Programming Assignment 2: Lexical Scoping

Ahmed Ezzat Afifi

23-10-2014

The document to solve the Programming Assignment 2: Lexical Scoping found here (https://class.coursera.org/rprog-008/human_grading/view/courses/972581/assessments/3/submissions)

I found the hints the course staff wrote in the discussion fourms to be very helpful for further understanding and actually says everything I was about to write down. might want to check it out as they actually did the job for me this time :) the link here (https://class.coursera.org/rprog-008/forum/thread?thread_id=174). And as always ask me if anything is not clear.

After that things are pretty easy now! our task is just to replicate the original makeVector and cachemean to work with Matrices instead of vectors

Not trying to confuse you but think of those parent functions as a terrible version of Classes in other languages. for methods/functions inside JS objects if you are familiar with Javascript.

That said, they are “technically” not classes at all, but the behaviour is extremely similar when accessing methods with $ instead of dot “.” operator Here is the original functions.

if the past lines didn’t make sense, just ignore them. you will understand this better when you learn Object Oriented programming in Python

Now the original code, we will be using as a guide for our functions

```r
makeVector <- function(x = numeric()) {
        m <- NULL
        set <- function(y) {
                x <<- y
                m <<- NULL
        }
        get <- function() x
        setmean <- function(mean) m <<- mean
        getmean <- function() m
        list(set = set, get = get,
             setmean = setmean,
             getmean = getmean)
}


cachemean <- function(x, ...) {
        m <- x$getmean()
        if(!is.null(m)) {
                message("getting cached data")
                return(m)
        }
        data <- x$get()
        m <- mean(data, ...)
        x$setmean(m)
        m
}
```

so? here (https://github.com/rdpeng/ProgrammingAssignment2/blob/master/cachematrix.R) they are giving is functions prototype.

Now our code will be looking like the prototype. so lets write it down

```r
makeCacheMatrix <- function(x = matrix()) {
 inv <- NULL #Now we want to calculate the inverse of the matrix not m "mean" of vector

 #Now we want to set the Matrix as the original vector function

 set <- function(y){
         x <<- y # same as original
         inv <- NULL

 }

 #Now get function to handle the matrix
 #Thanks to dynamic binding in R
 #We don't need to change anything since x is already defined as a matrix
 #In the function header

 #You can use curly braces around x; it does not matter since its only one line in the fu
nction.
 #I prefer them for clarity, but will leave it like that to be the same as the original
 get <- function() x

 setInv <- function(matinv) inv <<- matinv

 ## Get cached matrix
 getCache <- function() inv

 ## Return cachedmatrix type
 list(set = set, get = get, setInv = setInv, getCache = getCache)

}
```

Now for the same story goes with the cacheSolve. Same as the original function, but optimized to work with our makeCacheMatrix variables instead

```r
cacheSolve <- function(x, ...) {
  inv <- x$getCache()

  ## Check if matrix has already been cached
  if(!is.null(inv)) {
          message("getting cached inverse matrix data.")
          return(inv)
          }
  ## Get matrix
  data <- x$get()

  # Now in Original matrix they used mean function
  #This is where we want to get the inverse of our matrix
  #The R function for that is solve()
  #Check ?solve
  inv <- solve(data, ...)

  # now we set the inverse in our cache method
  x$setInv(inv)

  # And as always, the last line of a function is the function return

  inv

  }
```

Now what you should do is to ask understand the code :), write it down yourself and try it out as explained by the course staff fourm post.

and fork this repository (https://github.com/rdpeng/ProgrammingAssignment2/blob/master/cachematrix.R) into your github and edit the functions with the code you wrote.

and submit your code link here