

# **PROG102: Functions**

**Writing your own functions in R**

**MARINCS 100B | Intro to Marine Data Science | Winter 2025**

## Key concepts

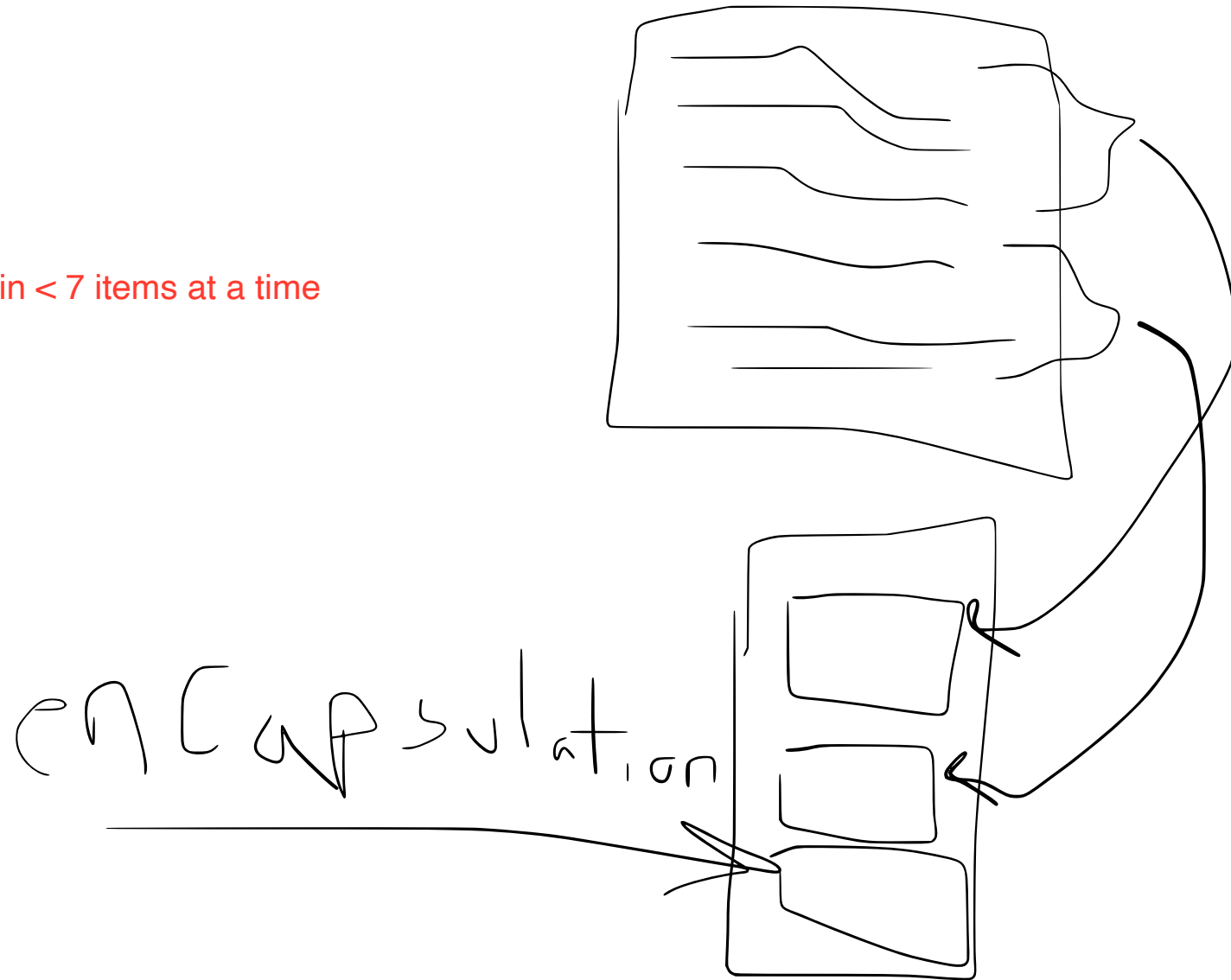
functions have 2 purposes:

- 1.) hide the details behind some kind of easy-to-remember name - this is called “encapsulation”
- 2.) you can apply the same code over and over again to new inputs (reusability)

syntax: the exact way of writing functions so that R will recognize them as functions

**Easy to read**

cognitive load of a human brain  $< 7$  items at a time

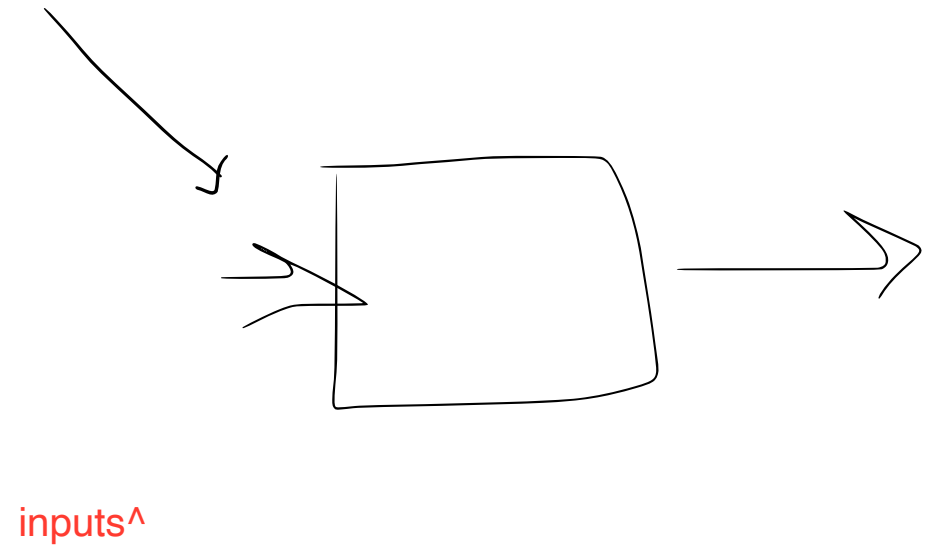
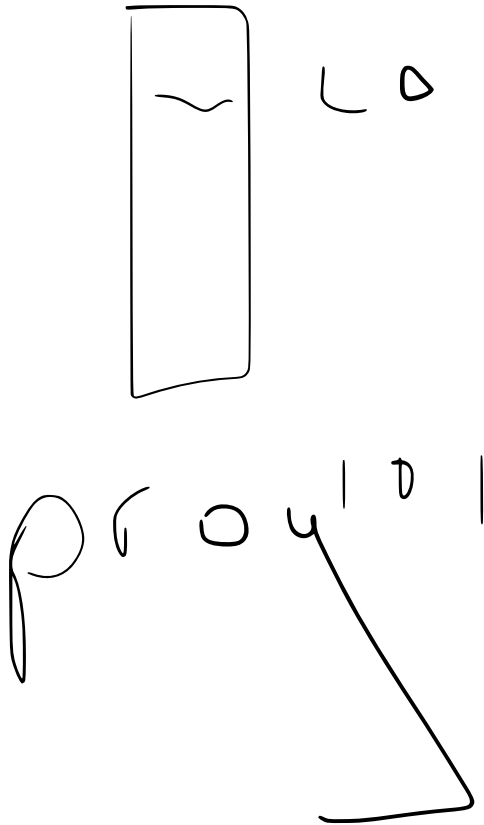


## Reusable

in prog 101, we had a piece of code that worked on a certain input (ex: site, date)

- if we wanted to change it and work on smth else, we could have to copy/paste it over and then edit it. this is not a great way to do it because there's a huge risk for error even when you're super experienced

- now, let's use PARAMETERS



# Syntax

**Demo in R**

## Recap

## **New vocabulary and lingering questions**

New vocabulary

Lingering questions



## Exercises

Label the five parts of this function:

```
first_and_last <- function(s) {  
  first_char <- substr(s, 1, 1)  
  last_char <- substr(s, nchar(s), 1)  
  result <- paste(first_char, last_char)  
  return(result)  
}
```

## Exercises

Match the function bodies on the left with the name that describes what they're doing on the right.

```
function(x) {  
  result <- x + 1  
  return(result)  
}
```

double

```
function(a) {  
  result <- a * 2  
  return(result)  
}
```

hypotenuse\_length

```
function(a, b) {  
  c_squared <- a^2 + b^2  
  result <- sqrt(c_squared)  
  return(result)  
}
```

increment

## Exercises

Write a function that turns a vector into a palindrome. For example, it should turn 1 2 3 into 1 2 3 3 2 1. Hint: you'll have to use a function called `rev()`. Choose a short but descriptive name for your function.

# **PROG102: Functions**

**How functions execute**

## **Key concepts**

## **The black box**

**Demo in R**

## Recap



## **New vocabulary and lingering questions**

New vocabulary

Lingering questions

## Exercises

- What value does the following code yield?
- How could you change `fish_mass` so the code yields 12 instead?
- How could you change the body of the function so the code yields 12?

```
fish_mass <- 5
temperature <- 20
fish_growth <- function(mass, temp) {
  growth <- 2 + 0.2 * temp
  mass <- mass + growth
  return(mass)
}
fish_growth(fish_mass, temperature)
```

## Exercises

In your own words, why does running this code generate an error?

```
calc_volume <- function(height, width, depth) {  
  area <- height * width  
  volume <- area * depth  
  return(volume)  
}  
vol <- calc_volume(3, 5, 1)  
area
```

# **PROG102: Functions**

**Default and named parameters**

## **Key concepts**

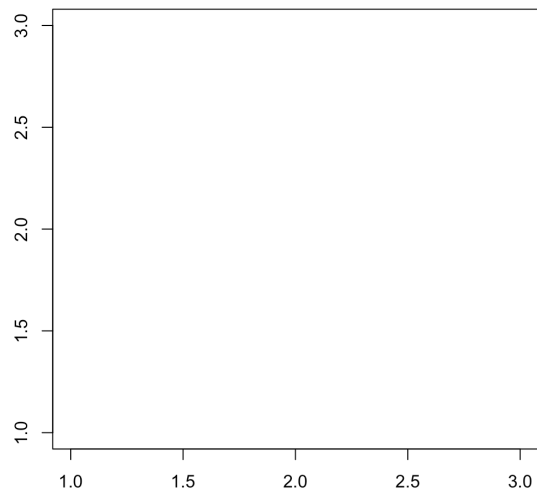
## **Default and named parameters**

```
round(x, digits = 0)
```

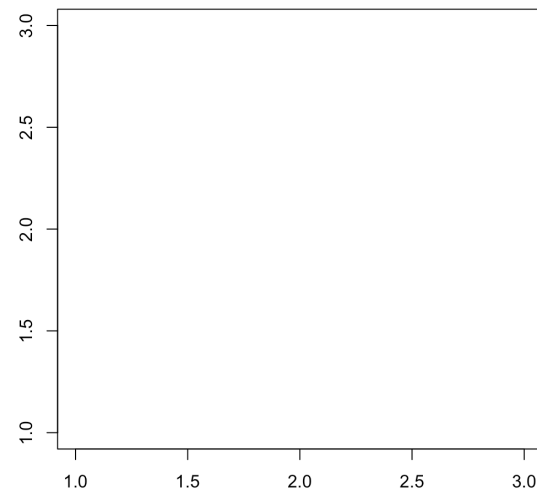
## Long parameter lists

```
plot(x, y = NULL, type = "p", xlim = NULL, ylim = NULL,  
     log = "", main = NULL, sub = NULL, xlab = NULL, ylab = NULL,  
     ann = par("ann"), axes = TRUE, frame.plot = axes,  
     panel.first = NULL, panel.last = NULL, asp = NA,  
     xgap.axis = NA, ygap.axis = NA,  
     ...)
```

```
plot(c(1, 2, 3), c(3, 2, 1))
```



```
plot(c(1, 2, 3), c(3, 2, 1),  
     xlab = "x", ylab = "y")
```



**Demo in R**



## Triple dots

```
max(..., na.rm = FALSE)
```

```
paste(..., sep = " ", collapse = NULL, recycle0 = FALSE)
```

## Recap

## **New vocabulary and lingering questions**

New vocabulary

Lingering questions

## Exercises

R represents *missing* data with the value NA. Say you're doing an experiment and you miss the second observation. In R you can write that as `c(1, NA, 3, 4)`.

Most summary functions, like `mean()`, `max()`, and `median()`, have a parameter called `na.rm`. What does this parameter do? What is its default value? How would you get the maximum value of the vector `c(1, NA, 3, 4)`?