Homework 5

MTH 3270 Data Science

Due Mon., Mar. 14

Read These Chapters of the Book	**Then Do These Exercises**
5	Problems 1-3* (**below**), Problems 3* (skip part *c*), 4* (**Ch 5**)
6	Problems 2, 3**, 5***, 7**** (**Ch 6**)

* **Problems 3** (**below**), **3** (**Ch 5**), and **4** (**Ch 5**) all use the `"nycflights13"` package, but in addition to the `flights` data, they also use the `planes` data set. Type `?planes` for more info.

** For **Problem 3** (**Ch 6**) you can copy and paste the following into R:

```
library(readr)      # For parse_number().

x1 <- c("1900.45", "$1900.45", "1,900.45", "nearly $2000")
x2 <- as.factor(x1)

parse_number(x1)
parse_number(x2)

as.numeric(x1)
as.numeric(x2)
```

*** For **Problem 5** (**Ch 6**), you can create the data frame using:

```
my.data <- data.frame(grp = rep(c("A", "B"), each = 2),
                      sex = rep(c("F", "M"), times = 2),
                      meanL = c(0.22, 0.47, 0.33, 0.55),
                      sdL = c(0.11, 0.33, 0.11, 0.31),
                      meanR = c(0.34, 0.57, 0.40, 0.65),
                      sdR = c(0.09, 0.33, 0.07, 0.27))
```

**** For **Problem 7** (**Ch 6**), you can create the data frame `ds1` using:

```
ds1 <- data.frame(id = rep(1:3, times = 2),
                  group = rep(c("T", "C"), each = 3),
                  vals = c(4, 6, 8, 5, 6, 10))
```

1 Consider the following data in the file **houses-for-sale.txt** (from **pg 126** of the textbook ***Modern Data Science with R***):

```
myURL <- "http://sites.msudenver.edu/ngrevsta/wp-content/
  uploads/sites/416/2021/02/houses-for-sale.txt"

Houses <- read.csv(myURL, header = TRUE, sep = "\t")
```

We'll use a *subset* of the variables, namely `fuel`, `heat`, `sewer`, and `construction`:

```
Houses_small <- select(Houses, fuel, heat, sewer, construction)
```

To *recode* `fuel` as `"gas"`, `"electric"`, etc., `sewer` as `"none"`, `"private"`, etc., and so on, we first create a *codebook* data frame that can be used to translate the **integers** to `"character"`:

```
myURL <- "http://sites.msudenver.edu/ngrevsta/wp-content/uploads/
  sites/416/2021/02/house_codes.txt"

Translations <- read.csv(myURL,
                         header = TRUE,
                         stringsAsFactors = FALSE,
                         sep = "\t")
```

```
head(Translations)

##   code system_type meaning
## 1    0   new_const      no
## 2    1   new_const     yes
## 3    1  sewer_type    none
## 4    2  sewer_type private
## 5    3  sewer_type  public
## 6    0 central_air      no
```

The same information can also be presented in a wide format:

```
codes <- Translations %>% pivot_wider(names_from = system_type,
                                      values_from = meaning,
                                      values_fill = "invalid")
```

```
codes
```

```
## # A tibble: 5 x 6
##    code new_const sewer_type central_air fuel_type
##   <int> <chr>     <chr>      <chr>       <chr>
## 1     0 no        invalid    no          invalid
## 2     1 yes       none       yes         invalid
## 3     2 invalid   private    invalid     gas
## 4     3 invalid   public     invalid     electric
## 5     4 invalid   invalid    invalid     oil
## # ... with 1 more variable: heat_type <chr>
```

As an example, below we use `left_join()` to merge `Houses_small` with `codes`, matching rows in `codes` by `code` to rows in `Houses_small` by `fuel`:

```
Houses_small <- left_join(x = Houses_small,
                          y = select(codes, code, fuel_type),
                          by = c(fuel = "code"))
```

Here's the resulting data set, with the *recoded* `fuel` variable:

```
head(Houses_small)
```

```
##   fuel heat sewer construction fuel_type
## 1    3    4     2            0  electric
## 2    2    3     2            0       gas
## 3    2    3     3            0       gas
## 4    2    2     2            0       gas
## 5    2    2     3            1       gas
## 6    2    2     2            0       gas
```

a) Report R commands that *recode* the remaining variables (`heat`, `sewer`, `construction`) in `Houses_small`, then *remove* the original (**integer**-valued) variables. You should end up with this:

```
head(Houses_small)
```

```
##   fuel_type heat_type sewer_type new_const
## 1  electric  electric    private        no
## 2       gas hot water    private        no
## 3       gas hot water     public        no
## 4       gas   hot air    private        no
## 5       gas   hot air     public       yes
## 6       gas   hot air    private        no
```

b) Now (using `Houses_small` obtained in Part *a*), describe in words what the following command does. Then rewrite it into a more readable version using the **pipe operator** `%>%`.

```
arrange(summarize(group_by(select(filter(Houses_small, new_const == "no"),
  fuel_type, heat_type), fuel_type), count = n()), desc(count))
```

Hint: Recall that when two function calls are *nested*, R evaluates the *inner* one first, then passes its returned value to the *outer* one.

2 Using the `flights` data set (from the `"nycflights13"` package), for each destination (`dest`), determine the *total* minutes of delay and the *average* minutes of delay. Report your R command(s).

3 The `flights` data set contains information about each *flight* in 2013. The `planes` data set contains information about each *airplane*.

a) Which variable would be the ***key*** for combining the two data frames using one of the `*_join()` functions?

b) Combine the `flights` and `planes` data sets using an appropriate `*_join()` function. Which `manufacturer` made the most flights in 2013? How many flights did it make?