

14 November, 2014
6:22 AM

[Forums / Programming Assignment 2 \(peer assessment\): Lexical Scoping](#)[Help](#)

Purpose of "set function"

[Subscribe for email updates.](#)No tags yet. [+ Add Tag](#)Sort replies by: [Oldest first](#) [Newest first](#) [Most popular](#)[joseph fernando](#) · a day ago 

Could someone kindly explain to me (as you would to a 3 year old) the purpose of the following bit of code:

```
set <- function(y) {  
  x <- y  
  m <- NULL  
}
```

when I run just the "set" function it says:

```
> a <- makeVector(c(1,2,3,4))  
> $get()  
Error: unexpected '$' in "$"  
> a$get()  
[1] 1 2 3 4  
> a$set()  
Error in a$set() : argument "y" is missing, with no default
```

Thanks in advance.

↑ 0 ↓ · flag

[Jim Pavlik](#) Signature Track · a day ago

I also don't understand the purpose of the set function and get the same error. And to add insult to injury, when I was "faking it til I was making it" and just transferred this set() function to my first attempt at an solution to the assignment, I got this error when I tried to run the makeCacheMatrix() function.

↑ 0 ↓ · flag

[+ Comment](#)

Sarah F Signature Track · a day ago

Firstly, you are getting that error because the set function requires an argument. The fact that it is

```
set <- function(y) {.....}
```

means that you need to pass an argument to set(). Try

```
a$set(c(1,2,3,4))
```

and you shouldn't get that error.

Secondly, for the purpose of set():

It allows you to change the data that is held within your list "a" without having to re-run makeVector().

Try running

```
a$set(c(5,6,7,8))
a$get()
```

Now you have a different vector in "a" and you can still use cachemean(a) to get the mean (it will be a different mean now).

As far as I can tell, set() is not really used or needed if you make your vector from makeVector().

↑ 2 ↓ · flag

+ Comment

joseph fernando · a day ago



Thanks Sarah. I thought the same, there is really no need for "set" if Makevector is what's used to initiate cache storage.

Would anyone else have a different view to Sarah and myself.

↑ 0 ↓ · flag

Bill Hilton · a day ago

You can use the set() method to reset the object to a new value, basically to reuse the object rather than create a new one, something you might do if building millions of these (say for a large array) and tight on memory.

The cache() function doesn't access set() but another function could.

↑ 1 ↓ · flag

[+ Comment](#)

Jim Pavlik Signature Track · 21 hours ago 

Thanks, Sarah and Bill. I'm not sure this solves my problem with my code, but it is very helpful nevertheless in understanding the example.

↑ 0 ↓ · flag

Bill Hilton · 21 hours ago 

Hi Jim, the error in your code (and Joseph's) is you need to pass a new vector when you call `a$set` ...

```
> a$set()  
Error in a$set() : argument "y" is missing, with no default
```

`a$set(c(5,6,7,8,9))` # this should work (or any vector) ... keeps object name 'a' but with new vector and sets cached flag to NULL. (edit: just noticed Sarah F gave the same advice :))

Try it. And access it again with cache and you'll see the first access is not cached (since we loaded a new value and set to NULL) but the second access will indicate that a new value has been cached.

It's basically a back-door way to reset the object's vector value to something new.

↑ 2 ↓ · flag

[+ Comment](#)

New post

To ensure a positive and productive discussion, please read our [forum posting policies](#) before posting.

B	<i>I</i>			Link	<code><code></code>	Pic	Math		Edit: Rich ▾	Preview
<div></div>										

- ☐ Make this post anonymous to other students
- ☒ Subscribe to this thread at the same time

Add post

Purpose of "set function"

https://class.coursera.org/rprog-009/forum/thread?thread_id=253