**Peer-graded Assignment: Programming Assignment 2: Lexical Scoping**

#This function creates a special "matrix" object that can cache its inverse

makeCacheMatrix <- function(x = matrix()) {

Inv <- NULL

set <- function(y) {

x <<- y

Inv <<- NULL

}

get <- function() x

setinverse <- function(inverse) Inv <<- inverse

getinverse <- function() Inv

list(set = set, get = get,

setinverse = setinverse,

getinverse = getinverse)

}

# This function computes the inverse of the special "matrix" returned by makeCacheMatrix

cacheSolve <- function(x, ...) {

Inv <- x$getinverse()

if(!is.null(Inv)) {

message("getting cached data")

return(Inv)

}

data <- x$get()

Inv <- inverse(data, ...)

x$setinverse(Inv)

Inv

}