From George Wu to Everyone: 10:03 AM

https://rdrr.io/snippets/

From Kyle Wathen to Everyone: 10:04 AM

We have a large group. Please make sure to mute yourself so that others can hear.

From Jan de Jong to Everyone: 10:05 AM

maybe better if Host mutes everyone?

From Kun Chen to Everyone: 10:15 AM

where to get these information?

From Kyle Wathen to Everyone: 10:15 AM

What she is showing is in MS Teams if you cannot see the team we can add you

From Ting Bai to Everyone: 10:15 AM

I don't have teams installed

From Kyle Wathen to Everyone: 10:16 AM

Teams was pushed recently, try typing Teams into your Windows search it may be there

From Ron Yu to Everyone: 10:17 AM

Do you see an icon that says 'Microsoft Teams' on your desktop?

From Dara Liu to Everyone: 10:17 AM

No from my end

From Kyle Wathen to Everyone: 10:18 AM

Okay you can also get it from SharePoint https://gileadconnect.sharepoint.com/:f:/r/teams/CSEAA-RTraining/Shared%20Documents/Session%201?csf=1&web=1&e=8hLtfm

From Fan Lin to Everyone: 10:18 AM

it is on program list

From Me to Everyone: 10:18 AM

search Microsoft teams by clicking on the circle icon at the lower left of the windows

From George Wu to Everyone: 10:19 AM

If it's not installed, likely will need to restart computer to get new pushed updates - but in the meanwhile Kyle's link should work through Web broswer

From Susan Guo to Everyone: 10:21 AM

I can get to the Teams. I missed how to open R.

From George Wu to Everyone: 10:22 AM

You can search for the RStudio program and open it

From Mukta Bahl to Everyone: 10:22 AM

Please use the link sent by Kyle - it will take you to session 1 materials

From George Wu to Everyone: 10:22 AM

depending on installation, a shortcut may be on the desktop

From Laura Jiang to Everyone: 10:24 AM

The console just installed has no plus sign at the right upper corner.

From John Austin to Everyone: 10:25 AM

What is the difference between <- and = ? Why is <- preferred?

From George Wu to Everyone: 10:26 AM

When R was originally developed, the only assignment operator was <-

From Kyle Wathen to Everyone: 10:27 AM

In R <- and = can do the same but it can cause issues with previsous versions and also confused with comparingTo view the all panes - click View -> Panes

From Vinodh Paida to Everyone: 10:28 AM

i don't have teams

From Kyle Wathen to Everyone: 10:28 AM

If you don

From George Wu to Everyone: 10:28 AM

You can access the sharepoint link as well: https://gileadconnect.sharepoint.com/:f:/r/teams/CSEAA-RTraining/Shared%20Documents/Session%201?csf=1&web=1&e=8hLtfm

From Vinodh Paida to Everyone: 10:29 AM

thank you

From Lyndsey Fou (Contractor) to Everyone: 10:31 AM

Please email me if you need access to R Training MS Teams.

From Fan Lin to Everyone: 10:31 AM

yes

From John Austin to Everyone: 10:33 AM

If you multiply a float type by an integer, if the answer can be an integer, it seems to force it to be an integer in the end. How can you keep the float type for the answer?

From Rachel Lu to Everyone: 10:33 AM

\*\* can also be the exponent, right?

From George Wu to Everyone: 10:33 AM

yes tha tis correct \*\* can also be used

From Kyle Wathen to Everyone: 10:34 AM

by default if you do float\*integer you get a float example: 4.2 \* 2 = 8.4

From John Austin to Everyone: 10:34 AM

0.5 \* 10 = 5is the 5 an int or a float?it doesn't show as 5.

From Shu Zan to Everyone: 10:34 AM

Is variable name case sensitive?

From Kyle Wathen to Everyone: 10:35 AM

it is not strongly typed like other languesYes case sensitiveif you do class( 5) or class( 5.5 ) both are numeric

From John Austin to Everyone: 10:35 AM

ah ok...do they all take the same memory space?or does memory adjust based on what type of numeric they are?

From Nan Chen to Everyone: 10:38 AM

variables are case sensitiveobject.size(A) give you the memory size

From John Austin to Everyone: 10:40 AM

can you expand vectors and matrices after they are created, or is their size fixed at creation?

From Kyle Wathen to Everyone: 10:40 AM

On expanding we will talk more later but easy to expandif you have a vector vX <- c(1,2,3)you can expand by vX <- c(vX, 4,5,6)

From John Austin to Everyone: 10:43 AM

object.size(5) says the number 5 takes up 56 bytes. Is there another type that more efficiently utilizes memory for big data?Also, for expanding, does that rewrite the whole vector into a new memory location, or does it do it inplace?

From Kyle Wathen to Everyone: 10:43 AM

John we will cover the rewrite later

From Gregory Campbell to Everyone: 10:44 AM

is there a "date" datatype?

From George Wu to Everyone: 10:45 AM

The memory questions are more advanced, but if you would like to read some about it please see this link: http://adv-r.had.co.nz/memory.html

From Kyle Wathen to Everyone: 10:46 AM

Yes there is a date type, example d <- as.Date("1/1/2020")

From John Austin to Everyone: 10:46 AM

thanks George, i'll look at that

From Pankhil Shah to Everyone: 10:48 AM

Can we clear commands from Console? If so how?

From George Wu to Everyone: 10:49 AM

in R studio you can click on edit -> clear console

From Rahul Penta to Everyone: 10:49 AM

There is a clear console option on the right top corner

From Gregory Campbell to Everyone: 10:49 AM

the little "broom" icon in the upper right of he console window.

From George Wu to Everyone: 10:49 AM

or Ctrl+L

From John Austin to Everyone: 10:58 AM

to confirm, do R indices start at 0 or 1? I did:h <- c(1, 2, 3)h[0] returns numeric(0) while h[1] returns 1. It seems it's 1 based for indexing, but what does h[0] do?

From Nan Chen to Everyone: 11:00 AM

it starts from 1index for vectorIt is different from C and Python

From Shuo Wang to Everyone: 11:00 AM

Thanks Nan. Yes 1 is the first index in R. This is different from Python and C++ which start from 0.

From John Austin to Everyone: 11:00 AM

what is returned with index 0? is that the object itself?

From Shuo Wang to Everyone: 11:01 AM

It will return a blank value if you use index 0.So should be avoided.

From Gianna Huang to Everyone: 11:09 AM

What's the difference between NA and NULL in R?

From George Wu to Everyone: 11:12 AM

so generally, NULL is an 'empty' null object - and returned when a value is undefined; NA is a object of length 1 and usually represents missingyou can try class(NA) vs class(NULL)and if you try something like NA > 1, or NULL > 1 NA will evaluate to NA, but NULL will not since it's an empty object

From Rachel Lu to Everyone: 11:12 AM

class(NA)

From George Wu to Everyone: 11:12 AM

or something like, c(5,6,NULL,4) = 5 6 4c(5,6,NA,4) gives 5 6 NA 4

From Me to Everyone: 11:13 AM

NA is a value, generally considered as missing value; NULL means no value

From John Austin to Everyone: 11:37 AM

Is there a resource or cheat sheet you'd recommend that compares data types, common functionalities, etc... when moving from SAS to R or Python to R, so we can quickly find how we want to do what we want to do in R coming from another languge?

From Kyle Wathen to Everyone: 11:38 AM

Do you like cheat sheets? In R Studio Click Help -> Cheatsheets to see a list of useful R related cheat sheets or go to https://rstudio.com/resources/cheatsheets/These are helpful but we can try to locate a Python vs R cheatsheet, I have not found one

From George Wu to Everyone: 11:39 AM

I haven't looked for Python -> R, more for R -> Python, but this short link looks okay: https://towardsdatascience.com/essential-guide-to-translating-between-python-and-r-7cb18b786e5d

From John Austin to Everyone: 11:40 AM

thanks! :) both of those look very useful :D

From Kyle Wathen to Everyone: 11:40 AM

Note that for Python vs R: There is a BIG difference between the use of a negative index.Example:vY[ -2 ] in R will drop element 2 from the vector where as in Python that can index from the end of the vector

From John Austin to Everyone: 11:43 AM

yea, that would've messed me up :) thanksnow that we're back on that slide, does old\_vector <- c(old\_vector, 4) do the operation in place, or does it create a new vector and assign new memory address? I'm wondering for appending values in a large loop if there is a more efficient way to append values to a vector

From Kyle Wathen to Everyone: 11:44 AM

For a large loop it is better to allocate first rather than keep appending.

From John Austin to Everyone: 11:44 AM

ok

From Me to Everyone: 11:46 AM

Good to know. I usually keep appending in simulation. Need to change my habit now. :) thanks Kyle~and thanks Austin for asking

From Kyle Wathen to Everyone: 11:47 AM

It can be a bigger issue for efficiency if you are allocating a data frame that is large in a function. I will show why in future sessions.

From Me to Everyone: 11:51 AM

is matrix a better choice than data frame?or list?data frame looks easier if a dataset including different types of variables need to be returned from a function

From John Austin to Everyone: 11:51 AM

do vectors, lists, dataframes, etc... have their own methods, for instance vector.sum() instead of sum(vector)?

From Kyle Wathen to Everyone: 11:52 AM

matrix vs data.frame vs list depends more on what you need to do: data frame can have different types where a matrix has to be all the same type. If you need a collection of things like that are not all the same length I use a list often for thatJohn - Not like python you can call sum( vX )If you did sum and called with a matrix it will sum all elements

From Xu Wen to Everyone: 11:59 AM

what's the difference between marvel\_df[3] and marvel\_df[,3] ? both select the third column of matrix marvel\_df, the first is show in vertical in Console, second is show in horizontal.

From George Wu to Everyone: 12:01 PM

marvel\_df[,3] should show the 3rd column, while marvel\_df[3] will show the 3rd entry in the data frame (will go by row in 1st column, then next column)