Lab Worksheet 3

Question 1: *The data set* `AirPassengers` *built into R lists total numbers of international airline passengers, 1949 to 1960.*

```
AirPassengers
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949 112 118 132 129 121 135 148 148 136 119 104 118
## 1950 115 126 141 135 125 149 170 170 158 133 114 140
## 1951 145 150 178 163 172 178 199 199 184 162 146 166
## 1952 171 180 193 181 183 218 230 242 209 191 172 194
## 1953 196 196 236 235 229 243 264 272 237 211 180 201
## 1954 204 188 235 227 234 264 302 293 259 229 203 229
## 1955 242 233 267 269 270 315 364 347 312 274 237 278
## 1956 284 277 317 313 318 374 413 405 355 306 271 306
## 1957 315 301 356 348 355 422 465 467 404 347 305 336
## 1958 340 318 362 348 363 435 491 505 404 359 310 337
## 1959 360 342 406 396 420 472 548 559 463 407 362 405
## 1960 417 391 419 461 472 535 622 606 508 461 390 432
```

Is the dataset tidy? Explain why or why not.

No the each entry in the rows is not attributed to one observational unit.

Question 2: *The function* `data()` *lists all data sets that are available in R by default. Look through the list and identify a data set that is tidy. Explain why the data set is tidy.*

I pick the data set… :iris

```
trees
```

```
##    Girth Height Volume
## 1    8.3     70   10.3
## 2    8.6     65   10.3
## 3    8.8     63   10.2
## 4   10.5     72   16.4
## 5   10.7     81   18.8
## 6   10.8     83   19.7
## 7   11.0     66   15.6
## 8   11.0     75   18.2
## 9   11.1     80   22.6
## 10  11.2     75   19.9
## 11  11.3     79   24.2
## 12  11.4     76   21.0
## 13  11.4     76   21.4
## 14  11.7     69   21.3
## 15  12.0     75   19.1
## 16  12.9     74   22.2
## 17  12.9     85   33.8
## 18  13.3     86   27.4
## 19  13.7     71   25.7
## 20  13.8     64   24.9
## 21  14.0     78   34.5
## 22  14.2     80   31.7
## 23  14.5     74   36.3
## 24  16.0     72   38.3
## 25  16.3     77   42.6
## 26  17.3     81   55.4
## 27  17.5     82   55.7
## 28  17.9     80   58.3
## 29  18.0     80   51.5
## 30  18.0     80   51.0
## 31  20.6     87   77.0
```

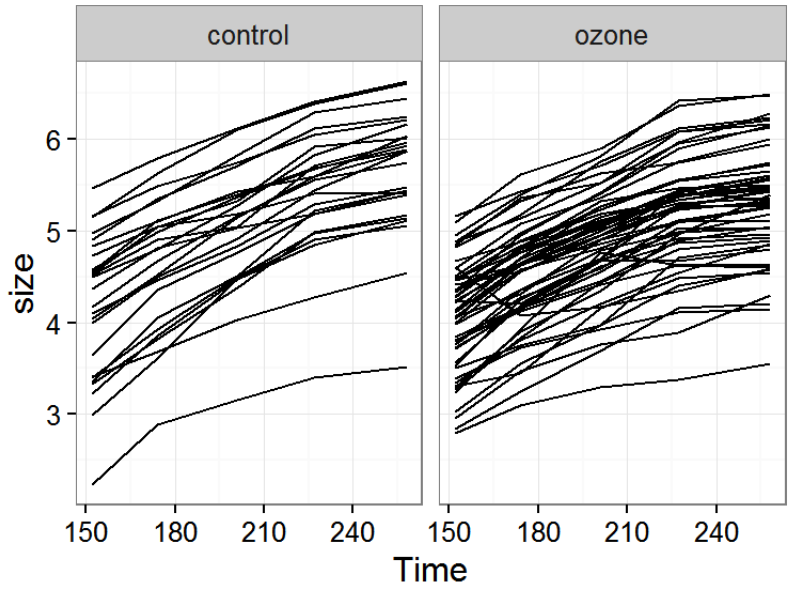
It is tidy. All variables correspond to one column each, and each row in the data set corresponds to one observational unit (tree)

Question 3: *In an in-class exercise, we made the following plot of the Sitka dataset:*

```
# download the sitka data set:
sitka <- read.csv("http://wilkelab.org/classes/SDS348/data_sets/sitka.csv")
head(sitka)
```

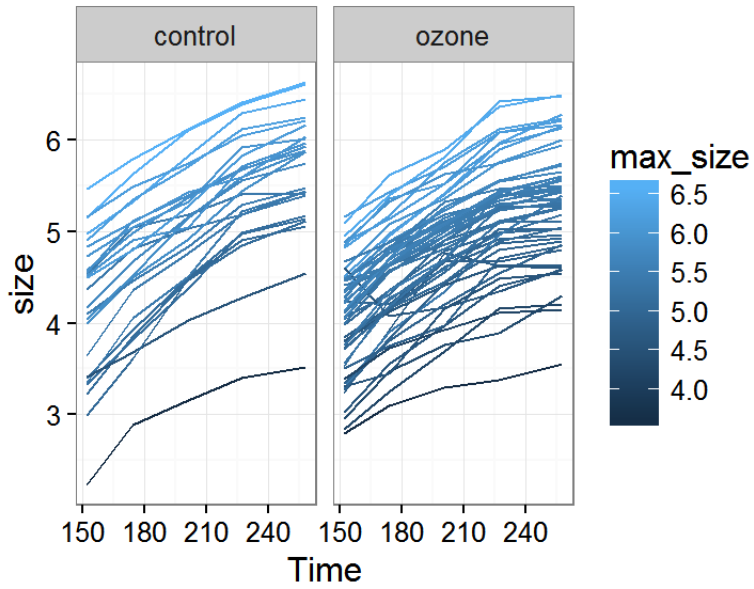
```
##   size Time tree treat
## 1 4.51  152    1 ozone
## 2 4.98  174    1 ozone
## 3 5.41  201    1 ozone
## 4 5.90  227    1 ozone
## 5 6.15  258    1 ozone
## 6 4.24  152    2 ozone
```

```
ggplot(sitka, aes(x=Time, y=size, group=tree)) + geom_line() + facet_wrap(~treat)
```



Now modify the plot so that the line for each tree is colored according to the maximum size of the tree.

```
sitka %>% group_by(tree) %>%
          mutate(max_size=max(size)) -> maxSit
ggplot(maxSit, aes(x=Time, y=size, group=tree, color = max_size)) + geom_line() + facet_wrap(~treat)
```



If that was easy…

Question 4: *The package nycflights13 contains information about all flights departing from one of the NY City airports in 2013. In particular, the data table* `flights` *lists on-time departure and arrival information for 336,776 individual flights:*

```
library(nycflights13)
```

```
## Warning: package 'nycflights13' was built under R version 3.1.3
```

```
flights
```

```
## Source: local data frame [336,776 x 16]
##
##     year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
##    (int) (int) (int)    (int)     (dbl)    (int)     (dbl)   (chr)   (chr)
## 1   2013     1     1      517         2      830        11      UA  N14228
## 2   2013     1     1      533         4      850        20      UA  N24211
## 3   2013     1     1      542         2      923        33      AA  N619AA
## 4   2013     1     1      544        -1     1004       -18      B6  N804JB
## 5   2013     1     1      554        -6      812       -25      DL  N668DN
## 6   2013     1     1      554        -4      740        12      UA  N39463
## 7   2013     1     1      555        -5      913        19      B6  N516JB
## 8   2013     1     1      557        -3      709       -14      EV  N829AS
## 9   2013     1     1      557        -3      838        -8      B6  N593JB
## 10  2013     1     1      558        -2      753         8      AA  N3ALAA
## ..   ...   ...   ...      ...       ...      ...       ...     ...     ...
## Variables not shown: flight (int), origin (chr), dest (chr), air_time
##   (dbl), distance (dbl), hour (dbl), minute (dbl)
```

We would like to collect some information about arrival delays of United Airlines (UA) flights. Do the following: pick all UA departures with non-zero arrival delay and calculate the mean arrival delay for each of the corresponding flight numbers. Which flight had the longest mean arrival delay and how long was that delay?

```
flights %>% filter(carrier == "UA" & arr_delay != 0) %>% group_by(flight) %>% summarize(mean_delay =
mean(arr_delay)) %>% arrange(desc(mean_delay)) %>% slice(1:1)
```

```
## Source: local data frame [1 x 2]
##
##   flight mean_delay
##    (int)      (dbl)
## 1   1510        283
```

Flight 1510 had the longest mean arrival delay at 283.