```
#######################################################
# History for lecture 2 -- Introduction to ggplot and R
#
# C. Andrews
# 2014-02-13
########################################################



# basic assignment statement

x <- 5

# Creating a vector and playing with it
v <- c(1,2,3,4)
v * 4
v*2 + 5
v + c(2,2,7)
v + c(2,1)
v <- c(4,9,4,67,23,6,2,5,43)

# basic statistical functions
max(v)
min(v)
mean(v)
median(v)
sum(v)


# looking at data frames
library(help=datasets)
mtcars
names(mtcars)
str(mtcars)
mtcars$cyl



# load the census data from the website
census <- read.csv("http://www.cs.middlebury.edu/˜candrews/classes/infovis/data/census.cs
v")
View(census)
str(census)
mean(census$income)
summary(census)

# install ggplot2 (only need to do this once)
install.packages("ggplot2")

# make the functions and objects in ggplot2 available for use
library(ggplot2)

# recrete the higher degree x income scatterplot
ggplot(census, aes(x=perCollege, y=income)) + geom_point()
ggplot(census, aes(x=perCollege, y=income, size=population)) + geom_point()
ggplot(census, aes(x=perCollege, y=income, size=population)) + geom_point(shape=21, color
="black", fill="white")

# load in the Doctor Who data set
drwho <- read.csv("http://www.cs.middlebury.edu/˜candrews/classes/infovis/data/drwho.csv"
)

# Create a bar chart of the amount of actual screen time each doctor had
# the stat attribute tells geom_bar not to do any kind of grouping or statistical
# evaluation on the duration (identity is essentially multiply by 1)
ggplot(drwho, aes(x=doctor, y=duration)) + geom_bar(stat="identity")

# in response to request, we are adding data about the Doctor’s companions
# this builds the points up in size and color to make them more visible
ggplot(drwho, aes(x=doctor)) + geom_bar(stat="identity", aes(y=duration)) + geom_point(ae
s(y=companions))
ggplot(drwho, aes(x=doctor)) + geom_bar(stat="identity", aes(y=duration)) + geom_point(ae
s(y=companions), color=red, size=5)
ggplot(drwho, aes(x=doctor)) + geom_bar(stat="identity", aes(y=duration)) + geom_point(ae
s(y=companions), color="red", size=5)
ggplot(drwho, aes(x=doctor)) + geom_bar(stat="identity", aes(y=duration)) + geom_point(ae
s(y=companions*100), color="red", size=5)


# Now we are taking a look at the Orange dataset (tracks the growth of five trees)
View(Orange)
str(Orange)

# putting this in a bar chart makes a stacked bar chart
ggplot(Orange, aes(x=age, y=circumference, fill=Tree)) + geom_bar(stat="identity")
# adjust the position to "dodge" to get side by side measures for each sample
ggplot(Orange, aes(x=age, y=circumference, fill=Tree)) + geom_bar(stat="identity", positi
on="dodge")

# taking a look at the mtcars dataset
str(mtcars)
# create a histogram of the number of cars with each cylinder configuration
ggplot(mtcars, aes(x=cyl))+geom_bar()

# making cyl a factor makes the graph a little nicer
ggplot(mtcars, aes(x=factor(cyl)))+geom_bar()

# back to the Doctor Who data set, we are creating a histogram of which years
# the doctors started. This is not interesting when we look at individual years
ggplot(drwho, aes(x=start))+geom_bar()

# change the bin width to be 10 year blocks and we can see the # of Doctors
# per decade
ggplot(drwho, aes(x=start))+geom_bar(binwidth=10)

# take a look at the movies data set from ggplot2
?movies
# create a histogram of how many movies were released each year
ggplot(movies, aes(x=year))+geom_bar()

# Another common graph type is the line graph, here we chart the growth of
# our orange trees
# first, we use color to make each tree use a different color
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line()

# or we could change the line type (this is attribute I couldn’t recall in class
ggplot(Orange, aes(x=age, y=circumference, linetype=Tree)) + geom_line()

# we can also combine layers to put points on the lines
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()

# adding geom_text allows us to add labels
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference))

# per request, here is what happens if you put quotes around the column name
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label="circumference"))

# add a y aesthetic to the labels to move them up from the points
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference, y=circumference+10))

# make the text black -- note we are setting the color, not mapping it, so it
# is not in the aes() function
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference, y=circumference+10), color="black")

# we can use scales to determine how colors are chosen
# here we move to a grey scale
```

```
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference, y=circumference+10), color="black") + scale_color_grey()

# this makes use of the blues scale from color brewer
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference, y=circumference+10), color="black") + scale_color_brewer()

# list all of the available color brewer scales
library(RColorBrewer)
display.brewer.all()

# create a manual set of colors for our Trees
ggplot(Orange, aes(x=age, y=circumference, color=Tree)) + geom_line() + geom_point()+geom
_text(aes(label=circumference, y=circumference+10), color="black") + scale_color_manual(v
alues=c("red","blue","green","purple","darkred"))

# subset() allows us to conditionally pick rows out of our data frame
# this is picking all of the movies from the 90s
movies90 <- subset(movies, year>=1990 & year <2000)
View(movies90)

# grab the baby name data set
names1880.2012 <- read.csv("http://www.cs.middlebury.edu/~candrews/classes/infovis/data/n
ames1880-2012.csv")
View(names1880.2012)

# create a subset that just includes two names
smNames <- subset(names1880.2012, (Name=="Andrew" | Name=="Christopher" )& Gender=="M")
View(smNames)

# geom_area creates a stacked layer graph
# [the problems we had in class was due to the ordering in the data -- using
# order in geom_area forces the stacks to be ordered correctly]
ggplot(smNames, aes(x=Year, y=Count, group=Name)) + geom_area(aes(fill=Name, order=Name))
```