Homework 1

NAME

DATE

Here's your chance to demonstrate that you can integrate the topics and skills you learned so far into a literate report.

When the question asks you to perform a coding task, insert a code chunk after each question where you will write the code to answer that question. Knitting after each completed code chunk will help you to ensure your final product works as intended! That way if it breaks, you know exactly where the error lies. It's like saving after every answer!

The first question is done for you as an example.

Object assignments

0. Calculate 3 + 4. Put the answer in the grey area below and knit the document. Make sure you can find this code and output in the resulting HTML file.

```
3+4
```

```
## [1] 7
```

1. Calculate 2^5 by typing this mathematical expression in the code chunk below and then knit the document.

```
2^5
```

```
## [1] 32
```

2. What are the values after each statement in the following?

```
(mass <- 47.5)            # a
```

```
## [1] 47.5
```

```
(age  <- 122 )            # b
```

```
## [1] 122
```

```
(mass <- mass * 2.0)      # c
```

```
## [1] 95
```

```
(age  <- age - 20)        # d
```

```
## [1] 102
```

```
(mass_index <- mass/age)  # e
```

```
## [1] 0.9313725
```

a. `mass` = 47.5
b. `age` = 122

c. mass = 95
d. age = 102
e. mass_index = 0.9313725

3. Assign a numeric value to the variable my_apples, assign a different numeric value to the variable my_oranges. Add these two together and assign the result to the variable my_fruit. Print the result of my_fruit to the report.

```
my_apples <- 6
my_oranges <- 19
my_fruit <- my_apples + my_oranges
my_fruit
```

```
## [1] 25
```

4. What is the data type of the variable my_fruit?

```
class(my_fruit)
```

```
## [1] "numeric"
```

Its class is "numeric"

5. A variable that classified as logical can hold only what two values? These are also called Boolean variables. (You don't need to write code to answer this question.)

TRUE and FALSE

6. Knit this document.

Vectors()

1. Use the class() function to explore what happens when you have vectors of different data types. The first one has been done for you. Report your answer for each in a sentence below.

```
num_only <- c(1,2,3,4)                    #a
char_only <- c("a","b","c","d")          #b
num_char <- c(1, 2, 3, "a")               #c
num_logical <- c(1, 2, 3, TRUE)           #d
char_logical <- c("a", "b", "c", TRUE)    #e
tricky <- c(1, 2, 3, "4")                 #f
class(num_only)
```

```
## [1] "numeric"
```

```
class(char_only)
```

```
## [1] "character"
```

```
class(num_char)
```

```
## [1] "character"
```

```
class(num_logical)
```

```
## [1] "numeric"
```

```
class(char_logical)
```

```
## [1] "character"
```

```
class(tricky)
```

```
## [1] "character"
```

- a. The `num_only` vector contains only numeric values, and it's class is numeric.
- b. The `char_only` vector contains only character values, and it's class is character.
- c. The `num_char` vector contains numeric and character values, and it's class is character.
- d. The `num_logical` vector contains numeric and Boolean values, and it's class is numeric.
- e. The `char_logical` vector contains character and Boolean values, and it's class is character.
- f. The `tricky` vector contains three numeric value and one character value("4"), and it's class is character.

Let's go to Vegas!

2. Create three vectors (`weekday`, `poker`, `roulette`) using the `c()` operator to describe the following outcome.

- `weekday`: The 5 weekdays. Use these three letter codes: "mon", "tue", "wed", "thu", "fri."
- `poker`:
  - On Monday you won $140
  - Tuesday you lost $50
  - Wednesday you won $20
  - Thursday you lost $120
  - Friday you won $240
- `roulette`:
  - On Monday you lost $24
  - Tuesday you lost $50
  - Wednesday you won $100
  - Thursday you lost $350
  - Friday you won $10

Hint: If I won $30 on monday and lost $20 on tuesday, my vector would look like `c(30, -20)`

```
weekday <- c("mon", "tue", "wed", "thu", "fri")
poker <- c(140, -50, 20, -120, 240)
roulette <- c(-24, -50, 100, -350, 10)
```

3. Use the `sum()` function to calculate how much money did you gain/lose on each game. What game did you do better on?

```
sum(poker)
```

```
## [1] 230
```

```
sum(roulette)
```

```
## [1] -314
```

```
sum(poker) > sum(roulette)
```

```
## [1] TRUE
```

- We gained `$230` on poker and lost `$314` on roulette. We did better on poker game.

4. Add the results of both vectors together, then calculate your net gain over the weekend. Did you come out ahead?

```
sum(poker+roulette)
```

```
## [1] -84
```

We came out behind.

5. On which days did you make money on poker? *Hint: subset the `weekday` vector using a logical statement about `poker`*

```
weekday[poker>0]
```

```
## [1] "mon" "wed" "fri"
```

6. On which days did you do better on poker than roulette? *Hint: compare the two vectors using a statement of inequality*

```
weekday[poker > roulette]
```

```
## [1] "mon" "thu" "fri"
```

7. Knit this document to make sure it still works.

Data Frames

Run the code chunk below to read in the Ames data set from the web. This data set is on all residential home sales in Ames, Iowa between 2006 and 2010. The data set contains many explanatory variables on the quality and quantity of physical attributes of residential homes in Iowa sold between 2006 and 2010. Most of the variables describe information a typical home buyer would like to know about a property (square footage, number of bedrooms and bathrooms, size of lot, etc.). A detailed discussion of variables can be found in the original paper: De Cock D. 2011. Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project. Journal of Statistics Education; 19(3).

```
ames <- openintro::ames
head(ames)
```

```
## # A tibble: 6 x 82
##   Order    PID  area  price MS.SubClass MS.Zoning Lot.Frontage Lot.Area Street
##   <int>  <int> <int>  <int>       <int> <fct>            <int>    <int> <fct>
## 1     1 5.26e8  1656 215000          20 RL                 141    31770 Pave
## 2     2 5.26e8   896 105000          20 RH                  80    11622 Pave
## 3     3 5.26e8  1329 172000          20 RL                  81    14267 Pave
## 4     4 5.26e8  2110 244000          20 RL                  93    11160 Pave
## 5     5 5.27e8  1629 189900          60 RL                  74    13830 Pave
## 6     6 5.27e8  1604 195500          60 RL                  78     9978 Pave
## # ... with 73 more variables: Alley <fct>, Lot.Shape <fct>, Land.Contour <fct>,
## #   Utilities <fct>, Lot.Config <fct>, Land.Slope <fct>, Neighborhood <fct>,
## #   Condition.1 <fct>, Condition.2 <fct>, Bldg.Type <fct>, House.Style <fct>,
## #   Overall.Qual <int>, Overall.Cond <int>, Year.Built <int>,
## #   Year.Remod.Add <int>, Roof.Style <fct>, Roof.Matl <fct>,
## #   Exterior.1st <fct>, Exterior.2nd <fct>, Mas.Vnr.Type <fct>,
## #   Mas.Vnr.Area <int>, Exter.Qual <fct>, Exter.Cond <fct>, Foundation <fct>,
## #   Bsmt.Qual <fct>, Bsmt.Cond <fct>, Bsmt.Exposure <fct>,
## #   BsmtFin.Type.1 <fct>, BsmtFin.SF.1 <int>, BsmtFin.Type.2 <fct>,
## #   BsmtFin.SF.2 <int>, Bsmt.Unf.SF <int>, Total.Bsmt.SF <int>, Heating <fct>,
## #   Heating.QC <fct>, Central.Air <fct>, Electrical <fct>, X1st.Flr.SF <int>,
## #   X2nd.Flr.SF <int>, Low.Qual.Fin.SF <int>, Bsmt.Full.Bath <int>,
## #   Bsmt.Half.Bath <int>, Full.Bath <int>, Half.Bath <int>,
## #   Bedroom.AbvGr <int>, Kitchen.AbvGr <int>, Kitchen.Qual <fct>,
## #   TotRms.AbvGrd <int>, Functional <fct>, Fireplaces <int>,
## #   Fireplace.Qu <fct>, Garage.Type <fct>, Garage.Yr.Blt <int>,
## #   Garage.Finish <fct>, Garage.Cars <int>, Garage.Area <int>,
## #   Garage.Qual <fct>, Garage.Cond <fct>, Paved.Drive <fct>,
## #   Wood.Deck.SF <int>, Open.Porch.SF <int>, Enclosed.Porch <int>,
## #   X3Ssn.Porch <int>, Screen.Porch <int>, Pool.Area <int>, Pool.QC <fct>,
## #   Fence <fct>, Misc.Feature <fct>, Misc.Val <int>, Mo.Sold <int>,
## #   Yr.Sold <int>, Sale.Type <fct>, Sale.Condition <fct>
```

1. List out the variable name and data types for five different variables in the ames data set

Order, integer PID, integer MS. Subclass, integer MS. Zoning, character Lot. Frontage, integer

2. How many observations does the ames data set have? How many variables?

2930 observations and 82 variables.

3. Extract the variable that measures the overall condition of the house (`Overall.Cond`) by position (using bracket index notation `[]`), and by variable names (using `$` notation)

```
ames[,22]
```

```
## # A tibble: 2,930 x 1
##    Overall.Cond
##           <int>
##  1            5
##  2            6
##  3            6
##  4            5
##  5            5
##  6            6
##  7            5
##  8            5
##  9            5
## 10            5
## # ... with 2,920 more rows
```

```
ames$Overall.Cond
```

```
##    [1] 5 6 6 5 5 6 5 5 5 5 5 7 5 5 5 5 7 2 5 6 6 6 5 7 6 6 5 5 5 5 5 5 5 6 6 5 5
##   [38] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 5 5 5 5 5 5 5 5 5 5
##   [75] 5 5 5 5 5 7 5 6 5 5 7 8 5 5 8 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 8 5 6
##  [112] 5 7 5 5 7 6 5 5 5 5 6 5 7 7 6 6 7 6 7 2 5 6 6 8 5 5 6 5 5 5 5 6 6 5 5 5 5
##  [149] 5 7 6 5 7 7 7 7 7 5 5 7 7 8 7 4 7 6 6 7 5 5 7 7 6 7 6 6 7 7 7 7 6 2 8 9 6
##  [186] 9 3 7 7 6 7 6 7 7 7 6 6 7 7 5 5 6 5 8 8 5 5 9 7 5 3 4 5 1 7 8 3 5 4 6 5 8
##  [223] 7 7 5 5 5 5 5 7 4 7 6 5 4 5 7 6 6 7 5 5 3 7 5 5 7 5 6 5 5 5 5 5 8 7 5 5 5
##  [260] 8 7 6 6 5 5 6 5 5 5 5 5 5 5 6 6 5 6 5 6 5 5 5 4 4 6 6 4 3 6 6 6 2 7 6 6 9
##  [297] 8 6 6 5 8 8 5 5 4 7 5 7 5 5 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 3 7 6 4 6
##  [334] 7 7 7 2 5 5 5 5 5 5 7 5 5 5 5 5 5 5 5 5 5 5 5 7 8 5 5 5 5 6 5 5 5 5 5 5 6
##  [371] 7 5 5 7 6 6 7 5 5 5 5 5 7 5 5 5 5 7 6 5 5 6 5 6 8 5 7 6 6 7 6 7 5 6 5 6 8
##  [408] 7 6 5 6 5 5 5 5 6 5 6 5 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
##  [445] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
##  [482] 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 7 6 5 5 5 5 5 5 5 5 5 5 5 5
##  [519] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5 5 6 6 5 5 7 7 4 3 5
##  [556] 6 6 5 5 8 6 5 6 8 8 5 5 5 5 5 5 5 5 5 9 5 5 5 4 6 6 7 6 6 6 7 5 5 5 7 5 5
##  [593] 6 6 5 6 5 5 5 7 7 6 5 6 5 6 6 5 6 6 6 6 6 5 7 4 4 5 5 6 8 7 5 6 5 8 6 6 3
##  [630] 4 7 3 6 8 6 7 5 5 5 5 6 4 7 6 5 5 7 6 8 5 5 7 6 6 5 7 7 8 8 7 5 9 5 6 7 5
##  [667] 5 7 4 5 9 6 7 5 6 7 4 5 5 4 6 6 7 5 5 7 7 5 7 6 5 8 6 8 5 6 8 8 6 4 6 5 8
##  [704] 5 4 7 4 6 4 6 8 8 5 4 7 5 7 7 7 9 3 9 4 6 6 7 5 3 6 6 7 7 6 6 9 8 6 5 7 5
##  [741] 6 7 3 7 6 6 5 6 8 6 8 8 6 1 5 5 7 5 6 6 7 9 5 5 5 1 6 3 6 6 5 6 7 5 5 6 5
##  [778] 5 5 5 2 5 8 6 5 5 5 5 5 5 3 7 6 6 3 5 4 5 5 6 5 8 7 5 5 5 7 5 5 5 5 5 5 5
##  [815] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 5 6 5 5 5 5 5 5 5 5 5 5 5 5 5
##  [852] 5 5 5 7 7 8 7 6 5 8 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 6 5
##  [889] 6 6 6 5 5 9 5 5 3 8 3 5 5 3 8 6 5 4 5 5 7 3 6 3 8 6 9 6 7 8 9 6 7 7 4 7 6
##  [926] 6 6 7 6 7 8 5 6 6 6 6 6 9 9 7 6 5 2 6 5 4 5 5 5 5 6 7 5 5 5 6 5 5 5 5 5 5
##  [963] 5 8 5 5 5 5 5 7 4 5 5 5 6 8 5 4 6 4 8 5 6 5 6 5 5 5 7 8 7 5 5 5 5 5 5 5 5
## [1000] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 2 5 5 6 6 6 7 5 5 7 5 6 6 6 7 5 5 6 6 5 5 8 6
## [1037] 5 5 5 5 5 6 6 5 3 5 6 6 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1074] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5 5 7 6 5 5 5 5
## [1111] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1148] 5 5 5 6 8 5 5 8 5 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 3 9
## [1185] 5 6 6 5 7 7 6 7 6 8 7 5 5 5 6 9 6 8 9 5 5 6 7 6 5 7 6 7 8 5 5 7 6 5 7 6 6
## [1222] 6 7 6 5 8 6 5 6 5 6 9 5 7 7 5 7 4 7 6 6 5 5 7 5 6 6 5 5 7 5 3 5 5 6 8 6 7
## [1259] 5 7 6 6 7 5 6 7 7 7 5 5 6 5 6 6 7 6 5 5 6 7 5 6 5 5 5 8 7 5 7 4 7 6 5 6 5
## [1296] 6 6 6 9 5 7 6 5 6 7 1 9 5 8 8 5 7 5 5 9 9 7 6 5 7 9 1 7 6 5 5 6 7 7 6 5 6
## [1333] 7 9 7 7 7 7 5 7 7 7 7 6 7 5 5 5 8 6 5 7 6 4 5 7 8 7 8 6 6 6 6 7 6 7 7 7 8
## [1370] 3 7 6 5 5 4 7 6 5 5 7 7 7 8 5 5 4 4 3 6 8 8 5 7 6 8 3 7 8 5 5 6 7 6 7 5 5
## [1407] 8 7 7 7 6 5 8 8 7 5 5 6 5 5 5 5 5 5 5 5 5 5 5 6 5 7 6 5 5 5 5 5 5 5 5 5 5
## [1444] 5 6 8 6 6 6 6 6 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5
## [1481] 5 5 5 5 5 7 7 5 6 5 6 6 5 5 6 7 5 7 5 5 5 5 4 5 5 7 8 5 6 6 7 6 7 6 3 6 8
## [1518] 8 5 8 6 6 7 6 7 7 6 5 9 5 3 6 7 6 4 5 6 9 9 9 5 7 6 6 8 8 4 8 6 7 5 7 5 5
## [1555] 7 4 7 4 6 5 5 5 5 5 5 5 6 6 4 5 6 6 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1592] 5 5 6 6 7 7 6 6 9 7 5 7 5 9 8 6 7 6 4 1 6 5 6 6 5 5 6 5 5 5 5 5 5 5 5 5 5
## [1629] 5 5 6 5 5 5 5 5 5 5 5 5 6 5 5 7 5 7 7 6 5 6 6 6 6 6 5 5 5 6 6 7 6 6 7 5 9
## [1666] 6 5 5 5 6 7 6 5 5 5 6 5 5 5 5 5 6 7 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1703] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1740] 5 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 6
## [1777] 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5
## [1814] 5 5 5 7 5 5 6 5 7 5 5 7 5 6 5 7 5 5 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [1851] 5 5 5 5 6 7 8 6 7 8 5 5 5 6 6 6 5 7 6 6 6 7 6 5 6 5 7 5 4 5 5 8 5 8 7 5 7
## [1888] 5 7 7 6 5 7 7 5 7 6 6 5 5 7 3 3 5 8 6 6 5 5 7 5 6 5 5 5 5 7 6 7 6 6 7 6 5
## [1925] 6 5 5 3 6 5 9 5 6 7 7 8 5 4 5 7 6 5 7 7 6 4 6 6 7 6 9 6 5 7 6 5 7 5 6 7 3
## [1962] 8 7 7 6 5 5 4 6 8 5 5 5 7 7 6 5 5 6 7 8 8 7 7 6 4 8 9 6 9 6 8 7 5 8 4 7 8
## [1999] 6 7 6 8 5 6 7 7 6 5 7 7 6 6 7 5 8 7 6 8 8 6 5 8 6 7 5 7 6 7 4 3 4 7 6 7 6
## [2036] 9 5 6 7 7 6 4 7 1 8 6 6 6 5 5 3 5 8 4 5 5 5 6 5 5 7 5 5 7 5 4 7 8 5 6 7 5
## [2073] 5 5 7 5 5 7 7 6 5 5 3 7 6 6 6 5 6 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [2110] 5 5 5 5 5 5 7 5 8 6 5 5 5 6 5 7 5 5 5 5 5 5 5 5 5 5 7 5 4 6 5 6 5 8 5 5
## [2147] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 5 6 5 6 6 5 6 7 6 5 5 5 5 5 5
## [2184] 7 3 5 6 6 8 5 7 5 5 6 7 6 7 5 5 7 7 8 7 8 8 5 6 6 7 6 6 7 5 6 6 6 7 5 8 8
## [2221] 7 6 5 4 6 6 5 5 5 3 5 7 7 6 7 5 6 7 5 2 5 4 6 6 9 5 5 6 6 5 5 5 5 6 6 6 5
## [2258] 6 8 5 5 5 5 5 7 5 5 5 5 5 5 5 5 4 5 5 5 5 5 5 5 4 4 7 5 4 5 6 5 6 3 5 5 6
## [2295] 6 7 7 7 3 6 6 7 7 5 6 5 6 5 6 6 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 7 5 5 5
## [2332] 5 5 5 5 5 5 6 5 5 5 5 7 5 7 5 6 5 5 6 5 6 5 5 6 6 5 6 6 5 5 4 4 5 5 6 5 7
## [2369] 5 7 5 5 5 5 7 5 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [2406] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [2443] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [2480] 5 5 5 5 4 5 4 5 6 6 7 5 6 6 6 6 6 7 6 4 5 5 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [2517] 5 6 5 5 5 5 5 6 6 7 6 7 5 6 5 5 5 5 7 5 6 5 7 8 5 6 5 5 7 6 6 6 5 7 7 8 6
## [2554] 7 7 7 5 8 6 6 5 3 7 6 6 5 5 3 6 5 4 7 6 6 5 5 6 6 5 6 6 5 8 5 6 6 7 5 6 6
## [2591] 5 5 5 6 6 8 7 6 3 8 6 8 4 4 4 8 5 5 5 7 5 5 6 6 7 6 5 8 6 6 4 7 5 5 4 4 4
## [2628] 5 6 4 5 5 6 6 5 8 4 6 6 8 7 7 8 8 6 5 7 6 5 8 5 7 7 5 7 8 4 8 2 7 6 4 6 7
## [2665] 7 5 9 7 5 3 5 5 7 7 7 6 4 5 7 5 6 7 6 7 6 5 5 5 8 4 5 8 4 6 6 4 8 6 5 5 5
## [2702] 4 7 3 5 5 5 8 6 6 7 6 5 7 5 6 6 4 4 5 4 4 5 9 6 6 6 4 7 5 5 5 5 5 8 7 7 8
## [2739] 5 6 8 4 5 7 5 3 5 4 5 8 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5
## [2776] 5 5 5 5 5 5 5 6 6 6 7 7 8 5 5 5 5 5 5 5 5 5 5 5 5 6 5 5 5 5 5 5 5 5 5 5 5
## [2813] 5 5 5 5 6 4 5 5 5 4 5 5 5 5 5 5 5 5 5 5 5 7 3 5 3 7 7 4 7 4 5 3 7 8 5 5 6
## [2850] 8 5 8 5 8 6 5 4 5 5 7 6 5 7 7 8 7 7 7 6 7 5 5 6 7 4 6 6 5 6 5 3 6 5 5 5 6
## [2887] 5 5 5 6 5 5 5 5 6 6 5 5 5 5 5 5 5 3 5 5 5 5 6 6 4 7 5 6 4 5 7 5 5 7 5 5 5
## [2924] 5 7 6 5 5 5 5
```

4. What is the maximum number of Full bathrooms (`Full.bath`) in this housing data set?

```r
max(ames$Full.Bath)
```

```
## [1] 4
```

5. Do any houses have more than 2 fireplaces (`Fireplaces`)? *Hint: Use the* `summary()` *function.*

```r
summary(ames$Fireplaces)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  0.0000  0.0000  1.0000  0.5993  1.0000  4.0000 
```

- There are houses that have more than 2 fireplaces.

6. What is the average sale price (`price`) for houses sold in 2010? (`Yr.Sold`). Be sure to match the data type of your logical statement with the data type of `Yr.Sold`. The year should be written as 2010, not "2010".

```r
mean(ames$price[ames$Yr.Sold==2010])
```

```
## [1] 172597.6
```