assignment_05_MunjewarSheetal.R

sheetal

2023-01-15

```
# Assignment: ASSIGNMENT 5.0
# Name: Munjewar, Sheetal
# Date: 2023-01-15

# Sample data set to play around.
# https://www.jaredlander.com/datasets/

# Efficient way to install and load the packages.
# Reference - https://statsandr.com/blog/an-efficient-way-to-install-and-load-r-packages/


# Check your current working directory using `getwd()`
getwd()
```

```
## [1] "E:/Data_Science_DSC510/DSC520-Statistics/dsc520/assignments/assignment05"
```

```
# List the contents of the working directory with the `dir()` function
dir()
```

```
## [1] "assignment_05_LastnameFirstname.R"
## [2] "assignment_05_MunjewarSheetal.pdf"
## [3] "assignment_05_MunjewarSheetal.R"
## [4] "assignment_05_MunjewarSheetal.spin.R"
## [5] "assignment_05_MunjewarSheetal.spin.Rmd"
## [6] "purrr-cheat-sheet.pdf"
## [7] "Scratch.R"
```

```
# If the current directory does not contain the `data` directory, set the
# working directory to project root folder (the folder should contain the `data` directory
# Use `setwd()` if needed
# setwd("E:\\Data_Science_DSC510\\DSC520-Statistics\\dsc520")
setwd("E:\\Data_Science_DSC510\\DSC520-Statistics\\dsc520")
housing_df <- read.csv("data/week-7-housing.csv")
head(housing_df)
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     649990           1               3                    R1
## 3  1/3/2006     572500           1               3                    R1
## 4  1/3/2006     420000           1               3                    R1
## 5  1/3/2006     369900           1               3           15       R1
## 6  1/3/2006     184667           1              15         18 51       R1
##            addr_full  zip5 ctyname postalctyn       lon      lat building_grade
## 1  17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2  11927 178TH PL NE 98052 REDMOND    REDMOND -122.1022 47.70731              9
## 3 13315 174TH AVE NE 98052            REDMOND -122.1085 47.71986              8
## 4  3303 178TH AVE NE 98052 REDMOND    REDMOND -122.1037 47.63914              8
## 5  16126 NE 108TH CT 98052 REDMOND    REDMOND -122.1242 47.69748              7
## 6   8101 229TH DR NE 98053            REDMOND -122.0341 47.67545              7
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2880        4               2               0
## 3                     2770        4               1               1
## 4                     1620        3               1               0
## 5                     1440        3               1               0
## 6                     4160        4               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               1       2006              0             R4      5570         R
## 3               1       1987              0             R6      8444         R
## 4               1       1968              0             R4      9600         R
## 5               1       1980              0             R6      7526         R
## 6               1       2005              0          URPSO      7280         R
##   present_use
## 1           2
## 2           2
## 3           2
## 4           2
## 5           2
## 6           2
```

```r
# Package names
# packages <- c("ggplot2","dplyr","tidyr","magrittr","tidyverse","purrr")
packages <- c("ggplot2","dplyr","magrittr","tidyverse","purrr")

# Install packages not yet installed
installed_packages <- packages %in% rownames(installed.packages())
if (any(installed_packages == FALSE)) {
  install.packages(packages[!installed_packages])
}

# Packages loading
invisible(lapply(packages, library, character.only = TRUE))
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.8     v purrr   1.0.0
## v tidyr   1.2.1     v stringr 1.5.0
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract()   masks magrittr::extract()
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x purrr::set_names() masks magrittr::set_names()
```

```
# Column 1 to 3 with sales price > 500000
hdf_2006 <- housing_df[housing_df["Sale.Price"] >= 500008,1:3]
head(hdf_2006)
```

```
##   Sale.Date Sale.Price sale_reason
## 1  1/3/2006     698000           1
## 2  1/3/2006     649990           1
## 3  1/3/2006     572500           1
## 7  1/4/2006    1050000           1
## 8  1/4/2006     875000           1
## 9  1/4/2006     660000           1
```

```
# Assignment-01 :
# Using the dplyr package, use the 6 different operations to analyze/transform
# the data - GroupBy, Summarize, Mutate, Filter, Select, and Arrange - Remember
# this isn't just modifying data, you are learning about your data also -
# so play around and start to understand your dataset in more detail.
```

```
# Select examples
str(housing_df)
```

```
## 'data.frame':    12865 obs. of  24 variables:
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ Sale.Price              : int  698000 649990 572500 420000 369900 184667 1050000 875000 660000 65
##  $ sale_reason             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ sale_instrument         : int  3 3 3 3 3 15 3 3 3 3 ...
##  $ sale_warning            : chr  "" "" "" "" ...
##  $ sitetype                : chr  "R1" "R1" "R1" "R1" ...
##  $ addr_full               : chr  "17021 NE 113TH CT" "11927 178TH PL NE" "13315 174TH AVE NE" "3303
##  $ zip5                    : int  98052 98052 98052 98052 98052 98053 98053 98053 98053 98052 ...
##  $ ctyname                 : chr  "REDMOND" "REDMOND" "" "REDMOND" ...
##  $ postalctyn              : chr  "REDMOND" "REDMOND" "REDMOND" "REDMOND" ...
##  $ lon                     : num  -122 -122 -122 -122 -122 ...
##  $ lat                     : num  47.7 47.7 47.7 47.6 47.7 ...
##  $ building_grade          : int  9 9 8 8 7 7 10 10 9 8 ...
##  $ square_feet_total_living: int  2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ bedrooms                : int  4 4 4 3 3 4 5 4 4 4 ...
##  $ bath_full_count         : int  2 2 1 1 1 2 3 2 2 1 ...
##  $ bath_half_count         : int  1 0 1 0 0 1 0 1 1 0 ...
##  $ bath_3qtr_count         : int  0 1 1 1 1 1 1 0 1 1 ...
##  $ year_built              : int  2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 ...
##  $ year_renovated          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr  "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : int  6635 5570 8444 9600 7526 7280 97574 30649 42688 94889 ...
##  $ prop_type               : chr  "R" "R" "R" "R" ...
##  $ present_use             : int  2 2 2 2 2 2 2 2 2 2 ...
```

```
housing_df %>% select(1,3) %>% tail()
```

```
##        Sale.Date sale_reason
## 12860 12/15/2016           1
## 12861 12/15/2016           1
## 12862 12/15/2016           1
## 12863 12/15/2016           1
## 12864 12/16/2016           1
## 12865 12/16/2016           1
```

```
housing_df %>% select(Sale.Date,Sale.Price,sale_warning) %>% tail()
```

```
##        Sale.Date Sale.Price sale_warning
## 12860 12/15/2016     824000
## 12861 12/15/2016     798930
## 12862 12/15/2016     750000
## 12863 12/15/2016     629000
## 12864 12/16/2016     835000
## 12865 12/16/2016     455500
```

```
housing_df %>% select(c(Sale.Date,Sale.Price)) %>% head()
```

```
##   Sale.Date Sale.Price
## 1  1/3/2006     698000
## 2  1/3/2006     649990
## 3  1/3/2006     572500
## 4  1/3/2006     420000
## 5  1/3/2006     369900
## 6  1/3/2006     184667
```

```
# Columns starts and end with 'S' or 's' -Not case sensitive.
housing_df %>% select(starts_with('s')) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     649990           1               3                    R1
## 3  1/3/2006     572500           1               3                    R1
## 4  1/3/2006     420000           1               3                    R1
## 5  1/3/2006     369900           1               3           15       R1
## 6  1/3/2006     184667           1              15        18 51       R1
##   square_feet_total_living sq_ft_lot
## 1                     2810      6635
## 2                     2880      5570
## 3                     2770      8444
## 4                     1620      9600
## 5                     1440      7526
## 6                     4160      7280
```

```
housing_df %>% select(ends_with('s')) %>% head()
```

```
##   bedrooms
## 1        4
## 2        4
## 3        4
## 4        3
## 5        3
## 6        4
```

```
housing_df %>% select(contains('sale')) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning
## 1  1/3/2006     698000           1               3
## 2  1/3/2006     649990           1               3
## 3  1/3/2006     572500           1               3
## 4  1/3/2006     420000           1               3
## 5  1/3/2006     369900           1               3           15
## 6  1/3/2006     184667           1              15        18 51
```

```
housing_df %>% select(matches('s.+l')) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning postalctyn
## 1  1/3/2006     698000           1               3                 REDMOND
## 2  1/3/2006     649990           1               3                 REDMOND
## 3  1/3/2006     572500           1               3                 REDMOND
## 4  1/3/2006     420000           1               3                 REDMOND
## 5  1/3/2006     369900           1               3           15    REDMOND
## 6  1/3/2006     184667           1              15        18 51    REDMOND
##   square_feet_total_living sq_ft_lot
## 1                     2810      6635
## 2                     2880      5570
## 3                     2770      8444
## 4                     1620      9600
## 5                     1440      7526
## 6                     4160      7280
```

```
housing_df %>% select(-1,-2) %>% head()
```

```
##   sale_reason sale_instrument sale_warning sitetype          addr_full  zip5
## 1           1               3                    R1  17021 NE 113TH CT 98052
## 2           1               3                    R1  11927 178TH PL NE 98052
## 3           1               3                    R1 13315 174TH AVE NE 98052
## 4           1               3                    R1  3303 178TH AVE NE 98052
## 5           1               3           15       R1  16126 NE 108TH CT 98052
## 6           1              15        18 51       R1   8101 229TH DR NE 98053
##   ctyname postalctyn       lon      lat building_grade square_feet_total_living
## 1 REDMOND    REDMOND -122.1124 47.70139              9                     2810
## 2 REDMOND    REDMOND -122.1022 47.70731              9                     2880
## 3            REDMOND -122.1085 47.71986              8                     2770
## 4 REDMOND    REDMOND -122.1037 47.63914              8                     1620
## 5 REDMOND    REDMOND -122.1242 47.69748              7                     1440
## 6            REDMOND -122.0341 47.67545              7                     4160
##   bedrooms bath_full_count bath_half_count bath_3qtr_count year_built
## 1        4               2               1               0       2003
## 2        4               2               0               1       2006
## 3        4               1               1               1       1987
## 4        3               1               0               1       1968
## 5        3               1               0               1       1980
## 6        4               2               1               1       2005
##   year_renovated current_zoning sq_ft_lot prop_type present_use
## 1              0             R4      6635         R           2
## 2              0             R4      5570         R           2
## 3              0             R6      8444         R           2
## 4              0             R4      9600         R           2
## 5              0             R6      7526         R           2
## 6              0          URPSO      7280         R           2
```

```
# filter examples

housing_df %>% filter(bath_full_count==2) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     649990           1               3                    R1
## 3  1/3/2006     184667           1              15        18 51       R1
## 4  1/4/2006     875000           1               3                    R1
## 5  1/4/2006     660000           1               3                    R1
## 6  1/4/2006     599950           1               3                    R1
##           addr_full  zip5 ctyname postalctyn       lon       lat building_grade
## 1 17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2 11927 178TH PL NE 98052 REDMOND    REDMOND -122.1022 47.70731              9
## 3  8101 229TH DR NE 98053            REDMOND -122.0341 47.67545              7
## 4  21404 NE 67TH ST 98053            REDMOND -122.0555 47.66510             10
## 5 7525 238TH AVE NE 98053            REDMOND -122.0227 47.67208              9
## 6  14924 NE 74TH CT 98052 REDMOND    REDMOND -122.1411 47.67142              9
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2880        4               2               0
## 3                     4160        4               2               1
## 4                     3720        4               2               1
## 5                     4160        4               2               1
## 6                     2180        3               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               1       2006              0             R4      5570         R
## 3               1       2005              0          URPSO      7280         R
## 4               0       1988              0            RA5     30649         R
## 5               1       1978              0            RA5     42688         R
## 6               0       1988              0             R5      7949         R
##   present_use
## 1           2
## 2           2
## 3           2
## 4           2
## 5           2
## 6           2
```

```r
housing_df %>% filter(year_built == 2003) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2 1/10/2006     482000           1               3                    R1
## 3 1/31/2006     148000          14              15           18       R1
## 4  2/1/2006     393000           1               3                    R1
## 5 2/17/2006     390000           1               3                    R1
## 6 2/23/2006     543000           1               3           40       R1
##           addr_full  zip5 ctyname postalctyn       lon      lat building_grade
## 1 17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2  9166 226TH PL NE 98053            REDMOND -122.0376 47.68249              7
## 3  6517 188TH PL NE 98052 REDMOND    REDMOND -122.0871 47.66449              9
## 4  9036 229TH PL NE 98053            REDMOND -122.0339 47.68188              7
## 5 9024 228TH WAY NE 98053            REDMOND -122.0347 47.68172              7
## 6  18893 NE 68TH ST 98052 REDMOND    REDMOND -122.0862 47.66605              9
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2360        3               2               1
## 3                     1930        3               2               1
## 4                     1560        3               2               1
## 5                     1530        3               1               1
## 6                     2080        3               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               0       2003              0          URPSO      4080         R
## 3               0       2003              0            R12      3430         R
## 4               0       2003              0          URPSO      4995         R
## 5               1       2003              0          URPSO      3056         R
## 6               0       2003              0            R12      3600         R
##   present_use
## 1           2
## 2           2
## 3           2
## 4         300
## 5           2
## 6           2
```

```r
housing_df %>% filter(year_built == 2003 & bath_full_count == 2 ) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2 1/10/2006     482000           1               3                    R1
## 3 1/31/2006     148000          14              15           18       R1
## 4  2/1/2006     393000           1               3                    R1
## 5 2/23/2006     543000           1               3           40       R1
## 6 2/24/2006     543000           1               3           41       R1
##           addr_full  zip5 ctyname postalctyn       lon      lat building_grade
## 1 17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2  9166 226TH PL NE 98053            REDMOND -122.0376 47.68249              7
## 3  6517 188TH PL NE 98052 REDMOND    REDMOND -122.0871 47.66449              9
## 4  9036 229TH PL NE 98053            REDMOND -122.0339 47.68188              7
## 5  18893 NE 68TH ST 98052 REDMOND    REDMOND -122.0862 47.66605              9
## 6  18893 NE 68TH ST 98052 REDMOND    REDMOND -122.0862 47.66605              9
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2360        3               2               1
## 3                     1930        3               2               1
## 4                     1560        3               2               1
## 5                     2080        3               2               1
## 6                     2080        3               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               0       2003              0          URPSO      4080         R
## 3               0       2003              0            R12      3430         R
## 4               0       2003              0          URPSO      4995         R
## 5               0       2003              0            R12      3600         R
## 6               0       2003              0            R12      3600         R
##   present_use
## 1           2
## 2           2
## 3           2
## 4         300
## 5           2
## 6           2
```

```
housing_df %>% filter(zip5 == 98052 & bath_full_count == 2 & Sale.Price > 500000 ) %>% head()
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     649990           1               3                    R1
## 3  1/4/2006     599950           1               3                    R1
## 4  1/4/2006     526787           1               3                    R1
## 5  1/5/2006     507950           1               3                    R1
## 6  1/6/2006     589950           1               3                    R1
##           addr_full  zip5 ctyname postalctyn       lon      lat building_grade
## 1 17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2 11927 178TH PL NE 98052 REDMOND    REDMOND -122.1022 47.70731              9
## 3  14924 NE 74TH CT 98052 REDMOND    REDMOND -122.1411 47.67142              9
## 4  7858 148TH CT NE 98052 REDMOND    REDMOND -122.1425 47.67407              8
## 5  7850 148TH CT NE 98052 REDMOND    REDMOND -122.1425 47.67390              8
## 6 11922 173RD PL NE 98052 REDMOND    REDMOND -122.1086 47.70678              8
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2880        4               2               0
## 3                     2180        3               2               1
## 4                     2480        3               2               1
## 5                     2480        3               2               1
## 6                     2570        4               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               1       2006              0             R4      5570         R
## 3               0       1988              0             R5      7949         R
## 4               0       2005              0             R5      2647         R
## 5               0       2005             0             R5      3099         R
## 6               0       2005             0             R4      4737         R
##   present_use
## 1           2
## 2           2
## 3           2
## 4           2
## 5           2
## 6           2
```

```
housing_df %>% filter((zip5 == 98052 | zip5 == 98053) & bath_full_count == 2 & Sale.Price > 500000 ) %>%
```

```
##    Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1   1/3/2006     698000           1               3                    R1
## 2   1/3/2006     649990           1               3                    R1
## 3   1/4/2006     875000           1               3                    R1
## 4   1/4/2006     660000           1               3                    R1
## 5   1/4/2006     599950           1               3                    R1
## 6   1/4/2006     526787           1               3                    R1
## 7   1/5/2006     803000           1               3                    R1
## 8   1/5/2006     507950           1               3                    R1
## 9   1/6/2006     765000           1               3                    R1
## 10  1/6/2006     589950           1               3                    R1
## 11 1/10/2006     513262           1               3                    R1
## 12 1/12/2006     717390           1               3                    R1
## 13 1/12/2006     552000           1               3                    R1
## 14 1/13/2006     523935           1               3                    R1
## 15 1/16/2006     572950           1               3                    R1
## 16 1/17/2006     905000           1               3           41       R1
## 17 1/17/2006     750073           1               3                    R1
## 18 1/17/2006     526718           1               3                    R1
## 19 1/18/2006     690749           1               3                    R1
## 20 1/18/2006     640000           1               3                    R1
## 21 1/23/2006     729000           1               3                    R1
## 22 1/24/2006     754500           1               3                    R1
## 23 1/24/2006     640900           1               3                    R1
## 24 1/24/2006     589950           1               3                    R1
## 25 1/26/2006    1053649           1               3                    R1
## 26 1/26/2006     870000           1               3           40       R1
## 27 1/26/2006     844148           1               3                    R1
## 28 1/26/2006     689000           1               3                    R1
## 29 1/27/2006     849990           1               3                    R1
## 30 1/27/2006     798000           1               3                    R1
## 31 1/27/2006     527593           1               3                    R1
## 32 1/30/2006     599500           1               3                    R1
## 33 1/30/2006     540430           1               3                    R1
## 34  2/1/2006    1080135           1               3                    R1
## 35  2/2/2006     579950           1               3                    R1
## 36  2/2/2006     538595           1               3                    R1
## 37  2/2/2006     535500           1               3                    R1
## 38  2/2/2006     530020           1               3                    R2
## 39  2/3/2006    1075000           1               3                    R1
## 40  2/3/2006     691321           1               3                    R1
## 41  2/3/2006     657953           1               3                    R1
## 42  2/3/2006     623990           1               3                    R1
## 43  2/6/2006     661500           1               3                    R1
## 44  2/6/2006     650100           1               3                    R1
## 45  2/7/2006     879950           1               3                    R1
## 46  2/7/2006     810000           1               3                    R1
## 47  2/7/2006     655000           1               3                    R1
## 48  2/7/2006     550000           1               3                    R1
## 49  2/8/2006     665906           1               3                    R1
## 50  2/9/2006     711000           1               3                    R1
##                    addr_full  zip5 ctyname postalctyn       lon      lat
## 1         17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139
## 2         11927 178TH PL NE 98052 REDMOND    REDMOND -122.1022 47.70731
## 3          21404 NE 67TH ST 98053            REDMOND -122.0555 47.66510
## 4         7525 238TH AVE NE 98053            REDMOND -122.0227 47.67208
## 5          14924 NE 74TH CT 98052 REDMOND    REDMOND -122.1411 47.67142
## 6          7858 148TH CT NE 98052 REDMOND    REDMOND -122.1425 47.67407
## 7         3624 264TH AVE NE 98053            REDMOND -121.9857 47.64184
## 8          7850 148TH CT NE 98052 REDMOND    REDMOND -122.1425 47.67390
## 9          8944 237TH PL NE 98053            REDMOND -122.0230 47.68150
## 10        11922 173RD PL NE 98052 REDMOND    REDMOND -122.1086 47.70678
## 11        11807 242ND PL NE 98053            REDMOND -122.0162 47.70323
## 12       16565 NE 119TH WAY 98052 REDMOND    REDMOND -122.1197 47.70583
## 13         6128 145TH CT NE 98052 REDMOND    REDMOND -122.1463 47.66187
## 14        11823 242ND PL NE 98053            REDMOND -122.0160 47.70348
## 15       17264 NE 119TH WAY 98052 REDMOND    REDMOND -122.1095 47.70728
## 16         24215 NE 59TH LN 98053            REDMOND -122.0145 47.65899
## 17        12240 243RD PL NE 98053            REDMOND -122.0148 47.70771
## 18        23844 NE ADAIR RD 98053            REDMOND -122.0202 47.70903
## 19        12223 243RD PL NE 98053            REDMOND -122.0155 47.70756
## 20         22233 NE 46TH ST 98053            REDMOND -122.0424 47.64993
## 21         3918 225TH CT NE 98053            REDMOND -122.0380 47.64553
## 22        12123 177TH CT NE 98052 REDMOND    REDMOND -122.1033 47.70873
## 23        12232 243RD PL NE 98053            REDMOND -122.0148 47.70755
## 24        11842 173RD PL NE 98052 REDMOND    REDMOND -122.1083 47.70629
## 25        23821 NE ADAIR RD 98053            REDMOND -122.0212 47.70902
## 26        10138 219TH PL NE 98053            REDMOND -122.0471 47.68932
## 27         16135 NE 41ST CT 98052 REDMOND    REDMOND -122.1246 47.64694
## 28        15847 NE 117TH ST 98052 REDMOND    REDMOND -122.1273 47.70426
## 29       12016 176TH AVE NE 98052 REDMOND    REDMOND -122.1046 47.70781
## 30        11444 176TH PL NE 98052 REDMOND    REDMOND -122.1047 47.70214
## 31        11815 242ND PL NE 98053            REDMOND -122.0161 47.70336
## 32         9903 187TH CT NE 98052 REDMOND    REDMOND -122.0909 47.68718
## 33        14884 NE 78TH WAY 98052 REDMOND    REDMOND -122.1421 47.67383
## 34        23837 NE ADAIR RD 98053            REDMOND -122.0207 47.70876
## 35        17229 NE 119TH CT 98052 REDMOND    REDMOND -122.1090 47.70637
## 36    12244 BIG LEAF WAY NE 98053            REDMOND -122.0133 47.70807
## 37 21035 NE NOVELTY HILL RD 98053            REDMOND -122.0582 47.68969
## 38    12300 BIG LEAF WAY NE 98053            REDMOND -122.0137 47.70826
## 39        8814 218TH AVE NE 98053            REDMOND -122.0476 47.68053
## 40        12215 243RD PL NE 98053            REDMOND -122.0156 47.70742
## 41        12201 243RD PL NE 98053            REDMOND -122.0154 47.70710
## 42        17735 NE 122ND ST 98052 REDMOND    REDMOND -122.1028 47.70898
## 43       13232 171ST AVE NE 98052            REDMOND -122.1116 47.71892
## 44         9912 187TH CT NE 98052 REDMOND    REDMOND -122.0903 47.68729
## 45       11705 157TH AVE NE 98052 REDMOND    REDMOND -122.1311 47.70518
## 46        3939 259TH WAY NE 98053            REDMOND -121.9927 47.64363
## 47       13808 178TH AVE NE 98052            REDMOND -122.1019 47.72446
## 48        19155 NE 66TH WAY 98052 REDMOND    REDMOND -122.0832 47.66473
## 49        12248 243RD PL NE 98053            REDMOND -122.0147 47.70787
## 50        17002 NE 133RD ST 98052            REDMOND -122.1145 47.71850
##    building_grade square_feet_total_living bedrooms bath_full_count
## 1               9                     2810        4               2
## 2               9                     2880        4               2
## 3              10                     3720        4               2
## 4               9                     4160        4               2
## 5               9                     2180        3               2
## 6               8                     2480        3               2
## 7              10                     3180        3               2
## 8               8                     2480        3               2
## 9               9                     4000        4               2
## 10              8                     2570        4               2
## 11              8                     1930        2               2
## 12              9                     3090        3               2
## 13              9                     2050        3               2
## 14              8                     1680        2               2
## 15              8                     2530        4               2
## 16             10                     3520        4               2
## 17              8                     2300        3               2
## 18              8                     1680        2               2
## 19              8                     2305        3               2
## 20              9                     3010        4               2
## 21             10                     3400        3               2
## 22              9                     3150        4               2
## 23              8                     2520        3               2
## 24              8                     2530        4               2
## 25              9                     2680        2               2
## 26             10                     4100        4               2
## 27             10                     3160        3               2
## 28              9                     2680        4               2
## 29              9                     3990        5               2
## 30              9                     3020        4               2
## 31              8                     1870        2               2
## 32              9                     2740        4               2
## 33              8                     2480        3               2
## 34              9                     2700        3               2
## 35              8                     2370        3               2
## 36              8                     1680        2               2
## 37              8                     3120        4               2
## 38              8                     1900        2               2
## 39             11                     4340        4               2
## 40              8                     2520        3               2
## 41              8                     2170        2               2
## 42              9                     2880        4               2
## 43             10                     2330        4               2
## 44              9                     2870        4               2
## 45              9                     2930        4               2
## 46             10                     3260        3               2
## 47              9                     2700        4               2
## 48              9                     2550        3               2
## 49              8                     2390        2               2
## 50             10                     2540        4               2
##    bath_half_count bath_3qtr_count year_built year_renovated current_zoning
## 1                1               0       2003              0             R4
## 2                0               1       2006              0             R4
## 3                1               0       1988              0            RA5
## 4                1               1       1978              0            RA5
## 5                1               0       1988              0             R5
## 6                1               0       2005              0             R5
## 7                1               0       1990              0            RA5
## 8                1               0       2005              0             R5
## 9                1               1       2005              0          URPSO
## 10               1               0       2005              0             R4
## 11               0               0       2005              0          URPSO
## 12               1               0       2006              0             R4
## 13               0               0       1986              0             R5
## 14               0               0       2005              0          URPSO
## 15               1               0       2006              0             R4
## 16               1               1       1998              0          RA2.5
## 17               1               0       2006              0          URPSO
## 18               0               0       2005              0          URPSO
## 19               1               0       2006              0          URPSO
## 20               0               1       1986              0            RA5
## 21               1               0       1999              0           RA10
## 22               1               0       2005              0             R4
## 23               1               0       2005              0          URPSO
## 24               1               0       2005              0             R4
## 25               1               0       2005              0          URPSO
## 26               1               0       1990              0            RA5
## 27               1               0       1989              0             R4
## 28               1               0       2004              0             R4
## 29               0               1       2005              0             R4
## 30               0               1       1977              0             R4
## 31               0               0       2005              0          URPSO
## 32               1               0       2005              0             R4
## 33               1               0       2005              0             R5
## 34               0               1       2006              0          URPSO
## 35               1               0       2006              0             R4
## 36               0               0       2006              0          URPSO
## 37               0               1       1968              0           RA5P
## 38               1               0       2006              0          URPSO
## 39               1               0       1992              0            RA5
## 40               1               0       2006              0          URPSO
## 41               1               0       2006              0          URPSO
## 42               0               1       2005              0             R4
## 43               1               0       1985              0          RA2.5
## 44               1               0       2005              0             R4
## 45               1               0       2005              0             R4
## 46               1               0       1990              0            RA5
## 47               1               0       1985              0             R6
## 48               1               0       2004              0            R12
## 49               1               0       2006              0          URPSO
## 50               1               0       1984              0          RA2.5
##    sq_ft_lot prop_type present_use
## 1       6635         R           2
## 2       5570         R           2
## 3      30649         R           2
## 4      42688         R           2
## 5       7949         R           2
## 6       2647         R           2
## 7      95013         R           2
## 8       3099         R           2
## 9       7611         R           2
## 10      4737         R           2
## 11      4958         R           2
## 12      5760         R           2
## 13     10827         R           2
## 14      4764         R           2
## 15      3792         R           2
## 16     46270         R           2
## 17      5697         R           2
## 18      6386         R           2
## 19      7583         R           2
## 20     35006         R           2
## 21     21807         R           2
## 22      6312         R           2
## 23      5580         R           2
## 24      3832         R           2
## 25      8517         R           2
## 26     47042         R           2
## 27      9521         R           2
## 28      4970         R           2
## 29     13519         R           2
## 30     52915         R           2
## 31      4639         R           2
## 32      5072         R           2
## 33      2747         R           2
## 34      7694         R           2
## 35      3442         R           2
## 36      6069         R           2
## 37     31520         R           2
## 38      3306         R          29
## 39    141570         R           2
## 40      5934         R           2
## 41      5833         R           2
## 42      5229         R           2
## 43     29888         R           2
## 44      6962         R           2
## 45     10605         R           2
## 46     71869         R           2
## 47      9837         R           2
## 48      5376         R           2
## 49      5704         R           2
## 50     28563         R           2
```

```
county = "REDMOND"
#housing_df %>% filter_(~ctyname == county)
```

```
# housing_df%>% head(,n=50)

# Get distinct values from zip5 column.
distinct(housing_df,zip5)
```

```
##    zip5
## 1 98052
## 2 98053
## 3 98074
## 4 98059
```

```
distinct(housing_df,bedrooms)
```

```
##    bedrooms
## 1         4
## 2         3
## 3         5
## 4         2
## 5         6
## 6         0
## 7         9
## 8         7
## 9         1
## 10        8
## 11       10
## 12       11
```

```
distinct(housing_df,ctyname)
```

```
##     ctyname
## 1   REDMOND
## 2
## 3 SAMMAMISH
```

```
distinct(housing_df,current_zoning)
```

```
##    current_zoning
## 1              R4
## 2              R6
## 3           URPSO
## 4             RA5
## 5              R3
## 6              R5
## 7           RA2.5
## 8            RA10
## 9             R12
## 10           RA5P
## 11             R1
## 12        RA2.5SO
## 13         RA2.5P
## 14           R4/C
## 15             EH
## 16            R1P
## 17             BC
## 18             R8
## 19            A10
## 20           R6/C
## 21            R18
## 22          A10SO
## 23          RA10P
## 24             GC
```

```
distinct(housing_df,housing_df$sale_reason)  #-- How to check for group_by with count.
```

```
##    housing_df$sale_reason
## 1                       1
## 2                      12
## 3                       8
## 4                      14
## 5                      18
## 6                      17
## 7                      10
## 8                       6
## 9                       2
## 10                      4
## 11                     13
## 12                     11
## 13                     16
## 14                     19
## 15                      7
## 16                      0
## 17                      3
```

```
distinct(housing_df,ctyname,.keep_all = TRUE)
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     572500           1               3                    R1
## 3 4/14/2006    1369900           1               3                    R1
##            addr_full  zip5   ctyname postalctyn       lon      lat
## 1  17021 NE 113TH CT 98052   REDMOND    REDMOND -122.1124 47.70139
## 2 13315 174TH AVE NE 98052              REDMOND -122.1085 47.71986
## 3   24620 NE 27TH PL 98074 SAMMAMISH    REDMOND -122.0104 47.63458
##   building_grade square_feet_total_living bedrooms bath_full_count
## 1              9                     2810        4               2
## 2              8                     2770        4               1
## 3             11                     4630        5               2
##   bath_half_count bath_3qtr_count year_built year_renovated current_zoning
## 1               1               0       2003              0             R4
## 2               1               1       1987              0             R6
## 3               0               2       2005              0             R1
##   sq_ft_lot prop_type present_use
## 1      6635         R           2
## 2      8444         R           2
## 3     18297         R           2
```

```
# List top 50 blank values in column ctyname
housing_df %>% filter(ctyname=="") %>% head(.,n=50)
```

```
##    Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1   1/3/2006     572500           1               3                    R1
## 2   1/3/2006     184667           1              15        18 51       R1
## 3   1/4/2006    1050000           1               3                    R1
## 4   1/4/2006     875000           1               3                    R1
## 5   1/4/2006     660000           1               3                    R1
## 6   1/4/2006     165000           1               3                    R1
## 7   1/5/2006     803000           1               3                    R1
## 8   1/6/2006     765000           1               3                    R1
## 9   1/9/2006     372500           1               3                    R1
## 10 1/10/2006     513262           1               3                    R1
## 11 1/10/2006     482000           1               3                    R1
## 12 1/11/2006     765000           1               3                    R1
## 13 1/11/2006     265000           1               3                    R1
## 14 1/13/2006     523935           1               3                    R1
## 15 1/13/2006     399900           1               3                    R1
## 16 1/17/2006     905000           1               3           41       R1
## 17 1/17/2006     750073           1               3                    R1
## 18 1/17/2006     526718           1               3                    R1
## 19 1/18/2006     690749           1               3                    R1
## 20 1/18/2006     640000           1               3                    R1
## 21 1/18/2006     485000           1               3                    R1
## 22 1/20/2006     462150           1               3                    R1
## 23 1/20/2006     418000           1               3           15       R1
## 24 1/20/2006     350000          12               3                    R1
## 25 1/23/2006    1445000           1               3                    R1
## 26 1/23/2006     729000           1               3                    R1
## 27 1/24/2006     640900           1               3                    R1
## 28 1/24/2006     443509           1               3                    R1
## 29 1/26/2006    1053649           1               3                    R1
## 30 1/26/2006     870000           1               3           40       R1
## 31 1/26/2006     446400           8               3           12       R1
## 32 1/27/2006     527593           1               3                    R1
## 33  2/1/2006    1900000           1               3        15 52       R1
## 34  2/1/2006    1080135           1               3                    R1
## 35  2/1/2006     451129           1               3                    R2
## 36  2/1/2006     393000           1               3                    R1
## 37  2/2/2006     538595           1               3                    R1
## 38  2/2/2006     535500           1               3                    R1
## 39  2/2/2006     530020           1               3                    R2
## 40  2/3/2006    1075000           1               3                    R1
## 41  2/3/2006     691321           1               3                    R1
## 42  2/3/2006     657953           1               3                    R1
## 43  2/6/2006     661500           1               3                    R1
## 44  2/7/2006     810000           1               3                    R1
## 45  2/7/2006     732500           1               3                    R1
## 46  2/7/2006     655000           1               3                    R1
## 47  2/8/2006     665906           1               3                    R1
## 48  2/9/2006     711000           1               3                    R1
## 49  2/9/2006     521674           1               3                    R1
## 50  2/9/2006     488610           1               3                    R1
##                   addr_full  zip5 ctyname postalctyn       lon      lat
## 1        13315 174TH AVE NE 98052            REDMOND -122.1085 47.71986
## 2          8101 229TH DR NE 98053            REDMOND -122.0341 47.67545
## 3          21634 NE 87TH PL 98053            REDMOND -122.0507 47.68053
## 4          21404 NE 67TH ST 98053            REDMOND -122.0555 47.66510
## 5         7525 238TH AVE NE 98053            REDMOND -122.0227 47.67208
## 6         2921 288TH AVE NE 98053            REDMOND -121.9577 47.63382
## 7         3624 264TH AVE NE 98053            REDMOND -121.9857 47.64184
## 8          8944 237TH PL NE 98053            REDMOND -122.0230 47.68150
## 9          26920 NE 50TH ST 98053            REDMOND -121.9795 47.65158
## 10        11807 242ND PL NE 98053            REDMOND -122.0162 47.70323
## 11         9166 226TH PL NE 98053            REDMOND -122.0376 47.68249
## 12        4811 228TH AVE NE 98053            REDMOND -122.0365 47.65149
## 13   25149 NE PATTERSON WAY 98053            REDMOND -122.0032 47.65814
## 14        11823 242ND PL NE 98053            REDMOND -122.0160 47.70348
## 15  24307 NE VINE MAPLE WAY 98053            REDMOND -122.0151 47.70263
## 16         24215 NE 59TH LN 98053            REDMOND -122.0145 47.65899
## 17        12240 243RD PL NE 98053            REDMOND -122.0148 47.70771
## 18        23844 NE ADAIR RD 98053            REDMOND -122.0202 47.70903
## 19        12223 243RD PL NE 98053            REDMOND -122.0155 47.70756
## 20         22233 NE 46TH ST 98053            REDMOND -122.0424 47.64993
## 21        17609 NE 141ST ST 98052            REDMOND -122.1040 47.72588
## 22        22862 NE 128TH PL 98053            REDMOND -122.0335 47.71358
## 23   3419 W AMES LAKE DR NE 98053            REDMOND -121.9637 47.63966
## 24        6028 215TH AVE NE 98053            REDMOND -122.0538 47.66165
## 25         20425 NE 71ST ST 98053            REDMOND -122.0665 47.66792
## 26         3918 225TH CT NE 98053            REDMOND -122.0380 47.64553
## 27        12232 243RD PL NE 98053            REDMOND -122.0148 47.70755
## 28  24253 NE VINE MAPLE WAY 98053            REDMOND -122.0153 47.70267
## 29        23821 NE ADAIR RD 98053            REDMOND -122.0212 47.70902
## 30        10138 219TH PL NE 98053            REDMOND -122.0471 47.68932
## 31         28616 NE 47TH PL 98053            REDMOND -121.9569 47.65066
## 32        11815 242ND PL NE 98053            REDMOND -122.0161 47.70336
## 33        6507 240TH WAY NE 98053            REDMOND -122.0175 47.66478
## 34        23837 NE ADAIR RD 98053            REDMOND -122.0207 47.70876
## 35    12252 BIG LEAF WAY NE 98053            REDMOND -122.0136 47.70820
## 36         9036 229TH PL NE 98053            REDMOND -122.0339 47.68188
## 37    12244 BIG LEAF WAY NE 98053            REDMOND -122.0133 47.70807
## 38 21035 NE NOVELTY HILL RD 98053            REDMOND -122.0582 47.68969
## 39    12300 BIG LEAF WAY NE 98053            REDMOND -122.0137 47.70826
## 40        8814 218TH AVE NE 98053            REDMOND -122.0476 47.68053
## 41        12215 243RD PL NE 98053            REDMOND -122.0156 47.70742
## 42        12201 243RD PL NE 98053            REDMOND -122.0154 47.70710
## 43       13232 171ST AVE NE 98052            REDMOND -122.1116 47.71892
## 44        3939 259TH WAY NE 98053            REDMOND -121.9927 47.64363
## 45         20709 NE 79TH ST 98053            REDMOND -122.0631 47.67378
## 46       13808 178TH AVE NE 98052            REDMOND -122.1019 47.72446
## 47        12248 243RD PL NE 98053            REDMOND -122.0147 47.70787
## 48        17002 NE 133RD ST 98052            REDMOND -122.1145 47.71850
## 49    12308 BIG LEAF WAY NE 98053            REDMOND -122.0138 47.70833
## 50    12324 BIG LEAF WAY NE 98053            REDMOND -122.0141 47.70843
##    building_grade square_feet_total_living bedrooms bath_full_count
## 1               8                     2770        4               1
## 2               7                     4160        4               2
## 3              10                     3960        5               3
## 4              10                     3720        4               2
## 5               9                     4160        4               2
## 6               9                     1850        3               2
## 7              10                     3180        3               2
## 8               9                     4000        4               2
## 9               7                     1620        3               1
## 10              8                     1930        2               2
## 11              7                     2360        3               2
## 12              9                     3520        4               3
## 13             10                     4920        4               4
## 14              8                     1680        2               2
## 15              8                     1350        2               2
## 16             10                     3520        4               2
## 17              8                     2300        3               2
## 18              8                     1680        2               2
## 19              8                     2305        3               2
## 20              9                     3010        4               2
## 21              8                     1920        3               2
## 22              8                     1570        2               2
## 23              8                     1620        3               2
## 24              6                     1420        4               1
## 25             11                     5440        4               3
## 26             10                     3400        3               2
## 27              8                     2520        3               2
## 28              8                     1350        2               1
## 29              9                     2680        2               2
## 30             10                     4100        4               2
## 31              7                     1770        3               3
## 32              8                     1870        2               2
## 33             11                     6610        4               3
## 34              9                     2700        3               2
## 35              8                     1510        2               2
## 36              7                     1560        3               2
## 37              8                     1680        2               2
## 38              8                     3120        4               2
## 39              8                     1900        2               2
## 40             11                     4340        4               2
## 41              8                     2520        3               2
## 42              8                     2170        2               2
## 43             10                     2330        4               2
## 44             10                     3260        3               2
## 45              9                     5710        5               3
## 46              9                     2700        4               2
## 47              8                     2390        2               2
## 48             10                     2540        4               2
## 49              8                     1900        2               2
## 50              8                     1510        2               1
##    bath_half_count bath_3qtr_count year_built year_renovated current_zoning
## 1                1               1       1987              0             R6
## 2                1               1       2005              0          URPSO
## 3                0               1       1993              0            RA5
## 4                1               0       1988              0            RA5
## 5                1               1       1978              0            RA5
## 6                0               0       2011              0            RA5
## 7                1               0       1990              0            RA5
## 8                1               1       2005              0          URPSO
## 9                0               1       1977              0            RA5
## 10               0               0       2005              0          URPSO
## 11               1               0       2003              0          URPSO
## 12               0               0       1987              0            RA5
## 13               1               0       2007              0            RA5
## 14               0               0       2005              0          URPSO
## 15               0               0       2005              0          URPSO
## 16               1               1       1998              0          RA2.5
## 17               1               0       2006              0          URPSO
## 18               0               0       2005              0          URPSO
## 19               1               0       2006              0          URPSO
## 20               0               1       1986              0            RA5
## 21               0               0       1986              0             R6
## 22               0               0       2005              0          URPSO
## 23               0               0       1979              0            RA5
## 24               1               0       1970              0            RA5
## 25               1               0       1991              0            RA5
## 26               1               0       1999              0           RA10
## 27               1               0       2005              0          URPSO
## 28               0               1       2005              0          URPSO
## 29               1               0       2005              0          URPSO
## 30               1               0       1990              0            RA5
## 31               0               0       1984              0            RA5
## 32               0               0       2005              0          URPSO
## 33               1               1       1990              0            RA5
## 34               0               1       2006              0          URPSO
## 35               0               0       2006              0          URPSO
## 36               1               0       2003              0          URPSO
## 37               0               0       2006              0          URPSO
## 38               0               1       1968              0           RA5P
## 39               1               0       2006              0          URPSO
## 40               1               0       1992              0            RA5
## 41               1               0       2006              0          URPSO
## 42               1               0       2006              0          URPSO
## 43               1               0       1985              0          RA2.5
## 44               1               0       1990              0            RA5
## 45               2               1       1977           2004            RA5
## 46               1               0       1985              0             R6
## 47               1               0       2006              0          URPSO
## 48               1               0       1984              0          RA2.5
## 49               1               0       2006              0          URPSO
## 50               0               1       2006              0          URPSO
##    sq_ft_lot prop_type present_use
## 1       8444         R           2
## 2       7280         R           2
## 3      97574         R           2
## 4      30649         R           2
## 5      42688         R           2
## 6     278891         R           2
## 7      95013         R           2
## 8       7611         R           2
## 9      47480         R           2
## 10      4958         R           2
## 11      4080         R           2
## 12     35348         R           2
## 13    112650         R           2
## 14      4764         R           2
## 15      4781         R          29
## 16     46270         R           2
## 17      5697         R           2
## 18      6386         R           2
## 19      7583         R           2
## 20     35006         R           2
## 21      7560         R           2
## 22      4766         R           2
## 23     29285         R           2
## 24     28087         R           2
## 25     36446         R           2
## 26     21807         R           2
## 27      5580         R           2
## 28      4939         R          29
## 29      8517         R           2
## 30     47042         R           2
## 31    220654         R           2
## 32      4639         R           2
## 33     37017         R           2
## 34      7694         R           2
## 35      4581         R          29
## 36      4995         R         300
## 37      6069         R           2
## 38     31520         R           2
## 39      3306         R          29
## 40    141570         R           2
## 41      5934         R           2
## 42      5833         R           2
## 43     29888         R           2
## 44     71869         R           2
## 45     10200         R           2
## 46      9837         R           2
## 47      5704         R           2
## 48     28563         R           2
## 49      3587         R          29
## 50      3966         R          29
```

```
# find NULL/NA values in a entire DF
is.na(housing_df) %>% head()
```

```
##      Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## [1,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
## [2,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
## [3,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
## [4,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
## [5,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
## [6,]     FALSE      FALSE       FALSE           FALSE        FALSE    FALSE
##      addr_full  zip5 ctyname postalctyn   lon   lat building_grade
## [1,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
## [2,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
## [3,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
## [4,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
## [5,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
## [6,]     FALSE FALSE   FALSE      FALSE FALSE FALSE          FALSE
##      square_feet_total_living bedrooms bath_full_count bath_half_count
## [1,]                    FALSE    FALSE           FALSE           FALSE
## [2,]                    FALSE    FALSE           FALSE           FALSE
## [3,]                    FALSE    FALSE           FALSE           FALSE
## [4,]                    FALSE    FALSE           FALSE           FALSE
## [5,]                    FALSE    FALSE           FALSE           FALSE
## [6,]                    FALSE    FALSE           FALSE           FALSE
##      bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot
## [1,]           FALSE      FALSE          FALSE          FALSE     FALSE
## [2,]           FALSE      FALSE          FALSE          FALSE     FALSE
## [3,]           FALSE      FALSE          FALSE          FALSE     FALSE
## [4,]           FALSE      FALSE          FALSE          FALSE     FALSE
## [5,]           FALSE      FALSE          FALSE          FALSE     FALSE
## [6,]           FALSE      FALSE          FALSE          FALSE     FALSE
##      prop_type present_use
## [1,]     FALSE       FALSE
## [2,]     FALSE       FALSE
## [3,]     FALSE       FALSE
## [4,]     FALSE       FALSE
## [5,]     FALSE       FALSE
## [6,]     FALSE       FALSE
```

```
summary(housing_df)
```

```
##   Sale.Date           Sale.Price       sale_reason    sale_instrument
##  Length:12865       Min.   :    698   Min.   : 0.00   Min.   : 0.000
##  Class :character   1st Qu.: 460000   1st Qu.: 1.00   1st Qu.: 3.000
##  Mode  :character   Median : 593000   Median : 1.00   Median : 3.000
##                     Mean   : 660738   Mean   : 1.55   Mean   : 3.678
##                     3rd Qu.: 750000   3rd Qu.: 1.00   3rd Qu.: 3.000
##                     Max.   :4400000   Max.   :19.00   Max.   :27.000
##  sale_warning         sitetype          addr_full              zip5
##  Length:12865       Length:12865       Length:12865       Min.   :98052
##  Class :character   Class :character   Class :character   1st Qu.:98052
##  Mode  :character   Mode  :character   Mode  :character   Median :98052
##                                                           Mean   :98053
##                                                           3rd Qu.:98053
##                                                           Max.   :98074
##    ctyname           postalctyn             lon              lat
##  Length:12865       Length:12865       Min.   :-122.2   Min.   :47.46
##  Class :character   Class :character   1st Qu.:-122.1   1st Qu.:47.67
##  Mode  :character   Mode  :character   Median :-122.1   Median :47.69
##                                        Mean   :-122.1   Mean   :47.68
##                                        3rd Qu.:-122.0   3rd Qu.:47.70
##                                        Max.   :-121.9   Max.   :47.73
##  building_grade  square_feet_total_living    bedrooms      bath_full_count
##  Min.   : 2.00   Min.   :  240            Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 8.00   1st Qu.: 1820            1st Qu.: 3.000   1st Qu.: 1.000
##  Median : 8.00   Median : 2420            Median : 4.000   Median : 2.000
##  Mean   : 8.24   Mean   : 2540            Mean   : 3.479   Mean   : 1.798
##  3rd Qu.: 9.00   3rd Qu.: 3110            3rd Qu.: 4.000   3rd Qu.: 2.000
##  Max.   :13.00   Max.   :13540            Max.   :11.000   Max.   :23.000
##  bath_half_count  bath_3qtr_count   year_built   year_renovated
##  Min.   :0.0000   Min.   :0.000   Min.   :1900   Min.   :   0.00
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1979   1st Qu.:   0.00
##  Median :1.0000   Median :0.000   Median :1998   Median :   0.00
##  Mean   :0.6134   Mean   :0.494   Mean   :1993   Mean   :  26.24
##  3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:2007   3rd Qu.:   0.00
##  Max.   :8.0000   Max.   :8.000   Max.   :2016   Max.   :2016.00
##  current_zoning       sq_ft_lot        prop_type          present_use
##  Length:12865       Min.   :    785   Length:12865       Min.   :  0.000
##  Class :character   1st Qu.:   5355   Class :character   1st Qu.:  2.000
##  Mode  :character   Median :   7965   Mode  :character   Median :  2.000
##                     Mean   :  22229                      Mean   :  6.598
##                     3rd Qu.:  12632                      3rd Qu.:  2.000
##                     Max.   :1631322                      Max.   :300.000
```

```
str(housing_df)
```

```
## ’data.frame’:    12865 obs. of  24 variables:
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ Sale.Price              : int  698000 649990 572500 420000 369900 184667 1050000 875000 660000 65
##  $ sale_reason             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ sale_instrument         : int  3 3 3 3 3 15 3 3 3 3 ...
##  $ sale_warning            : chr  "" "" "" "" ...
##  $ sitetype                : chr  "R1" "R1" "R1" "R1" ...
##  $ addr_full               : chr  "17021 NE 113TH CT" "11927 178TH PL NE" "13315 174TH AVE NE" "3303
##  $ zip5                    : int  98052 98052 98052 98052 98052 98053 98053 98053 98053 98052 ...
##  $ ctyname                 : chr  "REDMOND" "REDMOND" "" "REDMOND" ...
##  $ postalctyn              : chr  "REDMOND" "REDMOND" "REDMOND" "REDMOND" ...
##  $ lon                     : num  -122 -122 -122 -122 -122 ...
##  $ lat                     : num  47.7 47.7 47.7 47.6 47.7 ...
##  $ building_grade          : int  9 9 8 8 7 7 10 10 9 8 ...
##  $ square_feet_total_living: int  2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ bedrooms                : int  4 4 4 3 3 4 5 4 4 4 ...
##  $ bath_full_count         : int  2 2 1 1 1 2 3 2 2 1 ...
##  $ bath_half_count         : int  1 0 1 0 0 1 0 1 1 0 ...
##  $ bath_3qtr_count         : int  0 1 1 1 1 1 1 0 1 1 ...
##  $ year_built              : int  2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 ...
##  $ year_renovated          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr  "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : int  6635 5570 8444 9600 7526 7280 97574 30649 42688 94889 ...
##  $ prop_type               : chr  "R" "R" "R" "R" ...
##  $ present_use             : int  2 2 2 2 2 2 2 2 2 2 ...
```

```
housing_df %>%
  filter(zip5 == 98074 | zip5 == 98059 ) %>%
  mutate(year = format(as.Date(Sale.Date, format="%m/%d/%Y"),"%Y")) %>%
  mutate(month = format(as.Date(Sale.Date, format="%m/%d/%Y"),"%m")) %>%
  group_by(year,month) %>%
  summarize(average_sale_price=max(Sale.Price),sdev_sale_price=sd(Sale.Price)) %>%
  arrange(year)
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 49 x 4
## # Groups:   year [11]
##    year  month average_sale_price sdev_sale_price
##    <chr> <chr>              <int>           <dbl>
##  1 2006  04               1369900              NA
##  2 2006  05               1389900              NA
##  3 2006  06               1650000              NA
##  4 2006  07                875000              NA
##  5 2006  09               1230000              NA
##  6 2006  10               1020000              NA
##  7 2006  12               1099900              NA
##  8 2007  01                552000              NA
##  9 2007  03                950000              NA
## 10 2007  04               1035000          60104.
## # ... with 39 more rows
```

```
# Number of iteration count per value.
housing_df %>% count(sale_reason)
```

```
##    sale_reason     n
## 1            0     2
## 2            1 12202
## 3            2     1
## 4            3     1
## 5            4   134
## 6            6     1
## 7            7     3
## 8            8   152
## 9           10    10
## 10          11     1
## 11          12    65
## 12          13     2
## 13          14    49
## 14          16     3
## 15          17     1
## 16          18   235
## 17          19     3
```

```
# Report for all transactions with sales_reason 1 for all bedrooms, to understand market demand over th
# will summarize min/max/mean/sd on yearly basic.

housing_df %>%
  filter( sale_reason == 1 ) %>%
  mutate(year = format(as.Date(Sale.Date, format="%m/%d/%Y"),"%Y")) %>%
  group_by(year,bedrooms) %>%
  summarize(average_sale_price=mean(Sale.Price),sdev_sale_price=sd(Sale.Price),max_sale_price=max(Sale.
  arrange(year,bedrooms) %>%  print(n=100)
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 86 x 6
## # Groups:   year [11]
##    year  bedrooms average_sale_price sdev_sale_price max_sale_price min_sale_p~1
##    <chr>    <int>              <dbl>           <dbl>          <int>        <int>
##  1 2006         0            1125000         247437.        1390000       900000
##  2 2006         1             435000             NA          435000       435000
##  3 2006         2             510792.        176413.        1588359       240000
##  4 2006         3             526003.        184419.        1595000        31272
##  5 2006         4             723984.        321164.        3000000        76777
##  6 2006         5             711207.        252116.        1750000       200000
##  7 2006         6             772293.        482437.        1772500        32000
##  8 2006         7             479950         183777.         609900       350000
##  9 2006         8            1185000             NA         1185000      1185000
## 10 2006         9             413000             NA          413000       413000
## 11 2007         0            1640000             NA         1640000      1640000
## 12 2007         1            1106238.        499964.        1600000       465000
## 13 2007         2             559006.        158574.        1250000       279556
## 14 2007         3             584269.        209552.        1830000        67500
## 15 2007         4             775123.        378502.        2625000         1000
## 16 2007         5             887050.        534729.        2988000         5000
## 17 2007         6            1083342.        765964.        2625000       580000
## 18 2007         7             225000             NA          225000       225000
## 19 2007        10             450000             NA          450000       450000
## 20 2007        11            1825000             NA         1825000      1825000
## 21 2008         2             994685.       1019955.        3175000        45000
## 22 2008         3             900992.        908376.        3175000       240000
## 23 2008         4             727270.        289445.        3150000         1500
## 24 2008         5             762554.        260731.        1850000       320000
## 25 2008         6            1087516.        646626.        2189000       510000
## 26 2008         7            2992500        1417749.        3995000      1990000
## 27 2009         0             745000             NA          745000       745000
## 28 2009         1            2000000             NA         2000000      2000000
## 29 2009         2             420414.        152480.         960000          873
## 30 2009         3             465573.        133746.        1149542          998
## 31 2009         4             629498.        214076.        2000000        54321
## 32 2009         5             684959.        203880.        1585000       376000
## 33 2009         6             669750         207429.         900000       429000
## 34 2010         0             985000          98995.        1055000       915000
## 35 2010         1             808500         425984.        1300000       260000
## 36 2010         2             452440.        213384.        2300000       229000
## 37 2010         3             520596.        346398.        4400000          698
## 38 2010         4             655932.        248312.        2300000          698
## 39 2010         5             771397.        377512.        2300000       266900
## 40 2010         6             349333.        297539.         532000         6000
## 41 2010         7             435000             NA          435000       435000
## 42 2011         0             330535             NA          330535       330535
## 43 2011         1            452667.         103079.         520000       334000
## 44 2011         2            431706.         133458.         745391       120527
## 45 2011         3            466319.         276133.        4380542         4000
## 46 2011         4            761561.         750334.        4380542       120527
## 47 2011         5           1215543.        1210192.        4380542       275000
## 48 2011         6            622300          414116.        1230000       220000
## 49 2012         0            150000              NA          150000       150000
## 50 2012         2            468312.         159325.        1075000       155000
## 51 2012         3            509565.         387291.        3462000        50000
## 52 2012         4            718861.         530684.        3462000         4059
## 53 2012         5            739227.         404094.        2500000       237500
## 54 2012         6            624300          233853.         900000       410000
## 55 2012         7            285199              NA          285199       285199
## 56 2013         0           1300000              NA         1300000      1300000
## 57 2013         1            779938.         619802.        1586000       250000
## 58 2013         2            503984.         137289.         900000       150000
## 59 2013         3            533252.         170546.        1586000        35000
## 60 2013         4            674855.         208244.        3340000       185000
## 61 2013         5            757115.         351830.        2750000         2500
## 62 2013         6            792572.         326728.        1268000       525000
## 63 2013         7            955000          487904.        1300000       610000
## 64 2013         9            750000              NA          750000       750000
## 65 2014         0           1295648              NA         1295648      1295648
## 66 2014         1            445625          143633.         620000       278500
## 67 2014         2            522034.         171051.         930000         8000
## 68 2014         3            547172.         191762.        1700000         7000
## 69 2014         4            755985.         219701.        2140000         8000
## 70 2014         5            806569.         219001.        1750000       385000
## 71 2014         6            603395           82966.         779950       510000
## 72 2014         7           2280000              NA         2280000      2280000
## 73 2015         0            743000              NA          743000       743000
## 74 2015         2            586761.         223273.        2150000       167231
## 75 2015         3            601230.         195038.        2025000        91049
## 76 2015         4            821093.         244079.        2300000       110000
## 77 2015         5            862726.         294018.        2200000        18000
## 78 2015         6            838199.         165699.        1100000       550000
## 79 2015         8           1060000              NA         1060000      1060000
## 80 2016         0            744987.         251085.         953830       413617
## 81 2016         1            749081.         733849.        2165000        69484
## 82 2016         2            604059.         204172.        1840000       150000
## 83 2016         3            708982.         353209.        4311000        37800
## 84 2016         4            896035.         308560.        3750000       170000
## 85 2016         5           1072021.         585099.        3950000       244040
## 86 2016         6            727703.         176919.         917500       380126
## # ... with abbreviated variable name 1: min_sale_price
```

```
# Report for all transactions with sales_reason 18 and 2 BHK, to understand 2BKH market trend
# will summarize min/max/mean/sd on yearly basic.

housing_df %>%
  filter( sale_reason == 18 & bedrooms == 2 ) %>%
  mutate(year = format(as.Date(Sale.Date, format="%m/%d/%Y"),"%Y")) %>%
  group_by(year,bedrooms) %>%
  summarize(average_sale_price=mean(Sale.Price),sdev_sale_price=sd(Sale.Price),max_sale_price=max(Sale.
  arrange(year,bedrooms) %>%  print(n=100)
```

```
## ‘summarise()‘ has grouped output by ’year’. You can override using the
## ‘.groups‘ argument.
```

```
## # A tibble: 11 x 6
## # Groups:   year [11]
##    year  bedrooms average_sale_price sdev_sale_price max_sale_price min_sale_p~1
##    <chr>    <int>              <dbl>           <dbl>          <int>        <int>
##  1 2006         2             130058          78988.         185911        74205
##  2 2007         2              84360.         50748.         120244        48475
##  3 2008         2             160214          13839.         170000       150428
##  4 2009         2             340000         127279.         430000       250000
##  5 2010         2             575000             NA          575000       575000
##  6 2011         2             336911.        221310.         933742        90000
##  7 2012         2             346044.        216251.         709222       164000
##  8 2013         2             386675         207467.         688700       231000
##  9 2014         2             363050          31183.         385100       341000
## 10 2015         2             543333.         20207.         555000       520000
## 11 2016         2             496167.        213840.         835000       265000
## # ... with abbreviated variable name 1: min_sale_price
```

```
# Assignment 02 - Using the purrr package - perform 2 functions on your dataset.
```

```
# you could use zip_n, keep, discard, compact, etc.
# Reference link - https://www.r-bloggers.com/2020/05/one-stop-tutorial-on-purrr-package-in-r/
# Reference link - https://hookedondata.org/posts/2019-01-09_going-off-the-map-exploring-purrrs-other-f

# The easiest way - install the tidyverse
# install.packages("tidyverse")
# Install just purrr
# install.packages("purrr"

# map() - Use if you want to apply a function to each element of the list or a vector.
# map2() - Use if you’re going to apply a function to a pair of elements from two different lists or ve
# pmap() - Use if you need to apply a function to a group of elements from a list of lists.

# Converting sq feet to sq meter.
sq_meter <- function(x){
  return(x/10.764)
}
# Create a vector of number
sq_meter_vector1 <- housing_df %>% select(square_feet_total_living) %>% head()
head(sq_meter_vector1)
```

```
##   square_feet_total_living
## 1                     2810
## 2                     2880
## 3                     2770
## 4                     1620
## 5                     1440
## 6                     4160
```

```
# Using map() function to generate squares
map(sq_meter_vector1, sq_meter)                     #-- Shortan the result.
```

```
## $square_feet_total_living
## [1] 261.0554 267.5585 257.3393 150.5017 133.7793 386.4734
```

```
#str(housing_df)

# without function example.
v1 <- housing_df %>% select(square_feet_total_living) %>% head()
v2 <- housing_df %>% select(zip5) %>% head()
head(v1)
```

```
##   square_feet_total_living
## 1                     2810
## 2                     2880
## 3                     2770
## 4                     1620
## 5                     1440
## 6                     4160
```

```
head(v2)
```

```
##    zip5
## 1 98052
## 2 98052
## 3 98052
## 4 98052
## 5 98052
## 6 98053
```

```
map2(v1, v2, ~ .x + .y)
```

```
## $square_feet_total_living
## [1] 100862 100932 100822  99672  99492 102213
```

```
# keep() - A handy function, as the same suggests, using this function,
# we can observe only those elements in the list which pass a logic
ls2 <- list(23, 12, 14, 7, 2, 0, 24, 98)
keep(ls2, function(x) x > 5)
```

```
## [[1]]
## [1] 23
##
## [[2]]
## [1] 12
##
## [[3]]
## [1] 14
##
```

```
## [[4]]
## [1] 7
##
## [[5]]
## [1] 24
##
## [[6]]
## [1] 98
```

```
sales_v1 <- housing_df %>% select(Sale.Price) %>% head()
#sales_v1
sales_v1 %>% keep(function(x) mean(x) > 400000)
```

```
##   Sale.Price
## 1     698000
## 2     649990
## 3     572500
## 4     420000
## 5     369900
## 6     184667
```

```
# discard() - The function drops those values which fail to pass the logical tests.
# Say we want to drop NA values then you can use is.na()to discard observations which are represented N

ls3 <- list(23, NA, 14, 7, NA, NA, 24, 98)
discard(ls3, is.na)
```

```
## [[1]]
## [1] 23
##
## [[2]]
## [1] 14
##
## [[3]]
## [1] 7
##
## [[4]]
## [1] 24
##
## [[5]]
## [1] 98
```

```
# compact() - A simple, straightforward function that drops all the NULL values present in the list.
# Please do not confuse NA values with that of NULL values. These are two different types in R.

ls4 <- list(23, NULL, NA, 34)
compact(ls4)
```

```
## [[1]]
## [1] 23
##
## [[2]]
## [1] NA
##
## [[3]]
## [1] 34
```

```
# Assignment - Use the tibble,cbind and rbind function on your dataset
# Joins reference - https://www.datasciencemadesimple.com/join-in-r-merge-in-r/
# Join assignments left_join,right_join,inner_join,full_join,semi_join,anti_join()
```

```
hvector_01 <- housing_df %>% select(Sale.Date,Sale.Price,square_feet_total_living,zip5)
hvector_02 <- housing_df %>% select(Sale.Date,bedrooms,year_built,year_renovated,current_zoning,sq_ft_lo
head(hvector_01)
```

```
##   Sale.Date Sale.Price square_feet_total_living  zip5
## 1  1/3/2006     698000                     2810 98052
## 2  1/3/2006     649990                     2880 98052
## 3  1/3/2006     572500                     2770 98052
## 4  1/3/2006     420000                     1620 98052
## 5  1/3/2006     369900                     1440 98052
## 6  1/3/2006     184667                     4160 98053
```

```
str(hvector_02)
```

```
## 'data.frame':    12865 obs. of  6 variables:
##  $ Sale.Date     : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ bedrooms      : int  4 4 4 3 3 4 5 4 4 4 ...
##  $ year_built    : int  2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 ...
##  $ year_renovated: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning: chr  "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot     : int  6635 5570 8444 9600 7526 7280 97574 30649 42688 94889 ...
```

```
# cbind()
hvector_col <- cbind(hvector_01,hvector_02)
str(hvector_col)
```

```
## 'data.frame':    12865 obs. of  10 variables:
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ Sale.Price              : int  698000 649990 572500 420000 369900 184667 1050000 875000 660000 65
##  $ square_feet_total_living: int  2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ zip5                    : int  98052 98052 98052 98052 98052 98053 98053 98053 98053 98052 ...
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ bedrooms                : int  4 4 4 3 3 4 5 4 4 4 ...
##  $ year_built              : int  2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 ...
##  $ year_renovated          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr  "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : int  6635 5570 8444 9600 7526 7280 97574 30649 42688 94889 ...
```

```
head(hvector_col)
```

```
##   Sale.Date Sale.Price square_feet_total_living  zip5 Sale.Date bedrooms
## 1  1/3/2006     698000                     2810 98052  1/3/2006        4
## 2  1/3/2006     649990                      2880 98052  1/3/2006        4
## 3  1/3/2006     572500                      2770 98052  1/3/2006        4
## 4  1/3/2006     420000                      1620 98052  1/3/2006        3
## 5  1/3/2006     369900                      1440 98052  1/3/2006        3
## 6  1/3/2006     184667                      4160 98053  1/3/2006        4
##   year_built year_renovated current_zoning sq_ft_lot
## 1       2003              0             R4      6635
## 2       2006              0             R4      5570
## 3       1987              0             R6      8444
## 4       1968              0             R4      9600
## 5       1980              0             R6      7526
## 6       2005              0          URPSO      7280
```

```
# tibble()
hvector_01 <- tibble(housing_df %>% select(Sale.Date,Sale.Price,square_feet_total_living,zip5))
hvector_02 <- tibble(housing_df %>% select(Sale.Date,bedrooms,year_built,year_renovated,current_zoning,
hvector_col <- cbind(hvector_01,hvector_02)
str(hvector_col)
```

```
## 'data.frame':    12865 obs. of  10 variables:
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ Sale.Price              : int  698000 649990 572500 420000 369900 184667 1050000 875000 660000 65
##  $ square_feet_total_living: int  2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ zip5                    : int  98052 98052 98052 98052 98052 98053 98053 98053 98053 98052 ...
##  $ Sale.Date               : chr  "1/3/2006" "1/3/2006" "1/3/2006" "1/3/2006" ...
##  $ bedrooms                : int  4 4 4 3 3 4 5 4 4 4 ...
##  $ year_built              : int  2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 ...
##  $ year_renovated          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr  "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : int  6635 5570 8444 9600 7526 7280 97574 30649 42688 94889 ...
```

```
head(hvector_col)
```

```
##   Sale.Date Sale.Price square_feet_total_living  zip5 Sale.Date bedrooms
## 1  1/3/2006     698000                      2810 98052  1/3/2006        4
## 2  1/3/2006     649990                      2880 98052  1/3/2006        4
## 3  1/3/2006     572500                      2770 98052  1/3/2006        4
## 4  1/3/2006     420000                      1620 98052  1/3/2006        3
## 5  1/3/2006     369900                      1440 98052  1/3/2006        3
## 6  1/3/2006     184667                      4160 98053  1/3/2006        4
##   year_built year_renovated current_zoning sq_ft_lot
## 1       2003              0             R4      6635
## 2       2006              0             R4      5570
## 3       1987              0             R6      8444
## 4       1968              0             R4      9600
## 5       1980              0             R6      7526
## 6       2005              0          URPSO      7280
```

```
# for rbind rows and cols must be same.
# hvector_row <- rbind(hvector_01,hvector_02)
# str(hvector_row)
# head(hvector_row)
# Joins
hvector_01 <- housing_df %>% select(Sale.Date,Sale.Price,square_feet_total_living,zip5) %>% filter(zip5=
hvector_02 <- housing_df %>% select(Sale.Date,bedrooms,year_built,year_renovated,current_zoning,sq_ft_lo

# Get unique values from column.
unique(hvector_col$zip5)
```

```
## [1] 98052 98053 98074 98059
```

```
# Join assignments left_join,right_join,inner_join,full_join,semi_join,anti_join()
# https://www.datasciencemadesimple.com/join-in-r-merge-in-r/

left_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>% nrow
```

```
## [1] 452
```

```
right_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>% nrow
```

```
## [1] 12889
```

```
inner_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>% nrow
```

```
## [1] 452
```

```
anti_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>% nrow
```

```
## [1] 0
```

```
semi_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>% nrow
```

```
## [1] 73
```

```
left_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>%
  distinct(zip5) %>%
  head()
```

```
##    zip5
## 1 98074
```

```
right_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>%
  distinct(zip5,current_zoning,year_built) %>%
  head()
```

```
##    zip5 year_built current_zoning
## 1 98074       2005             R1
## 2 98074       2006          URPSO
## 3 98074       2006             R4
## 4 98074       1969             R5
## 5 98074       1987             R4
## 6 98074       1990            RA5
```

```r
inner_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>%
  distinct(zip5) %>%
  head()
```

```
##    zip5
## 1 98074
```

```r
anti_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>%
  distinct(zip5) %>%
  head()
```

```
## [1] zip5
## <0 rows> (or 0-length row.names)
```

```r
semi_join(hvector_01,hvector_02, by=c("Sale.Date" = "Sale.Date")) %>%
  distinct(zip5,square_feet_total_living,Sale.Price) %>%
  head()
```

```
##   Sale.Price square_feet_total_living  zip5
## 1    1369900                     4630 98074
## 2    1389900                     4330 98074
## 3    1650000                     5640 98074
## 4     875000                     3940 98074
## 5    1230000                     5340 98074
## 6    1020000                     3720 98074
```

```r
# Assignment - Split a string, then concatenate the results back together
```

```r
paste("Hello","Funny R ","World !!!")
```

```
## [1] "Hello Funny R  World !!!"
```

```r
# housing_df %>% paste(housing_df$addr_full, sep = " ") %>% head()
paste(housing_df$addr_full, sep = " ") %>% head(.,n=20)
```

```
##  [1] "17021 NE 113TH CT"  "11927 178TH PL NE"  "13315 174TH AVE NE"
##  [4] "3303 178TH AVE NE"  "16126 NE 108TH CT"  "8101 229TH DR NE"
##  [7] "21634 NE 87TH PL"   "21404 NE 67TH ST"   "7525 238TH AVE NE"
## [10] "17703 NE 26TH ST"   "14924 NE 74TH CT"   "7858 148TH CT NE"
## [13] "17905 NE 26TH ST"   "2921 288TH AVE NE"  "3624 264TH AVE NE"
## [16] "7850 148TH CT NE"   "8944 237TH PL NE"   "11922 173RD PL NE"
## [19] "3201 176TH CT NE"   "26920 NE 50TH ST"
```

```r
# sprintf
lang <- "R"
course <- "DSC-520"
sprintf(" %s : Statistics for Data Science, using %s studio !! ",course,lang)
```

```
## [1] " DSC-520 : Statistics for Data Science, using R studio !! "
```

```
# str_split
Address <- str_split(housing_df$addr_full, pattern = " ") %>% head()
str(Address)
```

```
## List of 6
##  $ : chr [1:4] "17021" "NE" "113TH" "CT"
##  $ : chr [1:4] "11927" "178TH" "PL" "NE"
##  $ : chr [1:4] "13315" "174TH" "AVE" "NE"
##  $ : chr [1:4] "3303" "178TH" "AVE" "NE"
##  $ : chr [1:4] "16126" "NE" "108TH" "CT"
##  $ : chr [1:4] "8101" "229TH" "DR" "NE"
```

```
# str_sub
unique(housing_df$year_renovated)
```

```
##  [1]    0 2004 1985 1978 2006 2002 1989 2000 2007 2012 1975 1998 2001 2003 1986
## [16] 2009 1993 1991 1990 1983 2016 2008 1997 1999 1987 1995 2005 1992 2014 1994
## [31] 1982 2010 1974 1970 2011 1981 1984 2015 1980 1988
```

```
# housing_df %>% select(Sale.Date,Sale.Price,year_built,year_renovated,addr_full) %>%
str_sub(string = housing_df$year_renovated, start=1, end=4) %>% head(.,n=100)
```

```
##   [1] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [11] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [21] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [31] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [41] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [51] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [61] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [71] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [81] "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"   
##  [91] "0"    "2004" "0"    "0"    "0"    "0"    "0"    "0"    "0"    "0"
```

```
test_df <- housing_df %>% select(Sale.Date,Sale.Price,year_built,year_renovated,addr_full)
test_df[str_sub(string=housing_df$year_renovated, start=1, end=5) == 2006, c("Sale.Date","Sale.Price","
```

```
##        Sale.Date Sale.Price year_built year_renovated              addr_full
## 343     4/6/2006     220000       1954           2006 3026 W AMES LAKE DR NE
## 960    8/18/2006     525000       1968           2006      16809 NE 106TH ST
## 1426  12/13/2006    1168000       1985           2006      16408 NE 132ND ST
## 7589   11/7/2012     912000       1975           2006       3402 181ST PL NE
## 8715   8/20/2013     435000       1954           2006 3026 W AMES LAKE DR NE
## 11086  7/10/2015     850000       1967           2006       17705 NE 24TH ST
```

```
test_2006 <- test_df[str_sub(string=housing_df$year_renovated, start=1, end=5) == 2006, c("Sale.Date","
str(test_2006)
```

```
## ’data.frame’:    6 obs. of  5 variables:
##  $ Sale.Date     : chr  "4/6/2006" "8/18/2006" "12/13/2006" "11/7/2012" ...
##  $ Sale.Price    : int  220000 525000 1168000 912000 435000 850000
##  $ year_built    : int  1954 1968 1985 1975 1954 1967
##  $ year_renovated: int  2006 2006 2006 2006 2006 2006
##  $ addr_full     : chr  "3026 W AMES LAKE DR NE" "16809 NE 106TH ST" "16408 NE 132ND ST" "3402 181ST
```

```
# str_detect
# Get distinct count for each value in the column.
housing_df %>% count(year_renovated)
```

```
##    year_renovated     n
## 1               0 12696
## 2            1970     3
## 3            1974     2
## 4            1975     3
## 5            1978     2
## 6            1980     2
## 7            1981     1
## 8            1982     1
## 9            1983     1
## 10           1984     2
## 11           1985     5
## 12           1986     5
## 13           1987     3
## 14           1988     1
## 15           1989     7
## 16           1990     6
## 17           1991     5
## 18           1992     2
## 19           1993     3
## 20           1994     3
## 21           1995     8
## 22           1997     4
## 23           1998     9
## 24           1999     6
## 25           2000     8
## 26           2001     6
## 27           2002     6
## 28           2003     7
## 29           2004     5
## 30           2005     8
## 31           2006     6
## 32           2007    16
## 33           2008     1
## 34           2009     3
## 35           2010     1
## 36           2011     2
## 37           2012     7
## 38           2014     6
## 39           2015     1
## 40           2016     2
```

```
test_df <- housing_df %>% select(Sale.Date,Sale.Price,year_built,year_renovated,addr_full)
test_df[str_detect(string=housing_df$year_renovated, pattern = "2007"), c("Sale.Date","Sale.Price","yea
```

```
##         Sale.Date Sale.Price year_built year_renovated
## 716     6/27/2006     510000       1967           2007
## 1125    9/25/2006     435000       1986           2007
## 1150   9/28/2006     375000       1964           2007
## 1153   10/2/2006     870000       1975           2007
## 1278   11/3/2006     684000       1975           2007
## 1739   3/28/2007     617000       1986           2007
## 2138    7/2/2007     975000       1975           2007
## 3810   4/10/2009      29537       1965           2007
## 5228   8/31/2010     749000       1988           2007
## 5818    5/2/2011     625000       1959           2007
## 8763   8/29/2013    1650000       1972           2007
## 9815   7/14/2014     850000       1975           2007
## 11342  9/10/2015    1060000       1975           2007
## 12236  6/27/2016     676000       1967           2007
## 12817 11/28/2016     900000       1977           2007
## 12818 11/28/2016     900000       1977           2007
##                           addr_full
## 716               16228 NE 100TH ST
## 1125              11729 201ST PL NE
## 1150              9514 167TH AVE NE
## 1153              8020 252ND AVE NE
## 1278               3324 181ST PL NE
## 1739              11729 201ST PL NE
## 2138               3324 181ST PL NE
## 3810              5840 156TH AVE NE
## 5228              4610 244TH AVE NE
## 5818              7834 134TH AVE NE
## 8763  2608 W LAKE SAMMAMISH PKWY NE
## 9815              8020 252ND AVE NE
## 11342              3324 181ST PL NE
## 12236             6611 238TH AVE NE
## 12817             2620 186TH AVE NE
## 12818             2620 186TH AVE NE
```

```
# str_replace
test_df <- housing_df %>% select(Sale.Date,Sale.Price,year_built,year_renovated,addr_full)
str_replace(housing_df$addr_full,"NE","Nebraska") %>% head()
```

```
## [1] "17021 Nebraska 113TH CT"  "11927 178TH PL Nebraska"
## [3] "13315 174TH AVE Nebraska" "3303 178TH AVE Nebraska"
## [5] "16126 Nebraska 108TH CT"  "8101 229TH DR Nebraska"
```