STAT460 – Homework 1
Due: Jan. 28 at the start of class.

1. Introduction to R

   (a) Using this code, generate a standard normal vector of length 100[1].

   ```
   set.seed(3)
   mu = ?
   sd  = ?
   y=rnorm(100,mu,sd)
   ```

   That is, $y \sim N(\text{mu}, \text{sd} * I_{100})$, where I_{100} is a matrix with 100 1's along the diagonal and 0's everywhere else. This encodes the distribution as having mean mu $\in \mathbb{R}^{100}$, each component having variance sd^2 (corresponding to the 1's), and each element in the vector is uncorrelated (the 0's).

   Find the sample mean, sample variance, and and standard deviation of y when

      i. mu = 0 and sd = 1

      ii. mu = 10 and sd = 15

   (b) Plotting

   ```
   x=rnorm(100)
   y=rnorm(100)
   pdf('name and location of file you want to create')
   plot(x,y,xlab="this is the x-axis",ylab="this is the y-axis",main="Plot of X vs Y")
   dev.off()
   ```

   Run this code to produce a plot of x versus y. Include this plot as your answer to 1b.

   (c) Indexing data

   ```
   A = matrix(1:20,nrow=4,ncol=5)
   ```

   What is the entry in the second row and third column of A? What is the R code needed to find this out?

   (d) Installing packages.

   Over the course of the semester we will need to install packages. Install the 'leaps' package, load it into your R session, and type 'regsubsets' then press enter. For your answer to this question, tell me what happened after typing 'regsubsets'. Note: if it says 'Error: object 'regsubsets' not found' there is something wrong. Here is the code for installing and loading a package.

   ```
   install.packages('leaps')
   library(leaps)
   ```

   ```
   > regsubsets
   function (x, ...)
   {
       UseMethod("regsubsets", x)
   }
   <environment: namespace:leaps>
   ```

[1] Note that this code will not run as written. In this problem, you will fill in values for mu and sd.

Enter the following matrix into `R`

(e) `A = matrix(1:12,ncol=4,nrow=3)`

Suppose we want to get the column mean for each column of the matrix A. Do this with

i. Hard coding

ii. For loop(s)

iii. The `apply` (or related) function