State` `Location Zip`  
## <chr> <chr> <chr> <dbl>  
## 1 Falcon Stadium Colorado Springs CO 80840  
## 2 InfoCision Stadium Akron OH 44399  
## 3 Bryant-Denny Stadium Tuscaloosa AL 35487  
## 4 Kidd Brewer Stadium Boone NC 28608  
## 5 Arizona Stadium Tucson AZ 85721  
## 6 Mountain America Stadium Tempe AZ 85287  
## 7 Razorback Stadium Fayetteville AR 72702  
## 8 Centennial Bank Stadium Jonesboro AR 72404  
## 9 Michie Stadium West Point NY 10996  
## 10 Jordan-Hare Stadium Auburn AL 36849  
## `Location Country Code` `Location Timezone` `Location Latitude`  
## <chr> <chr> <dbl>  
## 1 US America/Denver 39.0  
## 2 US America/New\_York 41.1  
## 3 US America/Chicago 33.2  
## 4 US America/New\_York 36.2  
## 5 US America/Phoenix 32.2  
## 6 US America/Phoenix 33.4  
## 7 US America/Chicago 36.1  
## 8 US America/Chicago 35.8  
## 9 US America/New\_York 41.4  
## 10 US America/Chicago 32.6  
## `Location Longitude` `Location Elevation` `Location Capacity`  
## <dbl> <dbl> <dbl>  
## 1 -105. 2025. 46692  
## 2 -81.5 321. 30000  
## 3 -87.6 70.1 101821  
## 4 -81.7 991. 30000  
## 5 -111. 742. 50782  
## 6 -112. 360. 65870  
## 7 -94.2 404. 80000  
## 8 -90.7 89.5 30382  
## 9 -74.0 99.2 38000  
## 10 -85.5 201. 87451  
## `Location Year Constructed` `Location Grass` `Location Dome`  
## <dbl> <lgl> <lgl>   
## 1 1962 FALSE FALSE   
## 2 2009 FALSE FALSE   
## 3 1929 TRUE FALSE   
## 4 1962 FALSE FALSE   
## 5 1928 FALSE FALSE   
## 6 1958 TRUE FALSE   
## 7 1938 FALSE FALSE   
## 8 1974 FALSE FALSE   
## 9 1924 FALSE FALSE   
## 10 1939 TRUE FALSE   
## # ℹ 124 more rows

The last couple steps have made the table much more concise, but there are still more columns which can be removed. As can be seen, the logo columns include links to logos for each school. While this information could be useful for some purposes, it takes up a ton of space and does not look good. Since this data is not really useful for our purposes, it is safe to just remove it.

fbs <- select(fbs, -c("Logos[0]", "Logos[1]"))  
fbs %>% print(width = Inf)

## # A tibble: 134 × 21  
## School Mascot Abbreviation Conference Division Color   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West <NA> #004a7b  
## 2 Akron Zips AKR Mid-American <NA> #00285e  
## 3 Alabama Crimson Tide ALA SEC <NA> #9e1632  
## 4 App State Mountaineers APP Sun Belt East #ffcc00  
## 5 Arizona Wildcats ARIZ Big 12 <NA> #0c234b  
## 6 Arizona State Sun Devils ASU Big 12 <NA> #8e0c3a  
## 7 Arkansas Razorbacks ARK SEC <NA> #a41f35  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 9 Army Black Knights ARMY American Athletic <NA> #ce9c00  
## 10 Auburn Tigers AUB SEC <NA> #002b5c  
## `Alt Color` Twitter `Location Name` `Location City`   
## <chr> <chr> <chr> <chr>   
## 1 #ffffff @AF\_Football Falcon Stadium Colorado Springs  
## 2 #84754e @ZipsFB InfoCision Stadium Akron   
## 3 #ffffff @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa   
## 4 #222222 @AppState\_FB Kidd Brewer Stadium Boone   
## 5 #ab0520 @ArizonaFBall Arizona Stadium Tucson   
## 6 #ffc72c @ASUFootball Mountain America Stadium Tempe   
## 7 #ffffff @RazorbackFB Razorback Stadium Fayetteville   
## 8 #000000 @AStateFB Centennial Bank Stadium Jonesboro   
## 9 #231f20 @ArmyWP\_Football Michie Stadium West Point   
## 10 #f26522 @AuburnFootball Jordan-Hare Stadium Auburn

## `Location State` `Location Zip` `Location Country Code` `Location Timezone`  
## <chr> <dbl> <chr> <chr>   
## 1 CO 80840 US America/Denver   
## 2 OH 44399 US America/New\_York   
## 3 AL 35487 US America/Chicago   
## 4 NC 28608 US America/New\_York   
## 5 AZ 85721 US America/Phoenix   
## 6 AZ 85287 US America/Phoenix   
## 7 AR 72702 US America/Chicago   
## 8 AR 72404 US America/Chicago   
## 9 NY 10996 US America/New\_York   
## 10 AL 36849 US America/Chicago   
## `Location Latitude` `Location Longitude` `Location Elevation`  
## <dbl> <dbl> <dbl>  
## 1 39.0 -105. 2025.   
## 2 41.1 -81.5 321.   
## 3 33.2 -87.6 70.1  
## 4 36.2 -81.7 991.   
## 5 32.2 -111. 742.   
## 6 33.4 -112. 360.   
## 7 36.1 -94.2 404.   
## 8 35.8 -90.7 89.5  
## 9 41.4 -74.0 99.2  
## 10 32.6 -85.5 201.   
## `Location Capacity` `Location Year Constructed` `Location Grass`  
## <dbl> <dbl> <lgl>   
## 1 46692 1962 FALSE   
## 2 30000 2009 FALSE   
## 3 101821 1929 TRUE   
## 4 30000 1962 FALSE   
## 5 50782 1928 FALSE   
## 6 65870 1958 TRUE   
## 7 80000 1938 FALSE   
## 8 30382 1974 FALSE   
## 9 38000 1924 FALSE   
## 10 87451 1939 TRUE

## `Location Dome`  
## <lgl>   
## 1 FALSE   
## 2 FALSE   
## 3 FALSE   
## 4 FALSE   
## 5 FALSE   
## 6 FALSE   
## 7 FALSE   
## 8 FALSE   
## 9 FALSE   
## 10 FALSE   
## # ℹ 124 more rows

Finally, there is one more column that can be removed. There is a country column that says what country every school is in. Since every FBS school is in the US, there is no need to have this column. Therefore, it can be safely removed.

fbs <- select(fbs, -"Location Country Code")  
fbs %>% print(width = Inf)

## # A tibble: 134 × 20  
## School Mascot Abbreviation Conference Division Color   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West <NA> #004a7b  
## 2 Akron Zips AKR Mid-American <NA> #00285e  
## 3 Alabama Crimson Tide ALA SEC <NA> #9e1632  
## 4 App State Mountaineers APP Sun Belt East #ffcc00  
## 5 Arizona Wildcats ARIZ Big 12 <NA> #0c234b  
## 6 Arizona State Sun Devils ASU Big 12 <NA> #8e0c3a  
## 7 Arkansas Razorbacks ARK SEC <NA> #a41f35  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 9 Army Black Knights ARMY American Athletic <NA> #ce9c00  
## 10 Auburn Tigers AUB SEC <NA> #002b5c

## `Alt Color` Twitter `Location Name` `Location City`   
## <chr> <chr> <chr> <chr>   
## 1 #ffffff @AF\_Football Falcon Stadium Colorado Springs  
## 2 #84754e @ZipsFB InfoCision Stadium Akron   
## 3 #ffffff @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa   
## 4 #222222 @AppState\_FB Kidd Brewer Stadium Boone   
## 5 #ab0520 @ArizonaFBall Arizona Stadium Tucson   
## 6 #ffc72c @ASUFootball Mountain America Stadium Tempe   
## 7 #ffffff @RazorbackFB Razorback Stadium Fayetteville   
## 8 #000000 @AStateFB Centennial Bank Stadium Jonesboro   
## 9 #231f20 @ArmyWP\_Football Michie Stadium West Point   
## 10 #f26522 @AuburnFootball Jordan-Hare Stadium Auburn   
## `Location State` `Location Zip` `Location Timezone` `Location Latitude`  
## <chr> <dbl> <chr> <dbl>  
## 1 CO 80840 America/Denver 39.0  
## 2 OH 44399 America/New\_York 41.1  
## 3 AL 35487 America/Chicago 33.2  
## 4 NC 28608 America/New\_York 36.2  
## 5 AZ 85721 America/Phoenix 32.2  
## 6 AZ 85287 America/Phoenix 33.4  
## 7 AR 72702 America/Chicago 36.1  
## 8 AR 72404 America/Chicago 35.8  
## 9 NY 10996 America/New\_York 41.4  
## 10 AL 36849 America/Chicago 32.6  
## `Location Longitude` `Location Elevation` `Location Capacity`  
## <dbl> <dbl> <dbl>  
## 1 -105. 2025. 46692  
## 2 -81.5 321. 30000  
## 3 -87.6 70.1 101821  
## 4 -81.7 991. 30000  
## 5 -111. 742. 50782  
## 6 -112. 360. 65870  
## 7 -94.2 404. 80000  
## 8 -90.7 89.5 30382  
## 9 -74.0 99.2 38000  
## 10 -85.5 201. 87451  
## `Location Year Constructed` `Location Grass` `Location Dome`  
## <dbl> <lgl> <lgl>   
## 1 1962 FALSE FALSE   
## 2 2009 FALSE FALSE   
## 3 1929 TRUE FALSE   
## 4 1962 FALSE FALSE   
## 5 1928 FALSE FALSE   
## 6 1958 TRUE FALSE   
## 7 1938 FALSE FALSE   
## 8 1974 FALSE FALSE   
## 9 1924 FALSE FALSE   
## 10 1939 TRUE FALSE   
## # ℹ 124 more rows

These are all the columns that can be completely removed. However, there is one more column that could be removed if we move its data elsewhere. As can be seen, the division column is empty for most schools. This is because the Sun Belt is the only conference that still has divisions. There is no need to have this column if it is mostly empty, but the data that is there is still important. Therefore, the best thing to do is to add the data from that column into the conference column.

fbs <- unite(fbs, "Conference", Conference, Division, sep = " ", na.rm = TRUE)  
filter(fbs, Conference == "Sun Belt East" | Conference == "Sun Belt West") %>% print(n = 14, width = Inf)

## # A tibble: 14 × 19  
## School Mascot Abbreviation Conference Color   
## <chr> <chr> <chr> <chr> <chr>   
## 1 App State Mountaineers APP Sun Belt East #ffcc00  
## 2 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 3 Coastal Carolina Chanticleers CCU Sun Belt East #007073  
## 4 Georgia Southern Eagles GASO Sun Belt East #003775  
## 5 Georgia State Panthers GAST Sun Belt East #1e539a  
## 6 James Madison Dukes JMU Sun Belt East #450084  
## 7 Louisiana Ragin' Cajuns UL Sun Belt West #ce181e  
## 8 Marshall Thundering Herd MRSH Sun Belt East #00ae42  
## 9 Old Dominion Monarchs ODU Sun Belt East #00507d  
## 10 South Alabama Jaguars USA Sun Belt West #00205B  
## 11 Southern Miss Golden Eagles USM Sun Belt West #FFAA3C  
## 12 Texas State Bobcats TXST Sun Belt West #4e1719  
## 13 Troy Trojans TROY Sun Belt West #AE0210  
## 14 UL Monroe Warhawks ULM Sun Belt West #231F20  
## `Alt Color` Twitter `Location Name` `Location City`  
## <chr> <chr> <chr> <chr>   
## 1 #222222 @AppState\_FB Kidd Brewer Stadium Boone   
## 2 #000000 @AStateFB Centennial Bank Stadium Jonesboro   
## 3 #876447 @CoastalFootball Brooks Stadium (SC) Conway   
## 4 #f0f0f0 @GSAthletics\_FB Allen E. Paulson Stadium Statesboro   
## 5 #ebebeb @GeorgiaStateFB Center Parc Stadium Atlanta   
## 6 #b5a068 @JMUFootball Bridgeforth Stadium Harrisonburg   
## 7 #ffffff @RaginCajunsFB Cajun Field Lafayette   
## 8 #be854c @HerdFB Joan C. Edwards Stadium Huntington   
## 9 #a1d2f1 @ODUFootball S.B. Ballard Stadium Norfolk   
## 10 #BF0D3E @SouthAlabamaFB Hancock Whitney Stadium Mobile   
## 11 #ffc423 @SouthernMissFB M. M. Roberts Stadium Hattiesburg   
## 12 #b4975a @TXStateFootball Bobcat Stadium (TX) San Marcos   
## 13 #88898c @TroyTrojansFB Veterans Memorial Stadium (AL) Troy   
## 14 #b18445 @ULM\_FB Malone Stadium Monroe

## `Location State` `Location Zip` `Location Timezone` `Location Latitude`  
## <chr> <dbl> <chr> <dbl>  
## 1 NC 28608 America/New\_York 36.2  
## 2 AR 72404 America/Chicago 35.8  
## 3 SC 29528 America/New\_York 33.8  
## 4 GA 30460 America/New\_York 32.4  
## 5 GA 30315 America/New\_York 33.7  
## 6 VA 22807 America/New\_York 38.4  
## 7 LA 70506 America/Chicago 30.2  
## 8 WV 25755 America/New\_York 38.4  
## 9 VA 23551 America/New\_York 36.9  
## 10 AL 85329 <NA> 30.7  
## 11 MS 39402 America/Chicago 31.3  
## 12 TX 78666 America/Chicago 29.9  
## 13 AL 36082 America/Chicago 31.8  
## 14 LA 71209 America/Chicago 32.5  
## `Location Longitude` `Location Elevation` `Location Capacity`  
## <dbl> <dbl> <dbl>  
## 1 -81.7 991. 30000  
## 2 -90.7 89.5 30382  
## 3 -79.0 13.7 21000  
## 4 -81.8 61.3 25000  
## 5 -84.4 287 25000  
## 6 -78.9 399. 24877  
## 7 -92.0 3.56 41426  
## 8 -82.4 164. 38016  
## 9 -76.3 3.15 20118  
## 10 -88.0 NA 25450  
## 11 -89.3 64.5 36000  
## 12 -97.9 178. 30000  
## 13 -86.0 155. 30000  
## 14 -92.1 23.0 30427  
## `Location Year Constructed` `Location Grass` `Location Dome`  
## <dbl> <lgl> <lgl>   
## 1 1962 FALSE FALSE   
## 2 1974 FALSE FALSE   
## 3 2003 FALSE FALSE   
## 4 1984 FALSE FALSE   
## 5 2017 FALSE FALSE   
## 6 1975 FALSE FALSE   
## 7 1971 FALSE FALSE   
## 8 1991 FALSE FALSE   
## 9 1936 FALSE FALSE   
## 10 NA FALSE FALSE   
## 11 1932 FALSE FALSE   
## 12 1981 FALSE FALSE   
## 13 1950 FALSE FALSE   
## 14 1978 FALSE FALSE

As seen in the above table, all Sun Belt teams now have their division included in their conference. All of the unnecessary columns have now been removed. However, some of the remaining column names could be improved. First of all, “Location Name” is not a good description for the column which lists each school’s stadium. Therefore, it would be best to change the name of that column to “Stadium”.

fbs <- rename(fbs, "Stadium" = "Location Name")  
fbs %>% print(width = Inf)

## # A tibble: 134 × 19  
## School Mascot Abbreviation Conference Color   
## <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West #004a7b  
## 2 Akron Zips AKR Mid-American #00285e  
## 3 Alabama Crimson Tide ALA SEC #9e1632  
## 4 App State Mountaineers APP Sun Belt East #ffcc00  
## 5 Arizona Wildcats ARIZ Big 12 #0c234b  
## 6 Arizona State Sun Devils ASU Big 12 #8e0c3a  
## 7 Arkansas Razorbacks ARK SEC #a41f35  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 9 Army Black Knights ARMY American Athletic #ce9c00  
## 10 Auburn Tigers AUB SEC #002b5c  
## `Alt Color` Twitter Stadium `Location City`   
## <chr> <chr> <chr> <chr>   
## 1 #ffffff @AF\_Football Falcon Stadium Colorado Springs  
## 2 #84754e @ZipsFB InfoCision Stadium Akron   
## 3 #ffffff @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa   
## 4 #222222 @AppState\_FB Kidd Brewer Stadium Boone   
## 5 #ab0520 @ArizonaFBall Arizona Stadium Tucson   
## 6 #ffc72c @ASUFootball Mountain America Stadium Tempe   
## 7 #ffffff @RazorbackFB Razorback Stadium Fayetteville   
## 8 #000000 @AStateFB Centennial Bank Stadium Jonesboro   
## 9 #231f20 @ArmyWP\_Football Michie Stadium West Point   
## 10 #f26522 @AuburnFootball Jordan-Hare Stadium Auburn   
## `Location State` `Location Zip` `Location Timezone` `Location Latitude`  
## <chr> <dbl> <chr> <dbl>  
## 1 CO 80840 America/Denver 39.0  
## 2 OH 44399 America/New\_York 41.1  
## 3 AL 35487 America/Chicago 33.2  
## 4 NC 28608 America/New\_York 36.2  
## 5 AZ 85721 America/Phoenix 32.2  
## 6 AZ 85287 America/Phoenix 33.4  
## 7 AR 72702 America/Chicago 36.1  
## 8 AR 72404 America/Chicago 35.8  
## 9 NY 10996 America/New\_York 41.4  
## 10 AL 36849 America/Chicago 32.6

## `Location Longitude` `Location Elevation` `Location Capacity`  
## <dbl> <dbl> <dbl>  
## 1 -105. 2025. 46692  
## 2 -81.5 321. 30000  
## 3 -87.6 70.1 101821  
## 4 -81.7 991. 30000  
## 5 -111. 742. 50782  
## 6 -112. 360. 65870  
## 7 -94.2 404. 80000  
## 8 -90.7 89.5 30382  
## 9 -74.0 99.2 38000  
## 10 -85.5 201. 87451  
## `Location Year Constructed` `Location Grass` `Location Dome`  
## <dbl> <lgl> <lgl>   
## 1 1962 FALSE FALSE   
## 2 2009 FALSE FALSE   
## 3 1929 TRUE FALSE   
## 4 1962 FALSE FALSE   
## 5 1928 FALSE FALSE   
## 6 1958 TRUE FALSE   
## 7 1938 FALSE FALSE   
## 8 1974 FALSE FALSE   
## 9 1924 FALSE FALSE   
## 10 1939 TRUE FALSE   
## # ℹ 124 more rows

Besides that column, every column after that also includes “Location” at the beginning of its name. Since there is no longer a “Location Name” column, that word does not really fit in the rest of the columns. Those columns do not need that word to describe what they are referring to anyway. Therefore, it would be best to remove that word from those columns.

fbs <- rename\_with(fbs, ~str\_remove(., "Location "))  
fbs %>% print(width = Inf)

## # A tibble: 134 × 19  
## School Mascot Abbreviation Conference Color   
## <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West #004a7b  
## 2 Akron Zips AKR Mid-American #00285e  
## 3 Alabama Crimson Tide ALA SEC #9e1632  
## 4 App State Mountaineers APP Sun Belt East #ffcc00  
## 5 Arizona Wildcats ARIZ Big 12 #0c234b  
## 6 Arizona State Sun Devils ASU Big 12 #8e0c3a  
## 7 Arkansas Razorbacks ARK SEC #a41f35  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018  
## 9 Army Black Knights ARMY American Athletic #ce9c00  
## 10 Auburn Tigers AUB SEC #002b5c

## `Alt Color` Twitter Stadium City State  
## <chr> <chr> <chr> <chr> <chr>  
## 1 #ffffff @AF\_Football Falcon Stadium Colorado Springs CO   
## 2 #84754e @ZipsFB InfoCision Stadium Akron OH   
## 3 #ffffff @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa AL   
## 4 #222222 @AppState\_FB Kidd Brewer Stadium Boone NC   
## 5 #ab0520 @ArizonaFBall Arizona Stadium Tucson AZ   
## 6 #ffc72c @ASUFootball Mountain America Stadium Tempe AZ   
## 7 #ffffff @RazorbackFB Razorback Stadium Fayetteville AR   
## 8 #000000 @AStateFB Centennial Bank Stadium Jonesboro AR   
## 9 #231f20 @ArmyWP\_Football Michie Stadium West Point NY   
## 10 #f26522 @AuburnFootball Jordan-Hare Stadium Auburn AL   
## Zip Timezone Latitude Longitude Elevation Capacity  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 80840 America/Denver 39.0 -105. 2025. 46692  
## 2 44399 America/New\_York 41.1 -81.5 321. 30000  
## 3 35487 America/Chicago 33.2 -87.6 70.1 101821  
## 4 28608 America/New\_York 36.2 -81.7 991. 30000  
## 5 85721 America/Phoenix 32.2 -111. 742. 50782  
## 6 85287 America/Phoenix 33.4 -112. 360. 65870  
## 7 72702 America/Chicago 36.1 -94.2 404. 80000  
## 8 72404 America/Chicago 35.8 -90.7 89.5 30382  
## 9 10996 America/New\_York 41.4 -74.0 99.2 38000  
## 10 36849 America/Chicago 32.6 -85.5 201. 87451  
## `Year Constructed` Grass Dome   
## <dbl> <lgl> <lgl>  
## 1 1962 FALSE FALSE  
## 2 2009 FALSE FALSE  
## 3 1929 TRUE FALSE  
## 4 1962 FALSE FALSE  
## 5 1928 FALSE FALSE  
## 6 1958 TRUE FALSE  
## 7 1938 FALSE FALSE  
## 8 1974 FALSE FALSE  
## 9 1924 FALSE FALSE  
## 10 1939 TRUE FALSE  
## # ℹ 124 more rows

Once this word is removed, there are still two column names left with multiple words and, by extension, quotation marks. The data would look much nicer without those quotation marks, so it would be best to turn those column names into one word. The first column name that needs to be changed is “Alt Color”. The best way to change it to one word while still distinguishing it from the “Color” column is to change “Color” to “Color1” and change “Alt Color” to “Color2”

fbs <- rename(fbs, "Color1" = "Color")  
fbs <- rename(fbs, "Color2" = "Alt Color")  
fbs %>% print(width = Inf)

## # A tibble: 134 × 19  
## School Mascot Abbreviation Conference Color1 Color2   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West #004a7b #ffffff  
## 2 Akron Zips AKR Mid-American #00285e #84754e  
## 3 Alabama Crimson Tide ALA SEC #9e1632 #ffffff  
## 4 App State Mountaineers APP Sun Belt East #ffcc00 #222222  
## 5 Arizona Wildcats ARIZ Big 12 #0c234b #ab0520  
## 6 Arizona State Sun Devils ASU Big 12 #8e0c3a #ffc72c  
## 7 Arkansas Razorbacks ARK SEC #a41f35 #ffffff  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018 #000000  
## 9 Army Black Knights ARMY American Athletic #ce9c00 #231f20  
## 10 Auburn Tigers AUB SEC #002b5c #f26522  
## Twitter Stadium City State Zip  
## <chr> <chr> <chr> <chr> <dbl>  
## 1 @AF\_Football Falcon Stadium Colorado Springs CO 80840  
## 2 @ZipsFB InfoCision Stadium Akron OH 44399  
## 3 @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa AL 35487  
## 4 @AppState\_FB Kidd Brewer Stadium Boone NC 28608  
## 5 @ArizonaFBall Arizona Stadium Tucson AZ 85721  
## 6 @ASUFootball Mountain America Stadium Tempe AZ 85287  
## 7 @RazorbackFB Razorback Stadium Fayetteville AR 72702  
## 8 @AStateFB Centennial Bank Stadium Jonesboro AR 72404  
## 9 @ArmyWP\_Football Michie Stadium West Point NY 10996  
## 10 @AuburnFootball Jordan-Hare Stadium Auburn AL 36849

## Timezone Latitude Longitude Elevation Capacity `Year Constructed`  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 America/Denver 39.0 -105. 2025. 46692 1962  
## 2 America/New\_York 41.1 -81.5 321. 30000 2009  
## 3 America/Chicago 33.2 -87.6 70.1 101821 1929  
## 4 America/New\_York 36.2 -81.7 991. 30000 1962  
## 5 America/Phoenix 32.2 -111. 742. 50782 1928  
## 6 America/Phoenix 33.4 -112. 360. 65870 1958  
## 7 America/Chicago 36.1 -94.2 404. 80000 1938  
## 8 America/Chicago 35.8 -90.7 89.5 30382 1974  
## 9 America/New\_York 41.4 -74.0 99.2 38000 1924  
## 10 America/Chicago 32.6 -85.5 201. 87451 1939  
## Grass Dome   
## <lgl> <lgl>  
## 1 FALSE FALSE  
## 2 FALSE FALSE  
## 3 TRUE FALSE  
## 4 FALSE FALSE  
## 5 FALSE FALSE  
## 6 TRUE FALSE  
## 7 FALSE FALSE  
## 8 FALSE FALSE  
## 9 FALSE FALSE  
## 10 TRUE FALSE  
## # ℹ 124 more rows

The other column name that needs to be changes is “Year Constructed”. While changing the name to one word, it would be best to indicate that this is the year each stadium opened, not when construction on it started. The best way to solve both of these problems would be to rename the “Year Constructed” column to “Opening”.

fbs <- rename(fbs, "Opening" = "Year Constructed")  
fbs %>% print(width = Inf)

## # A tibble: 134 × 19  
## School Mascot Abbreviation Conference Color1 Color2   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West #004a7b #ffffff  
## 2 Akron Zips AKR Mid-American #00285e #84754e  
## 3 Alabama Crimson Tide ALA SEC #9e1632 #ffffff  
## 4 App State Mountaineers APP Sun Belt East #ffcc00 #222222  
## 5 Arizona Wildcats ARIZ Big 12 #0c234b #ab0520  
## 6 Arizona State Sun Devils ASU Big 12 #8e0c3a #ffc72c  
## 7 Arkansas Razorbacks ARK SEC #a41f35 #ffffff  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018 #000000  
## 9 Army Black Knights ARMY American Athletic #ce9c00 #231f20  
## 10 Auburn Tigers AUB SEC #002b5c #f26522  
## Twitter Stadium City State Zip  
## <chr> <chr> <chr> <chr> <dbl>  
## 1 @AF\_Football Falcon Stadium Colorado Springs CO 80840  
## 2 @ZipsFB InfoCision Stadium Akron OH 44399  
## 3 @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa AL 35487  
## 4 @AppState\_FB Kidd Brewer Stadium Boone NC 28608  
## 5 @ArizonaFBall Arizona Stadium Tucson AZ 85721  
## 6 @ASUFootball Mountain America Stadium Tempe AZ 85287  
## 7 @RazorbackFB Razorback Stadium Fayetteville AR 72702  
## 8 @AStateFB Centennial Bank Stadium Jonesboro AR 72404  
## 9 @ArmyWP\_Football Michie Stadium West Point NY 10996  
## 10 @AuburnFootball Jordan-Hare Stadium Auburn AL 36849

## Timezone Latitude Longitude Elevation Capacity Opening Grass Dome   
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <lgl> <lgl>  
## 1 America/Denver 39.0 -105. 2025. 46692 1962 FALSE FALSE  
## 2 America/New\_York 41.1 -81.5 321. 30000 2009 FALSE FALSE  
## 3 America/Chicago 33.2 -87.6 70.1 101821 1929 TRUE FALSE  
## 4 America/New\_York 36.2 -81.7 991. 30000 1962 FALSE FALSE  
## 5 America/Phoenix 32.2 -111. 742. 50782 1928 FALSE FALSE  
## 6 America/Phoenix 33.4 -112. 360. 65870 1958 TRUE FALSE  
## 7 America/Chicago 36.1 -94.2 404. 80000 1938 FALSE FALSE  
## 8 America/Chicago 35.8 -90.7 89.5 30382 1974 FALSE FALSE  
## 9 America/New\_York 41.4 -74.0 99.2 38000 1924 FALSE FALSE  
## 10 America/Chicago 32.6 -85.5 201. 87451 1939 TRUE FALSE  
## # ℹ 124 more rows

After this, the columns are now cleaned up how we want them. Now that the columns have been dealt with, it is time to make sure the cell values are in order. First of all, we need to handle any null values. Therefore, we need to get a list of all values of NA in the data, so we can fix them accordingly.

filter(fbs, if\_any(everything(), is.na)) %>% print(width = Inf)

## # A tibble: 7 × 19  
## School Mascot Abbreviation Conference Color1 Color2   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Hawai'i Rainbow Warriors HAW Mountain West #003420 #ffffff  
## 2 Kennesaw State Owls KENN Conference USA #fdbb30 #000000  
## 3 Rutgers Scarlet Knights RUTG Big Ten #d21034 #ffffff  
## 4 South Alabama Jaguars USA Sun Belt West #00205B #BF0D3E  
## 5 UAB Blazers UAB American Athletic #003b28 #ffc845  
## 6 UNLV Rebels UNLV Mountain West #b10202 #ffffff  
## 7 UTSA Roadrunners UTSA American Athletic #002A5C #f47321  
## Twitter Stadium City State Zip  
## <chr> <chr> <chr> <chr> <dbl>  
## 1 @HawaiiFootball Clarence T.C. Ching Athletics Complex Honolulu HI 96822  
## 2 @KennesawStFB Fifth Third Stadium Kennesaw GA 48187  
## 3 @Rfootball SHI Stadium Piscataway NJ 8854  
## 4 @SouthAlabamaFB Hancock Whitney Stadium Mobile AL 85329  
## 5 @UAB\_FB Protective Stadium Birmingham AL 35203  
## 6 @UNLVFootball Allegiant Stadium Las Vegas NV 89154  
## 7 <NA> Alamodome San Antonio TX 78299

## Timezone Latitude Longitude Elevation Capacity Opening Grass Dome   
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <lgl> <lgl>  
## 1 <NA> 21.3 -158. 20 9000 NA TRUE FALSE  
## 2 <NA> 34.0 -84.6 297. 8318 2010 NA FALSE  
## 3 <NA> 40.5 -74.5 NA 52454 NA FALSE FALSE  
## 4 <NA> 30.7 -88.0 NA 25450 NA FALSE FALSE  
## 5 <NA> 33.5 -86.8 NA 0 NA FALSE FALSE  
## 6 <NA> 36.2 -115. NA 65000 NA FALSE TRUE   
## 7 America/Chicago 29.4 -98.5 202. 36582 1993 FALSE TRUE

As can be seen, there are seven schools who have cell values listed as NA. Affected columns include the timezone, elevation, construction, and grass columns. Since I want this data set to have all relevant information, I will use my own research to fill out each of these cells as accurately as possible.

fbs <- mutate(fbs, Twitter = ifelse(School == "UTSA", "@UTSAFTBL", Twitter))  
fbs <- mutate(fbs, Timezone = ifelse(School == "Hawai'i", "Pacific/Honolulu", Timezone))  
fbs <- mutate(fbs, Timezone = ifelse(School == "Kennesaw State" | School == "Rutgers", "America/New\_York", Timezone))  
fbs <- mutate(fbs, Timezone = ifelse(School == "South Alabama" | School == "UAB", "America/Chicago", Timezone))  
fbs <- mutate(fbs, Timezone = ifelse(School == "UNLV", "America/Los\_Angeles", Timezone))  
fbs <- mutate(fbs, Elevation = ifelse(School == "Rutgers", 50, Elevation))  
fbs <- mutate(fbs, Elevation = ifelse(School == "South Alabama", 33, Elevation))  
fbs <- mutate(fbs, Elevation = ifelse(School == "UAB", 597, Elevation))  
fbs <- mutate(fbs, Elevation = ifelse(School == "UNLV", 2190, Elevation))  
fbs <- mutate(fbs, Opening = ifelse(School == "Hawai'i", 2015, Opening))  
fbs <- mutate(fbs, Opening = ifelse(School == "Rutgers", 1994, Opening))  
fbs <- mutate(fbs, Opening = ifelse(School == "South Alabama", 2020, Opening))  
fbs <- mutate(fbs, Opening = ifelse(School == "UAB", 2021, Opening))  
fbs <- mutate(fbs, Opening = ifelse(School == "UNLV", 2020, Opening))  
fbs <- mutate(fbs, Grass = ifelse(School == "Kennesaw State", FALSE, Grass))  
filter(fbs, if\_any(everything(), is.na)) %>% print(width = Inf)

## # A tibble: 0 × 19  
## # ℹ 19 variables: School <chr>, Mascot <chr>, Abbreviation <chr>, Conference <chr>, Color1 <chr>, Color2 <chr>, Twitter <chr>, Stadium <chr>, City <chr>, State <chr>, Zip <dbl>, Timezone <chr>, Latitude <dbl>, Longitude <dbl>, Elevation <dbl>, Capacity <dbl>, Opening <dbl>, Grass <lgl>, Dome <lgl>

As seen from the empty tibble, all cells with values of NA have been updated to the most accurate data that could be found. However, there is still one more problem. In the earlier table, UAB’s Protective Stadium had a listed capacity of 0. This is clearly not accurate and must be fixed. First of all, we must check to see if any other schools have this problem.

filter(fbs, Capacity == 0) %>% print(width = Inf)

## # A tibble: 1 × 19  
## School Mascot Abbreviation Conference Color1 Color2 Twitter  
## <chr> <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 UAB Blazers UAB American Athletic #003b28 #ffc845 @UAB\_FB  
## Stadium City State Zip Timezone Latitude Longitude  
## <chr> <chr> <chr> <dbl> <chr> <dbl> <dbl>  
## 1 Protective Stadium Birmingham AL 35203 America/Chicago 33.5 -86.8  
## Elevation Capacity Opening Grass Dome   
## <dbl> <dbl> <dbl> <lgl> <lgl>  
## 1 597 0 2021 FALSE FALSE

Thankfully, no other schools have this problem, so this will be a pretty simple fix. Protective Stadium has a capacity of 47,100, so we will simply add that value to that cell.

fbs <- mutate(fbs, Capacity = ifelse(School == "UAB", 47100, Capacity))  
filter(fbs, Capacity == 0) %>% print(width = Inf)

## # A tibble: 0 × 19  
## # ℹ 19 variables: School <chr>, Mascot <chr>, Abbreviation <chr>, Conference <chr>, Color1 <chr>, Color2 <chr>, Twitter <chr>, Stadium <chr>, City <chr>, State <chr>, Zip <dbl>, Timezone <chr>, Latitude <dbl>, Longitude <dbl>, Elevation <dbl>, Capacity <dbl>, Opening <dbl>, Grass <lgl>, Dome <lgl>

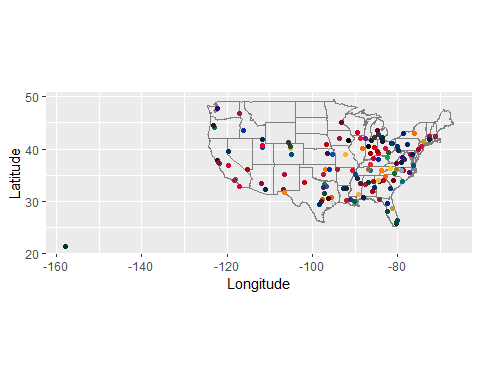
As seen from the empty table, this problem has been fixed. With that last fix, this data set is now tidy. Below we can get a look at this tidy data set before writing it into a new csv file. We can also use this data to map every FBS school. This is just the beginning of this data set’s usefulness, and it would be interesting to do more with it in the future.

fbs %>% print(width = Inf)

## # A tibble: 134 × 19  
## School Mascot Abbreviation Conference Color1 Color2   
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 Air Force Falcons AFA Mountain West #004a7b #ffffff  
## 2 Akron Zips AKR Mid-American #00285e #84754e  
## 3 Alabama Crimson Tide ALA SEC #9e1632 #ffffff  
## 4 App State Mountaineers APP Sun Belt East #ffcc00 #222222  
## 5 Arizona Wildcats ARIZ Big 12 #0c234b #ab0520  
## 6 Arizona State Sun Devils ASU Big 12 #8e0c3a #ffc72c  
## 7 Arkansas Razorbacks ARK SEC #a41f35 #ffffff  
## 8 Arkansas State Red Wolves ARST Sun Belt West #e81018 #000000  
## 9 Army Black Knights ARMY American Athletic #ce9c00 #231f20  
## 10 Auburn Tigers AUB SEC #002b5c #f26522  
## Twitter Stadium City State Zip  
## <chr> <chr> <chr> <chr> <dbl>  
## 1 @AF\_Football Falcon Stadium Colorado Springs CO 80840  
## 2 @ZipsFB InfoCision Stadium Akron OH 44399  
## 3 @AlabamaFTBL Bryant-Denny Stadium Tuscaloosa AL 35487  
## 4 @AppState\_FB Kidd Brewer Stadium Boone NC 28608  
## 5 @ArizonaFBall Arizona Stadium Tucson AZ 85721  
## 6 @ASUFootball Mountain America Stadium Tempe AZ 85287  
## 7 @RazorbackFB Razorback Stadium Fayetteville AR 72702  
## 8 @AStateFB Centennial Bank Stadium Jonesboro AR 72404  
## 9 @ArmyWP\_Football Michie Stadium West Point NY 10996  
## 10 @AuburnFootball Jordan-Hare Stadium Auburn AL 36849

## Timezone Latitude Longitude Elevation Capacity Opening Grass Dome   
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <lgl> <lgl>  
## 1 America/Denver 39.0 -105. 2025. 46692 1962 FALSE FALSE  
## 2 America/New\_York 41.1 -81.5 321. 30000 2009 FALSE FALSE  
## 3 America/Chicago 33.2 -87.6 70.1 101821 1929 TRUE FALSE  
## 4 America/New\_York 36.2 -81.7 991. 30000 1962 FALSE FALSE  
## 5 America/Phoenix 32.2 -111. 742. 50782 1928 FALSE FALSE  
## 6 America/Phoenix 33.4 -112. 360. 65870 1958 TRUE FALSE  
## 7 America/Chicago 36.1 -94.2 404. 80000 1938 FALSE FALSE  
## 8 America/Chicago 35.8 -90.7 89.5 30382 1974 FALSE FALSE  
## 9 America/New\_York 41.4 -74.0 99.2 38000 1924 FALSE FALSE  
## 10 America/Chicago 32.6 -85.5 201. 87451 1939 TRUE FALSE  
## # ℹ 124 more rows

fbs %>% ggplot(aes(Longitude, Latitude, color = Color1)) + borders("state") + geom\_point() + coord\_quickmap() + scale\_color\_identity()



write\_csv(fbs, "fbs2.csv")

source: https://collegefootballdata.com/exporter/teams/fbs?year=2024