R code for Data Science for Beginners

Day 4: Individual Exercise

Aric Jensen

2024-09-12

Clean up your workspace

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.2
v ggplot2   4.0.0     v tibble    3.3.0
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.1.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
rm(list=ls(all=TRUE)) # remove all the named objects visible in the environment
cat("\014") # clean your console
```

1. Let's do more exercises with dplyr (with a different dataset)

Please download the nycflights13 data by installing this package called `nycflights13`

```
# install.packages("nycflights13")
library("nycflights13")
head(flights)
```

```
# A tibble: 6 x 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
# 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>
```

1-1: Please find all March flights in the data (the dataset is named "flights") flights

```
march_flights <- flights %>%
  filter(month==3)
march_flights
```

```
# A tibble: 28,834 x 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     3     1        4           2159       125      318             56
 2  2013     3     1       50           2358        52      526            438
 3  2013     3     1      117           2245       152      223           2354
 4  2013     3     1      454            500        -6      633            648
 5  2013     3     1      505            515       -10      746            810
 6  2013     3     1      521            530        -9      813            827
 7  2013     3     1      537            540        -3      856            850
 8  2013     3     1      541            545        -4     1014           1023
 9  2013     3     1      549            600       -11      639            703
10  2013     3     1      550            600       -10      747            801
```

```
# i 28,824 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>
```

1-2 :Create a new variable as date with a format like this 1/1/2013, using the `mutate()` function

```
flights %>% select(1:3) %>% mutate(date = paste(month, day, year, sep="/"))
```

```
# A tibble: 336,776 x 4
    year month   day date
   <int> <int> <int> <chr>
 1  2013     1     1 1/1/2013
 2  2013     1     1 1/1/2013
 3  2013     1     1 1/1/2013
 4  2013     1     1 1/1/2013
 5  2013     1     1 1/1/2013
 6  2013     1     1 1/1/2013
 7  2013     1     1 1/1/2013
 8  2013     1     1 1/1/2013
 9  2013     1     1 1/1/2013
10  2013     1     1 1/1/2013
# i 336,766 more rows
```

1-3: Change column name tailnum to tail_number

```
flights %>% rename (tail_number = tailnum) %>% select(tail_number)
```

```
# A tibble: 336,776 x 1
   tail_number
   <chr>
 1 N14228
 2 N24211
 3 N619AA
 4 N804JB
 5 N668DN
 6 N39463
```

```
 7 N516JB
 8 N829AS
 9 N593JB
10 N3ALAA
# i 336,766 more rows
```

1-4: Group flights by their origins

```
flights %>% group_by(origin)
```

```
# A tibble: 336,776 x 19
# Groups:   origin [3]
    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>
```

1-5: Count how many flights departing from JFK on 2013-12-31?

```
flights %>% filter(origin == "JFK" & year == 2013 & month == 12 & day == 31) %>% nrow()
```

```
[1] 283
```

1-6: Calculate the average hours of delay in departure for all flights from JFK

```
flights %>% group_by(origin) %>%
  summarise(delay_avg = mean(dep_delay, na.rm = T))
```

```
# A tibble: 3 x 2
  origin delay_avg
  <chr>      <dbl>
1 EWR         15.1
2 JFK         12.1
3 LGA         10.3
```

Finally, execute the entire contents of this file. Make sure that you don't get any error message. If you get an error message, it's probably because you forgot to comment out something.