cachematrix.R

peter

Tue Jul 21 15:37:13 2015

## Create a matrix containing a list of functions that set the value of the   
## matrix, get the value of the matrix, set the value of the inverse matrix,  
## and get the value of the inverse matrix.  
  
makeCacheMatrix <- function(x = matrix()) {  
   
 m <<- NULL  
  
## specify the internal functions related to the matrix.   
   
 set <- function(y) {  
 x <<- y  
 m <<- NULL  
 }  
 get <- function() x  
   
 ## specify the internal functions related to the inverse.  
   
 setmatrix <- function (solve) m <<- solve  
 getmatrix <- function () m  
   
 ## create a name list so internal functions can be retrieved by name externally.  
   
 list(set=set, get=get, setmatrix=setmatrix, getmatrix=getmatrix)  
}  
  
  
## Function to determine if inverse is already in cache  
## if it is, message "getting cached data" and return cached matrix  
  
cacheSolve <- function(x, ...) {  
 m <- x$getmatrix()  
 if(!is.null(m)) {  
 message("getting cached data")  
 return(m)  
 }  
   
 ## if the inverse is not already in the cache, retrieve the matrix,  
 ## then apply the solve function,  
 ## storing the value in m (and the cache).  
   
 matrix <- x$get()  
 m <- solve(matrix, ...)  
 x$setmatrix(m)  
 m   
}