



# Intro to Modular w/ R Functions

October 01, 2024

# Intro to Modular w/ R Functions

## Today's Co-organizer:

Jade Young

[jade.young@cuanschutz.edu](mailto:jade.young@cuanschutz.edu)

## Facilitators:

Yichi (Ella) Chen & Kewalin Samart

## Presenter:

Zhixin Lun

[zhixin.lun@cuanschutz.edu](mailto:zhixin.lun@cuanschutz.edu)

## Volunteer signup



## Today's sponsor:

Mitchell Lab, DBMI



Investigates augmented reality applications for public engagement and user-centered design for clinical guidelines on mobile devices

# Who We R!?



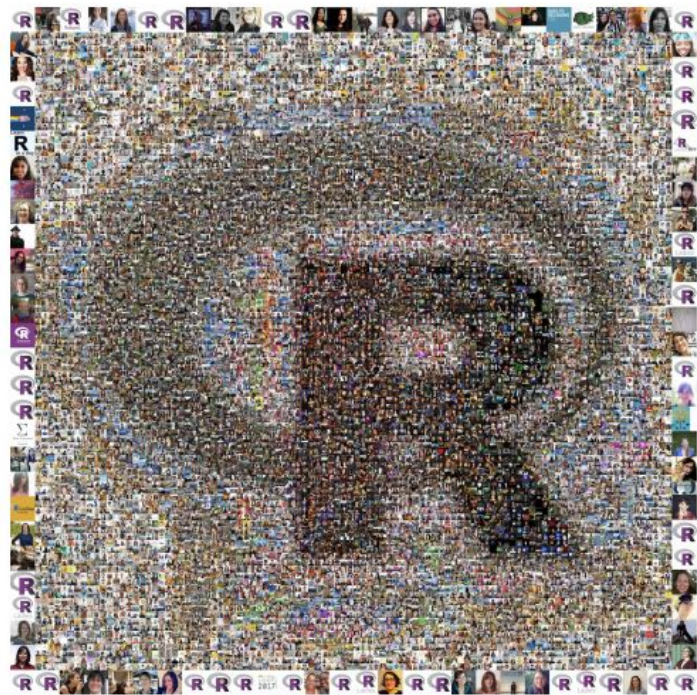
R-Ladies Kick-off Event on Jul 25, 2024




RLA Co-organizer team: Jade, Keenan, Janani, Kewalin, Stacey (left to right)




Worldwide organization that promotes **gender diversity** in the **#rstats** community via meetups and mentorship in a **friendly** and **safe** environment







Part of **R-Ladies - 240 groups**

**R-Ladies Aurora**

 Aurora, CO, USA

 69 members · Public group

 Organized by **R-Ladies Global** and **4 others**

# New Members?

1. Join us, RSVP for events on **Meetup**

<https://www.meetup.com/rladies-aurora/>

2. Join our discussion forum on **Discord**

<https://bit.ly/RLA-discord>



Part of **R-Ladies** - 240 groups

**R-Ladies Aurora**

★★★★★ (6)

Aurora, CO, USA

104 members · Public group

Organized by **R-Ladies Global** and 4 others

Share:

[About](#) [Events](#) [Members](#) [Photos](#) [Discussions](#) [More](#)

[Create event](#) [Manage group](#)

### What we're about

**Mission**  
*R-Ladies* is a worldwide organization that promotes gender diversity in the open-source R community.

**Who we are/What we do**  
*R-Ladies Aurora* is a local chapter of R-Ladies Global (<https://www.rladies.org>). We meet to discuss, learn, teach, present, and work on all things R! Our priority is to provide a safe community space for anyone identifying as a minority gender interested in and working with R. The group exists to promote gender diversity in the R community worldwide. We are pro-actively inclusive of queer, trans, and all minority identities, with additional sensitivity to intersectional identities. We meet in person or virtually to learn about the R programming language, algorithms, and advanced tools. Our events are open for everyone to attend.

**Who could join?**  
*R-Ladies Aurora* welcomes members of all R proficiency levels, whether you are a new or aspiring R user or an experienced R programmer interested in mentoring, networking & expert upskilling. Our non-profit, civil society community is designed to develop our members' R skills & knowledge through social, collaborative learning & sharing. Supporting minority identity access to STEM skills & careers, the Free Software Movement, and contributing to the global R community! Anyone and everyone interested in R/Data Science should join our group and participate/contribute in any way you can, be they learners, teachers, developers, or innovators! We encourage women and minority genders to present/lead most sessions and conversations, *but* we have no restrictions regarding membership, participation, and discussions. So, please join us

### Organizers

**R-Ladies Global and 4 others**  
[Message](#)

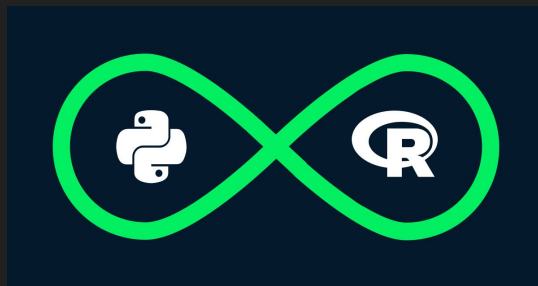
### Members (104)



# Upcoming Workshops

## Fall' 24 R-Ladies Aurora Series:

- **Intro to R Functions**
- DataViz 2.0
- R Package Development
- Presenting to Non-Scientific Audiences
- Python in R

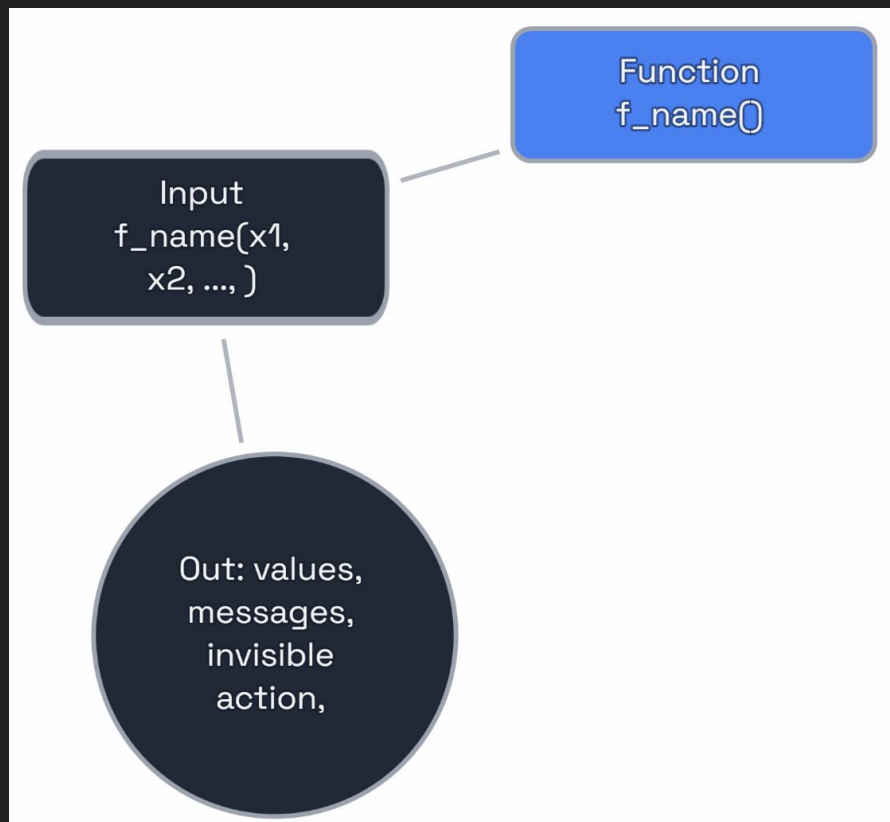


# Today! Oct 01, 2024

## TOPICS



- Intro to R functions

Ping us on Discord with any Questions




# Presentation Materials

<https://github.com/rladies-aurora>

**rladies-aurora**

Q Type ↵ to search

[Overview](#) [Repositories 5](#) [Discussions](#) [Projects](#) [Packages](#) [Teams](#) [People 4](#) [Settings](#)



## R-Ladies Aurora






12 followers <https://github.com/rladies-aurora> <https://meetup.com/rladies-aurora> <https://bit.ly/rladies-aurora-discord>

### Repositories






Q Find a repository... Type Language Sort New

**202410-rfunctions** Public






Forked from [rladies-eastlansing/2021-rfunctions](#)  
Workshop on Functions in R | @RLadiesAurora

 HTML  0  4  0  0 Updated now


**event\_planning** Private

 0  MIT  0  2  0 Updated last week

**.github** Public

 0  MIT  0  0  0 Updated last week

# Intro | Meet & Greet

- Pizza + Sign-up 
- Familiar w/ R and ggplot? → *Shuffle*
- Introduce yourself to your neighbor
- Who you are | Name, affiliation
- Do you have the same version of R (4.0+), RStudio & Tidyverse?
  - NO? Installation time!
    - <https://rstudio-education.github.io/hopr/starting.html>
- Need help? **Pink sticky up!** **All set: Blue!**





# R Functions

- You should consider writing a function whenever you've copied and pasted a block of code **more than twice**.  
-- Book by Wickham & Grolemund
- Book by Wickham & Grolemund, R4DS
  - Chapter 19 on Functions  
<https://r4ds.had.co.nz/functions.html>



# R Functions

Convert the temperature into different units:

$$C = (F - 32) \times \frac{5}{9}$$

```
C_vec[1] = (F_vec[1] - 32)*5/9
```

```
C_vec[2] = (F_vec[2] - 32)*5/9
```

```
C_vec[3] = (F_vec[3] - 32)*5/9
```

```
C_vec[4] = (F_vec[4] - 32)*5/9
```



# Function Structure

```
function_name <- function(argument1 , argument2, ...) {  
  # function body  
    Line 1  
    Line 2  
    ...  
}
```

Call a function:

```
function_name(argument1 = x, argument2 = y, ...)
```



Demo time!



# Function Structure

```
f_to_c <- function(f_value){  
  c_value = (f_value - 32)*5/9  
  return(c_value)  
}
```

Both **function** and **return** are keywords in R, you also notice the color in R Studio



# R Functions

Convert the temperature in different units:

$$C = (F - 32) \times \frac{5}{9}$$

```
C_vec = sapply(F_vec, FUN=f_to_c) ✓
```

**sapply**: apply the function on every element in the vector F\_vec.





What's *wrong* with this function?

```
current_temp_F = 100  
current_temp_C = 38
```

```
f_to_c_mtk <- function(f_value) {  
  current_temp_C = (f_value - 32)*5/9}
```

Attempting to change a global variable  
inside a function!!!

```
f_to_c_mtk(85)  
f_to_c_mtk(65)  
f_to_c_mtk(45)
```



# Local Variables and Global Variables

- Beginners are easy to mess up their difference:
  - **Global variables** are for the assignment done in the main environment or outside the chunk of function.
  - **Local variables** are for the assignment done inside the chunk of function.
- Don't attempt to change a global variable inside a function.
- **Correct way: return a value from the function and assign it to global variables.**
- Observe the Environment window in the RStudio all the time for new assignment.



## Correct way

```
current_temp_F = 100  
current_temp_C = 38
```

```
f_to_c_corr <- function(f_value) {  
  c_value = (f_value - 32)*5/9  
  return(c_value)  
}
```



```
current_temp_C = f_to_c_corr(85)  
current_temp_C = f_to_c_corr(65)  
current_temp_C = f_to_c_corr(45)
```



# From Easy to Moderate Function

```
f_to_c_human <- function(f_value) {  
  if (f_value >= 91.8 & f_value <= 100.8) {  
    c_value = (f_value - 32)*5/9  
    return(c_value)  
  }  
  else {  
    message("Too low or too high!!")  
  }  
}
```



# From Easy to Moderate Function

```
f_to_c_human_msg <- function(f_value) {  
  if (f_value >= 91.8 & f_value <= 100.8) {  
    c_value = (f_value - 32)*5/9  
    return(c_value)  
  }  
  else if (f_value < 91.8) {  
    message("Too low!!")  
  }  
  else {  
    message("Too high!!")  
  }  
}
```



# Error Control

- Beginners usually don't pay enough attention on the error control.
- A lot of programming bugs are due to inappropriate error control.
- Error control can handle the incorrect or unexpected input.
- The function `f_to_c_human_msg()` is an example with error control.
- Think about what is your expected result when encounter unexpected inputs.
  - Ignore the error and keep the function running?
  - Stop the function?





# Stop the function

```
f_to_c_human_stop <- function(f_value) {  
  if (f_value >= 91.8 & f_value <= 100.8) {  
    c_value = (f_value - 32)*5/9  
    return(c_value)  
  }  
  else if (f_value < 91.8) {  
    stop("Too low!!") # stop the function  
  }  
  else {  
    stop("Too high!!") # stop the function  
  }  
}
```



## Compare the difference

```
C_vec = sapply(F_vec, FUN = f_to_c_human_msg)
C_vec = sapply(F_vec, FUN = f_to_c_human_stop)
```

message: Keep the function running and store NULL values for invalid input in C\_vec.

stop: stop the function and reset the vector C\_vec.



# You Don't Need to Reinvent the Wheel!

- Try to explore if there are some existing functions or packages in R.
- Create your own functions ONLY if you can't find any available functions.
- Don't limit the function output as values, numbers, messages. It can be figures!
- Let's try to run a function using **ggplot2** functions.



# Good Coding Practices

- Function names: verbs, never start with numbers. Consider using underscores to separate words, e.g., `impute_missing()`, `calc_avg()`.
- Argument names: nouns
- Output: explicitly return values or messages using `return()`, `message()`



# Useful Materials

- R for Data Science <https://r4ds.had.co.nz/>
- Advanced R <https://adv-r.hadley.nz/>
- Best Coding Practices for R  
<https://bookdown.org/content/d1e53ac9-28ce-472f-bc2c-f499f18264a3/>
- Intro to functions in R, R-Ladies East Lansing  
<https://github.com/rladies-eastlansing/2021-rfunctions>



# Thank you!

R-Ladies Aurora

