Forums / Programming Assignment 2 (peer assessment): Lexical Scoping Help

Can we have a demonstration of the mean example?

Subscribe for email updates.

No tags yet. + Add Tag

Sort replies by: Oldest first Newest first Most popular

Joseph Levy · 16 days ago

It is unclear to me how the two functions in the example work together.

↑ 33 ↓ · flag

Gregory D. Horne COMMUNITY TA · 16 days ago

Copy-and-paste the example functions into a file and test their behaviour.

1. create a numeric vector
2. compute the mean
3. output the vector
4. output the mean
5. compute the mean again (observe its behaviour)
6. change the existing vector
7. compute the mean again (observe its behaviour)

[Update: (2014-07-20) Abusing the CTAs only drives these volunteers away from participating in future sessions. I suggest the down-voting brigade grow-up PDQ before sanctions are imposed. Under no circumstances will a CTA provide solutions to the assignments. We will provide pointers and guidance, and might choose to provide the occasional reference code sample to illustrate a concept when words along are not sufficient to adequately convey the information. We are volunteers, not paid staff, and serve the learning community willingly; most of us have other professional & personal responsibilities and obligations as well. Do NOT abuse the privilege you have been given by Coursera and the many universities participating in this experiment in educational democracy.]

↑ -33 ↓ · flag

A post was deleted

Ryan T Johnson Signature Track · 12 days ago

Here let me follow your instructions:

```
> source('~/R_class/specdata/makeVector.R')
> source('~/R_class/specdata/cachemean.R')
t <- numeric(5)
> mean(t)
[1] 0
```

Then you get to "output the vector" which if I'm following instructions really don't know what you mean.
I think you guys really need to provide an example or some more explanation as to what these functions are doing.

↑ 24 ↓ · flag

Cvetanka Jordanoska Signature Track · 4 days ago 🔗

I did the same and got the following error:

```
Error in x$getmean : $ operator is invalid for atomic vectors
```

So, what I'm missing...

↑ 4 ↓ · flag

Scott Francis · 4 days ago 🔗

i believe the function name is 'get' not 'getmean' .. AND it i*IS* a function, so be sure to call it with '()'

> x$get()

↑ 1 ↓ · flag

Cvetanka Jordanoska Signature Track · 4 days ago 🔗

It's already solved! Thanks for the response Scott.

↑ 0 ↓ · flag

Anonymous · a day ago 🔗

I am also getting same error how it is solved??

↑ 0 ↓ · flag

+ Comment

Joseph Levy · 15 days ago

Thank you!

0 · flag

Ryan T Johnson Signature Track · 12 days ago

So if that TA post was helpful to you would you care to explain it? I can create vectors and calculate means but I don't have any idea how I'm supposed to be using makeVector or cachemean.

4 · flag

Gregory D. Horne COMMUNITY TA · 12 days ago

Joseph, I was providing a sequence of steps to run against the sample functions makeVector() and cachemean(). Sorry if it was not clear in my original response.

-19 · flag

Adam Gruer Signature Track · 12 days ago

Ryan, check my example below that shows how I believe the functions should be used to more or less meet Gregory's list of steps. It certainly took me a long time and a lot of searching to tease this out, but once I ran each of those lines of code it all started to make much more sense.

6 · flag

Ryan T Johnson Signature Track · 12 days ago

Adam, I should have mentioned, your example below is the only way I understood this exercise. Thank you. But that's part of what irks me. Joseph asked a question, got a unclear answer from course TAs and then you had to come in and explain it for us all. From my previous experience in Matlab and Python, including an example call to the function is standard practice somewhere within the function body usually. Instead, you are doing that for us.

These assignments seem poorly explained and provide you with less information than most other functions would. I understand the need to develop research skills but at least provide a single example when you introduce a new function!

48 · flag

A post was deleted

Thomas Gludovacz · 9 days ago

This course has inadvertently opened my eyes to the value of example call functions when writing python.

↑ 6 ↓ · flag

Ryan T Johnson Signature Track · 9 days ago 🔗

Haha! Exactly! An example call is just good coding practice. You'd think they'd want to demonstrate that.

↑ 2 ↓ · flag

Andre Susantin Signature Track · 8 days ago 🔗

Higher order function in some other language

↑ 0 ↓ · flag

🗑A post was deleted

Karen Williams Signature Track · 5 days ago 🔗

If you take a look at what we're being graded on here, actually writing the functions correctly is not the important part. What we're being graded on is one of the basic, most important parts of doing statistics with R: borrowing, modifying, and sharing back existing solutions, with our own contributions clearly understandable.

We're being graded by our peers (us), and we're being told not to actually run the code to decide if it works. We're just supposed to see if it looks plausible and is well documented. That's what we should do. Don't stress this one too much.

↑ 2 ↓ · flag

Anonymous · a day ago 🔗

I entirely disagree. If we are given an example, it needs to have purpose...it should teach us something worthwhile and useful. The amount of time spent on trying to figure out how to even call the function in this week's assignment could have been better spent learning to actually do some statistical tests in R...or even washing my laundry.

↑ 2 ↓ · flag

Vishal Signature Track · a day ago 🔗

Actually, if you stress this assignment , you will learn about "superassignment" <<- operator and clojures. I am not an ace programmer, I am just a beginner but I did spend a lot of hours understanding "<<-" and when I understood , it was like some light bulb was switched on in my brain. However, hard it looks, if you spend some time, you will learn something. It is tough but I didn't expect it to be easy. It was a real learning curve(so far ..do not know about third assignment yet) . I think it will teach us "how to search" the solution space, how to present our problem on forums in detailed manner so others can replicate . In all, I think course will make us more of a rational/logical thinker. I like to think of it as puzzles but it does suck oxygen out of your brain at times then you see the whilte blinding light and come back to life :) Good luck with your efforts

↑ 2 ↓ · flag

+ Comment

Adam Gruer Signature Track · 14 days ago

Here is a sample of my testing of the demo functions. I think it is in line with what Gregory has listed?

```
#Call  the makeVector() function and assign it's
#  return value ( a list of four functions) to a variable, v
#  v is now a list of four functions
v <- makeVector()

#use v's set function to create a vector
#  containing the numbers 20 through to 40
v$set(20:40)

#use v's get function to retrieve the vector created
v$get()

#pass the list v to the cachemean() function
#   the mean of the numeric vector 20:40 should be returned
cachemean(v)

#pass the list v to the cachemean() function a second time
#  the mean of the numeric vector 20:40 should be returned
#  also a message "retrieving value from cache" indicating that the mean
#  is not being calculated this time but is being retrieved from the cached
#  value
cachemean(v)

#use v's set function to create a new vector
#  containing the numbers 23,23,34.6,654.35
v$set(c(23,23,34.6,654.35))

#pass the list v to the cachemean() function
#   the mean of the numeric vector 23,23,34.6,654.35 should be returned
cachemean(v)

#pass the list v to the cachemean() function a second time
#  the mean of the numeric vector 23,23,34.6,654.35 should be returned
#  also a message "retrieving value from cache" indicating that the mean
#  is not being calculated this time but is being retrieved from the cached
#  value
cachemean(v)
```

↑ 206 ↓ · flag

Ryan T Johnson Signature Track · 12 days ago

Thank you Adam for a clear, logical walk through a custom function that is being thrown at us with little explanation.

↑ 10 ↓ · flag

Mario Ibanez Signature Track · 11 days ago

Thanks Adam!

↑ 1 ↓ · flag

sriramu MS · 11 days ago

Thank you Adam. Much appreciated..

↑ 0 ↓ · flag

GIUSEPPE VERARDI · 11 days ago

Kudos to you Adam for this excellent explanation. Without it, I would have never get rid of this exercise

↑ 2 ↓ · flag

Sinkari Kassim · 11 days ago

concise explanation

↑ 1 ↓ · flag

Philippe Jette Signature Track · 11 days ago

Well done Adam! That being said, the work done here essentially constitutes the entire assignment. Once equipped with an understanding of the functions, the modifications required to invert a matrix are minimal. I think the point was to get us to understand the code.

↑ -29 ↓ · flag

Kaushik Lakshman · 10 days ago

Thank you very much mate. The method of writing these functions was a bit different to what we've seen as beginners to R. Thanks again for telling us how to interpret it.

↑ 2 ↓ · flag

Jose Delcour · 10 days ago

Thanks, Adam. I am still struggling with this course and your explanations are very clear and make sense to me. Why is this course so convoluted when it is possible to explain thing much clearer and through examples ?

↑ 9 ↓ · flag

Matthew Reader Signature Track · 10 days ago

Thanks Adam. I too had a hard time understanding the instructions and what the example function was actually doing. Your example really clarified what was going on for me.

To me, this example in R seems very similar to object oriented programming and classes in Python, where you can create a class, assign different attributes to that class, and get information back out of an object using methods defined within the class.

↑ 3 ↓ · flag

Jack Hummeland Jacobsen · 9 days ago

Thanks a LOT Adam.

The structure of the code, the passing of variables and how some of the functions were used had me utterly bewildered. How about some comments explaining the basic flow of the program for us newbies??

This assignment and sample code looked hopeless to me even though I have followed the videos and answered the quizzes so far. There seems to be a mismatch between what we were taught and what we were expected to code for ourselves IMO.

That said, I DO appreciate this course being available..

↑ 16 ↓ · flag

Amit Nagar Signature Track · 9 days ago

Wonderful explanation, Adam. Crystal clear. Thank You.

↑ 0 ↓ · flag

VALAT Didier Signature Track · 8 days ago

Thank you Adam for this clear example.

↑ 0 ↓ · flag

Valeriya Gagarina · 7 days ago

Thank you so much Adam! Great example and explanation!!!

↑ 0 ↓ · flag

Jose D. Mercado Signature Track · 6 days ago

Thanks Adam, you saved us a lot of time and frustration!

↑ 0 ↓ · flag

A post was deleted

Giacomo Bianchi Signature Track · 3 days ago

Praise to Adam,
that really helped me as a beginner to understand the complexity of this functions and the mental process within. You've been more effective than our instructor's fast walkthrough on the topic...

↑ 4 ↓ · flag

Le Thi Thu Hang Signature Track · 3 days ago

Thank you a lot , Adam :).
I wonder one thing in the Adam's post :
In the step
#pass the list v to the cachemean() function
# the mean of the numeric vector 23,23,34.6,654.35 should be returned cachemean(v).

I actually don't get why the cachemean function will process m as null value since we still have the old vector as 20:40.

Can some one help me?

↑ 0 ↓ · flag

Anonymous · 2 days ago

Because calling v$set(c(...)) resets m to NULL

↑ 2 ↓ · flag

Don Voltz · 2 days ago

What is the purpose of the list at the end of makeVector()

I can not understand why it is there.

Thanks Don

↑ 0 ↓ · flag

Senthil Kumar Vadivelu Signature Track · a day ago

Thanks a lot, Adam. Things are clear now :)

↑ 0 ↓ · flag

Cara Signature Track · a day ago

I'm somehow missing something so basic that no one else seems to have missed. I dont remember ever seeing the $ symbol in any lectures, ever. "v$get" is going to return whats in the named spot, get, in the list V? which is then exactly the same as calling the get function stored in the named spot?

But in the cachmean example, what does x$getmean do exactly? x is a parameter passed to chachemean, right? so why do we say x$getmean? getmean does not take any parameters. X is either empty or contains the vector which was passed to set. I dont understand what the $ is doing here.

↑ 0 ↓ · flag

Dmitry Goldenberg Signature Track · a day ago

I think the example up above was v <- makeVector(), so you'd do

z <- v$getmean()

to get the mean value of the new vector. I think $ is basically a way in R to reference attributes or methods on an object. This wasn't too clear from the lectures but there was a mention of how to access e.g. a given column in a data frame, such as mydf$col1 gives you the column named col1.

↑ 0 ↓ · flag

Cara Signature Track · a day ago

so back to the example in the assignment - the x inside the two functions are completely different. the x passed to cachemean is actually the list created in makevector. I honestly think the X's were confusing me because . ..

well, because of thinking that x was global. But its not global, its in the parent space - but in the prof's makevector, when there is:

```
    set <- function(y) {
    x <<- y
```

is x set in the environment of makevector (which should be the parent of set) or in the parent workspace above makevector?

I am starting to get this a little better and I appreciate the patience - I dont think I've worked with anything much like this, passing functions seems to confuse me though I know its common practice.

↑ 0 ↓ · flag

+ Comment

Anonymous · 11 days ago

In a normal OO world with variables and methods in an object , methods won't get replicated when you create multiple objects from the same class(prototype), only data will. Here in R, functions are stored in lists. How are functions handled here.?. Do they get replicated?.If not how they are handled differently from other objects of lists.

↑ 0 ↓ · flag

David McDuffee · 11 days ago

Creating the "object" is copying the list. Of course the functions are replicated; they're members of the list.

↑ 1 ↓ · flag

+ Comment

Scott Francis · 11 days ago

Hi all,

I got a bit confused in this assignment as well without a use case. Thanks Gregory for outlining the usage and to Adam for codifying for the vector example. Other Cousera courses I've taken often post test case code -- either as part of the assignment or on forums. Here's a short test script I worked up for this assignment. I don't believe this violates the letter or intent of any policy as it's just a way to communicate more clearly about expectations.

Hopefully, Gregory, et.al. can look this over to make sure I've understood the assignment...

```
# create cacheable matrix object

m<- makeCacheMatrix( )



# initailize with a an easy to inspect matrix

m$set( matrix( c(0, 2, 2, 0 ), 2, 2))



# note use of parens to retrive the matrix part of the object

m$get()

#     [,1] [,2]

#[1,]    0    2

#[2,]    2    0



# test the inverse cacher

cacheSolve( m )

#       [,1] [,2]

# [1,]  0.0  0.5

# [2,]  0.5  0.0



# and again... should be cached now
```

```

cacheSolve( m )

#getting cached data <-- NOTE THE MESSAGE

# ...



# test that the inverse works and experiment with how to use the functions

# m$get() returns the matrix and cacheSolve(m) returns the inverse that we can

# use like regular matrices to do things like multilplications...

#

# product of matrix mult should be identity matrix AND we should get the cached message

m$get() %*% cacheSolve(m)

#getting cached data  <-- Yup... cached!

#     [,1] [,2]

#[1,]    1    0

#[2,]    0    1       <-- eye() think it's and identity Matrix...



# let R test identify for us

all.equal( diag(2), m$get() %*% cacheSolve(m) )

# getting cached data <-- hey.. it's still cached

#[1] TRUE             <-- R agrees it's an identity



# save the inverse off and let's see if we can break it...

m1<- cacheSolve(m)

```

```


# set m to some new values

m$set( matrix( rnorm(4), 2, 2) )



# does fetching the inverse without any pre-caching work?

all.equal( diag(2), m$get() %*% cacheSolve(m) )

#[1] TRUE             <-- yup, sure does...



# what about testing m x the inverse we squirreled away in m1?

all.equal( diag(2), m$get() %*% m1 )

#[1] "Mean relative difference: xxxx"  <-- Nah... that's a train wreck.



# try a bigger matrix and see if we can notice the caching effects

m$set( matrix( rnorm( 1000000 ), 1000, 1000 ) )

cacheSolve(m)

cacheSolve(m)

# on my quad i5, seems like the second call is faster...



# and double check it all still works...

all.equal( diag( 1000 ), m$get() %*% cacheSolve(m) )

# getting cached data

# [1] TRUE

```

↑ 40 ↓ · flag

Rajeswaran Viswanathan · 5 days ago

Scott, this is very helpful. The only additional item I would like to add is for you to check if the list is returned properly by printing the output of m after the assignment. Overall the changes required was only about 5 lines of code after we copy and modify the sample code given.

↑ 0 ↓ · flag

Edward Hopkins Signature Track · 2 days ago

Thank you Scott, I used your testing script to finish my assignment. I would also recommend changing the last two lines of the below section of your script to assess how much faster retrieving the cached inverse versus calculating the inverse can be, by using "system.time". Just learned about this in the week 4 lectures.

```
# try a bigger matrix and see if we can notice the caching effects

m$set( matrix( rnorm( 1000000 ), 1000, 1000 ) )

system.time(cacheSolve(m))

system.time(cacheSolve(m))
```

↑ 5 ↓ · flag

Aravind Arunachalam Signature Track · 6 hours ago

I think the system.time is a wonderful demonstration of the caching effects .

↑ 0 ↓ · flag

+ Comment

Rebeca Palma Polo Signature Track · 7 days ago

Awesome batteries of tests, Adam and Scott, now I am 100% sure I have understood everything, thank you SO MUCH!!

↑ 0 ↓ · flag

+ Comment

Andre Susantin Signature Track · 7 days ago

Scott, Adam,
Great job,

I wonder why the test unit is not being included in the assignment description.
To solve this assignment, the example literally, show the answer to this assignment.

This assignment teaches a closure and, higher order function, which is non-trivial to grasp for people coming from declarative programming background.
Only recently, I dived into other language with higher order, closure, such as functional programming (scala) and javascript.
Just my comments.

↑ 3 ↓ · flag

+ Comment

shuangyuan wei Signature Track · 5 days ago 🔗

Thank you so much, Adam and Scott, very helpful! I had a hard time to figure out where to start. Much appreciated!

↑ 0 ↓ · flag

+ Comment

Anonymous · 4 days ago 🔗

Thanks to Scott and Adam for their help. I'm relief to see that it's not only me having problem in understanding the assignment.
Having being a teacher myself, I think in a good assignment one should never introduce new concepts (like the <<- operator) and instructions should be very clear about its goal and steps.
Most of the people here are not used to programming, and may already struggle in having to become familiar with the language.
Those who already know some other language (I know some Python as an example) may be used to a truly different style of coding, and find hard to get what's going on here.

↑ 5 ↓ · flag

Edoardo Aroni Signature Track · 4 days ago 🔗

Since it has been mention the <<- operator.   Please see below its definition from the "R Language Definition" if you need it.
The operators `<<-` *and* `->>` *are normally only used in functions, and cause a search to made through parent environments for an existing definition of the variable being assigned. If such a variable is found (and its binding is not locked) then its value is redefined, otherwise assignment takes place in the global environment.*

↑ 0 ↓ · flag

Dave Holtzhouser Signature Track · 2 days ago 🔗

"..cause a search to made through parent environments for an existing definition of the variable being assigned."

What happens in the event that the same variable exists at multiple levels? Does it stop at the first one it finds as it works its way up the parent environments, or does it continue?

↑ 0 ↓ · flag

+ Comment

Carlos Tolvett Sepúlveda Signature Track · 4 days ago 🔗

Thank you guys.

↑ 0 ↓ · flag

+ Comment

Tavpritesh Sethi · 4 days ago 🔗

But for the excellent discussion here, this function was so cryptic. Thanks a lot Scott and Adam.

↑ 0 ↓ · flag

+ Comment

Tavpritesh Sethi · 4 days ago 🔗

Can anyone give a real-life use case for the need to bundle four functions together into one? Does is have to do with providing a common environment (the parameter passed to makeVector) to all of those four? I am really trying to make sense of how this is practically useful.

↑ 1 ↓ · flag

Tavpritesh Sethi · 4 days ago 🔗

(Apart from the caching example, that is already demonstrated)

↑ 0 ↓ · flag

+ Comment

Aritz Bercher Signature Track · 3 days ago 🔗

I'm still a little bit confused with this cache memory. Adam's example was very helpful, but I would like to know, in this "hierarchy" of environment, where are the stored the variable stored in the cache memory with the operator '<<-'. I replaced the '<<-' by '<-' in the given example, and when I follow Adam's example, and enter the three first commands:

```
v <- makeVector()
v$set(20:40)

v$get()
```

I get:

```
numeric(0)
```

The way I understand this result is the following (sorry if it's not crystal clear): When I call the function v$get(), R run v$get() and comes to the variable 'x' which has been defined in the environment of 'makevector' and had no value in this environment, so returns 'numeric(0)'.

But then, what happens when we use '<<-' ?

Does anyone have a clear explanation of the scoping rules involving the cache memory?

↑ 0 ↓ · flag

+ Comment

Ana Carolina Conchon Costa · 3 days ago

Hi, I'm new at R, and I don't quite understand what I have to do is this assignment. I forked the GitHub repository and cloned the forked GitHub repository. But I don't know how should I edit the R file. Can anyone explain to me?

↑ 0 ↓ · flag

+ Comment

Vishal Signature Track · 3 days ago

Hi Ana,

1. Start git bash , check where have you cloned the file. it will be most likely in your home directory with a name like "Programmingassignment2"
2. Once you know the location, open R studio and open that R file and edit it and save it NOTE:
3. After cloning, make sure to run "git init" and "git add -A" from the same directory in that order so that you R file is tracked. 2.When you save that R file in studio, it will be tracked if you have done

Above step

4. Once you are satisfied with the changes, commit it to local Git repository using "git commit -m message". Message could be like "matrix inversion assignment"
5. Now , your file is in "local" repo , push it to github using "git push" command.

Note, your local repo should point to web github repobefore pushing the file

You can check if that is the case using command "git remote -v"

Hope that helps,

Good luck.

Vishal

↑ 1 ↓ · flag

Ana Carolina Conchon Costa · 3 days ago 🔗

Thanks Vishal, it really helped!

↑ 0 ↓ · flag

Cvetanka Jordanoska Signature Track · 3 days ago 🔗

After cloning the repo can we just press the edit button, edit the code and commit the changes?

↑ 0 ↓ · flag

+ Comment

Tavpritesh Sethi · 2 days ago 🔗

The makeVector function accepts not only a vector but any numeric data structure. For example, *myVec<-makeVector(x=matrix(0,nrow=2,ncol=2))* gives no error. Setting the "vector" using *myVec$set<-matrix(1:4,ncol=2,nrow=2)* proceeds without warnings as well. However, the call *myVec$get()* unexpectedly shows

```
     [,1] [,2]
[1,]    0    0
[2,]    0    0
```

implying that the "vector" was not correctly set. Is there an explanation to this strange behavior?

↑ 0 ↓ · flag

Travis Reynolds · 2 days ago 🔗

I think the reason is that mVec$set() is a function. The matrix should be constructed within the brackets, and not assigned using <-. That's why myVec is still filled with 0's (its original content).

↑ 1 ↓ · flag

Tavpritesh Sethi · a day ago

Ahh. Thanks Travis. But allowing any numeric data-structure to go into makeVector was perhaps to keep it simple, I guess.

↑ 0 ↓ · flag

+ Comment

Dmitry Goldenberg Signature Track · 2 days ago

Honestly, folks, perhaps I'm missing some point here but why the need for all this extra complexity of setmean and cachemean??

All I need to do is "extend" the vanilla vector to not only hold its cached mean value but also be able to lazily compute it:

```
> makeVector <- function(x = numeric()) {
        m <- NULL
        set <- function(y) {
                x <<- y
                m <<- NULL
        }
        get <- function() x
        getmean <- function() {
             if(!is.null(m)) {
               print("getting cached data")
               return(m)
             }
             m <<- mean(x)
             print("computed cached data and returning it")
             m
        }
        list(set = set, get = get, getmean = getmean)
 }
>
> x <- makeVector(c(2:10))
>
> x$getmean()
[1] "computed cached data and returning it"
[1] 6
>
> x$getmean()
```

```
[1] "getting cached data"
[1] 6
```

Could someone comment and let us all know why we need the 2 main functions? Thanks.

⬆ 2 ⬇ · flag

Mark Elliott Hack · 19 hours ago 🔗

The only reason is that you were asked to do that. I happen to agree that this is artificial and since the makeCacheMatrix has to be intervened so that the value is cleared, these are two dependent routines.

IMHO if I implemented this for my use, I would intervene only makeCacheMatrix and add a solve method. If there were a clean way of extending a generic cached class without having to touch the base class then OK , extend it. I can not find a way to do that, so honestly one class with an additional method is IMHO a cleaner design point. I detest separate code which is tightly coupled, but we are not here to discuss the sense of the exercise.

I have done what was asked, created two tightly coupled methods, and charge into the valley of death with the 600.

⬆ 0 ⬇ · flag

Karen Williams Signature Track · 18 hours ago 🔗

The point of the exercise wasn't to write the function. The point of the exercise was using our GitHub account to fork a repo, make changes, and check them in to our forked repo. (Take a look at what we're actually being graded on.)

That being said, I did learn a little bit more about R programming, so that's something as well.

⬆ 0 ⬇ · flag

Cara Signature Track · 17 hours ago 🔗

where did you find what we are graded on? i looked and couldnt find it

⬆ 0 ⬇ · flag

Karen Williams Signature Track · 17 hours ago 🔗

When you go to the assignment itself, just keep scrolling down. There are sections where it shows what our peers are supposed to grade us on. Most of it is about whether we had the right kind of GitHub link. We're told to not test the code, just decide if it looks correct.

⬆ 0 ⬇ · flag

+ Comment

Bernd Schnabl Signature Track · a day ago

Hi
Very helpful thread !!
Still one question nags me.
What exactly is the purpose of the parameter x in the function makeVector. (see comment 1 below)
I cannot see that it is used anywhere !!

```
makeVector <- function(x = numeric()) {  ## comment 1 **) what is this x used for ?
  m <- NULL

  set <- function(y) {
    x <<- y
    m <<- NULL
  }
  get <- function() x ## this is the vector from the global environment and not the param
eter x from the
                      ## makeVector function

  setmean <- function(mean) m <<- mean
  getmean <- function() m

```

↑ 3 ↓ · flag

David McDuffee · a day ago

It's the vector that's being passed to makeVector, or (the default, if no vector is passed) it's an empty numeric vector.

The same "x" is returned by the "get" function, and the same "x" is also set by the "set" function. Notice that in the "set" function, the <<- operator is used -- that's to bump up the search to the "parent" environment of the code in the "set" function, which is a child inside makeVector().

↑ 0 ↓ · flag

Dmitry Goldenberg Signature Track · a day ago

Bernd,

When you invoke makeVector, you're specifying the value of the x parameter, like so:

makeVector(c(2:10))

c(2:10) is the value of x, it's a numeric vector of your choosing. Saying x=numeric(), I believe, tells R to initialize the x parameter to an empty numeric vector by default, so you can call makeVector(). If you do that:

makeVector()$get

R will say numeric(0) meaning you've got an empty numeric vector created.

If you don't provide that default in function definition, then call makeVector()$get, R will complain:

Error in q$get() : argument "x" is missing, with no default

↑ 0 ↓ · flag

Tavpritesh Sethi · a day ago 🔗

makeVector(matrix(0,3,3)) also works! I think there should be a condition to check if the numeric input is indeed of the class vector.

↑ 0 ↓ · flag

Rinat Abdurafikov · a day ago 🔗

It seems the x is NOT from the global environment. At least when you just enter "x" and hit "enter" in the console, the object is not found. It is not seen in RStudio's "environment view" (that top-right window)..

To me it feels like a hidden attribute of an object you create with makeVector() function. But how to see it explicitly is an open question.

↑ 0 ↓ · flag

David McDuffee · a day ago 🔗

For most purposes, a matrix is simply a vector with a dimension attribute.

```
> x = matrix(0,3,3)
> x
     [,1] [,2] [,3]
[1,]    0    0    0
[2,]    0    0    0
[3,]    0    0    0
```

```
> dim(x) = c(1,9)
> x
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[1,]    0    0    0    0    0    0    0    0    0
> dim(x) = c(9,1)
> x
      [,1]
 [1,]    0
 [2,]    0
 [3,]    0
 [4,]    0
 [5,]    0
 [6,]    0
 [7,]    0
 [8,]    0
 [9,]    0
```

If someone asks for the mean of those numbers, why not just give it to them?

↑ 0 ↓ · flag

Dmitry Goldenberg Signature Track · 21 hours ago

Are you saying, why bother with cachemean? The idea was to cache the value once you compute it, so you don't have to recompute it, which may be time-consuming/expensive especially if the vector is large.

↑ 0 ↓ · flag

David McDuffee · 21 hours ago

Dmitry, if that question is directed at me, no, I was not saying "why bother with cachemean". I was replying to Tavpritesh Sethi's comment a couple of posts back which implied (to me) that it shouldn't work if you passed it a matrix.

↑ 0 ↓ · flag

+ Comment

Don Voltz · 21 hours ago

It took me a while to understand with is going on with this code. I come from an OOP background and we are in essence creating an object but this is not like objects in many OOP languages. It is more a psudo-object with some work-arounds to function like any other object.

In makeVector, we are returning an R list collection. In R, functions can be assigned to variables which is what is going on in this case. get, set, getmean, and setmean are all just variable names.

These variables are assigned to the functions with get and getmean returning either a vector or the mean of a vector, the set and setmean take a parameter and assign it to one of the variables in the overall function makeVector (x holds the vector and m holds the mean of the vector).

In OOP languages, you only need to define methods within a class which is what this looks like. I could not understand the meaning of the list assignment at the end of makeVector. What is happening in this R closure is all of the variables assigned to function are placed inside a list with their names being the same as the variables. Remember from an earlier lecture, lists contain objects. These can be unnamed objects, in which case they would be accessed using square bracket notation (ie [[1]] would return the first item in the list, ext). List items (components) can also be accessed by their name, if they were named (which is the case here.

So we are creating the list with the following code

```
list(set = set, get = get,
             setmean = setmean,
             getmean = getmean)
```

since this is the last thing done inside the makeVector() code, it is returned from this function.

Now when we assign the makeVector is a variable assigned to the function which wraps these other functions as well as the list.

This variable is passed to cachemean and is assigned to the variable x

So in essence, we have a number of variables assigned to the same block of code (makeVector and x).

To access items in the list (which is held by makeVector and x), we can use the character $.

So all of the calls to x$get or makeVector$get will return the vector because get is the variable assigned to the function that returns the initial vector passed into makeVector or assigned using the set method. We are in essence only accessing the list using the $ and the name of the variable, the magic comes because these variables are assigned to functions.

If you are familiar with OOP, this substitutes for the . operator in objects.

If we had an object makeVector in a true OOP language, you could access the method set and get by makeVector.set and makeVector.get respectively. You would not need to assign each variable ('method') to a named position in a list, like we are doing here. This adds to complexity and confusion, but once understood brings power to R that is otherwise not possible.

To answer some of the other questions about why use cachemean, you might gain some additional understanding looking at my code and the explanation I placed in the comments. You can see more of my thoughts on my submitted code found here.

https://github.com/dmv1/ProgrammingAssignment2/blob/master/cachematrix.R

hope this helps clear us some of the confusion.

Don

↑ 2 ↓ · flag

Madan Pushpakath · 16 minutes ago

It really helps..Thanks for the detailed explanation

↑ 0 ↓ · flag

+ Comment

Dilir Akhtar Khan Signature Track · 20 hours ago

Adam, thank you for the demonstration.

↑ 0 ↓ · flag

+ Comment

New post

To ensure a positive and productive discussion, please read our forum posting policies before posting.

B *I* Link <code> Pic Math Edit: Rich ▾ Preview

□ Make this post anonymous to other students

☑ Subscribe to this thread at the same time

Add post