The American Community Survey distributes downloadable data about United States communities. Download the 2006 microdata survey about housing for the state of Idaho using `download.file()` from here:

`https://dl.dropbox.com/u/7710864/data/csv_hid/ss06hid.csv`

`or here:`

`https://spark-public.s3.amazonaws.com/dataanalysis/ss06hid`

and load the data into `R`.

Code Book

The code book, describing the variable names is here:

`https://dl.dropbox.com/u/7710864/data/PUMSDataDict06.pdf`

`or here:`

`https://spark-public.s3.amazonaws.com/dataanalysis/PUMSData`

How many housing units in this survey were worth more than $1,000,000?

```
# load the data into R
idahoData <- read.csv("ss06hid.csv", header=TRUE)

# Is it just Idaho data?
table(idahoData$ST)
#Check the PDF - what does 16 mean?

#any missing data?
summary(idahoData$ST)

# How many housing units are worth
# more than $1,000,000?
table(idahoData$TYPE,idahoData$VAL)
```

```
#from local files
idahoData <- read.csv("daquiz2.csv", header=TRUE)
```

Question 4

- Use the data you loaded from Question 3.
- Consider the variable FES.
- Which of the "tidy data" principles does this variable violate?

```
# let's look!
unique(idahoData$FES)
```

Options

(i) Each tidy data table contains information about only one type of observation.
(Not so)

(ii) Each variable in a tidy data set has been transformed to be interpretable. (No)

(iii) Tidy data has no missing values.

(iv) Tidy data has one variable per column.

Use the data you loaded from Question 3.

- How many households have 3 bedrooms and and 4 total rooms?
- How many households have 2 bedrooms and 5 total rooms?
- How many households have 2 bedrooms and 7 total rooms?

```
#USING TABLE
#Rooms on Rows , Bedrooms on Columns
#dnn adds dimension names

table(idahoData$RMS,idahoData$BDS,dnn=list("RMS","BDS"))
```

Another Way of Doing it

```
# How many households have 3 bedrooms and 4 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==3 &
!is.na(idahoData$BDS) & idahoData$RMS==4,])
# How many households have 2 bedrooms and 5 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==2 &
!is.na(idahoData$BDS) & idahoData$RMS==5,])
# How many households have 2 bedrooms and 7 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==2 &
!is.na(idahoData$BDS) & idahoData$RMS==7,])
```

- Use the data from Question 3.
- Create a logical vector that identifies the households on greater than 10 acres who sold more than $10,000 worth of agriculture products.
- Assign that logical vector to the variable ‘`agricultureLogical`‘.
- Apply the ‘`which()` function like this to identify the rows of the data frame where the logical vector is ‘TRUE‘.

```
# Like this (this wont run yet)
which(agricultureLogical)
```

What are the first 3 values that result?

```
# Showing off a bit
q6cols <- c("ACR", "AGS")
which(names(idahoData) %in% q6cols)

# logical vector
agricultureLogical <- idahoData$ACR==3 & idahoData$AGS==6

# and:
which(agricultureLogical)
```

Question 7

- Use the data from Question 3.
- Create a logical vector that identifies the households on greater than 10 acres who sold more than $10,000 worth of agriculture products.
- Assign that logical vector to the variable `agricultureLogical`.
- Apply the `which()` function like this to identify the rows of the data frame where the logical vector is TRUE and assign it to the variable indexes.

```
indexes =  which(agricultureLogical)
```

If your data frame for the complete data is called `dataFrame` you can create a data frame with only the above subset with the command:

```
subsetDataFrame  = dataFrame[indexes,]
```

Note that we are subsetting this way because the NA values in the variables will cause problems if you subset directly with the logical statement.

How many households in the subsetDataFrame have a missing value for the mortgage status (MRGX) variable?

```
indexes <- which(agricultureLogical)
subsetIdahoData <- idahoData[indexes,]

# And then:
nrow(subsetIdahoData[is.na(subsetIdahoData$MRGX),])
```

Question 8

- Use the data from Question 3.
- Apply ‘`strsplit()`‘ to split all the names of the data frame on the characters ”wgtp”.
- What is the value of the 123 element of the resulting list?

```
List <- strsplit(names(idahoData), "wgtp")
List[123]
```

Question 9

What are the 0% and 100% quantiles of the variable `YBL`? Is there anything wrong with these values? *Hint: you may need to use the `na.rm` parameter.*

```
quantile(idahoData$YBL, na.rm=TRUE)
#  0%  25%  50%  75% 100%
#  -1    3    5    7   25
```

Question 10

In addition to the data from Question 3, the American Community Survey also collects data about populations. Using ‘`download.file()`‘, download the population record data from:

`https://dl.dropbox.com/u/7710864/data/csv_hid/ss06pid.csv`

`or here:`

`https://spark-public.s3.amazonaws.com/dataanalysis/ss06pid`

- Load the data into R. Assign the housing data from Question 3 to a data frame ‘housingData‘ and the population data from above to a data frame ‘populationData‘.
- Use the merge command to merge these data sets based only on the common identifier "SERIALNO".
- What is the dimension of the resulting data set?

Merging Data Sets

```
housingData <- read.csv("ss06hid.csv", header=TRUE)
popuData <- read.csv("ss06pid.csv", header=TRUE)

dim(merge(housingData,
popuData, by="SERIALNO", all=TRUE))
```