Function Definitions Intro

CSC121
Mark Kazakevich
Last time, we talked about parts 1 and 3

The three parts to a what a function in R does:
1. ‘Call’ the function with some input
2. Do something with that input
3. Produce a ‘Return value’
Let’s talk about Part 2

The three parts to a what a function in R does:
1. ‘Call’ the function with some input
2. Do something with that input
3. Produce a ‘Return value’
We saw that Functions hide the details

- We don’t know how exactly R computes the square root of a number
- But we know that we have a `sqrt` function that does it and we’re happy to use it
- We feed it an argument, and it does the work to give us the square root of the number
That’s nice, but what if we want to do more than just use the built-in functions?

- For example, let’s take the \texttt{sin} function, which computes the sine of an angle, specified in \texttt{radians}
  - \texttt{\textgreater{} sin(pi / 2)}
    - \texttt{\textgreater{} [1] 1}

- But what if we wanted to use the \texttt{sin} function with degrees instead of radians?
  - We would have to convert from degrees to radians, and put that as the argument to \texttt{sin}
    - \texttt{radians = degrees*( π / 180)}
sin with degrees examples

● Find sine of 90 degrees
  ○ radians = 90 * (π / 180)

> sin(90 * (pi / 180))
> [1] 1

● Find sine of 173 degrees
  ○ radians = 173 * (π / 180)

> sin(173 * (pi / 180))
> [1] 0.1218693

● Find sine of 270 degrees
  ○ radians = 270 * (π / 180)

> sin(270 * (pi / 180))
> [1] -1

● Find sine of 0 degrees
  ○ radians = 0 * (π / 180)

> sin(0 * (pi / 180))
> [1] 0
What’s the problem with this approach?

- We have to convert to radians every time, and we have to write out this much bigger argument to \( \sin \) every time.
- This can take a long time, especially if you’re doing this calculation often.
- What’s a good solution?
Wouldn’t it be nice to have a sine function that already has the angle in degrees as the argument?
Well, that function doesn’t exist in R, so there’s only one thing we can do...
Create our own!
How do we ‘define’ a function?

FunctionName <- function(arguments) {
  # function body
}

How do we ‘define’ a function?

1. Give it a name
2. State that it’s a function
3. Give each argument a name
4. Write the R commands that make the function work

```r
FunctionName <- function(arguments) {
  # function body
}
```
The Function Body

FunctionName <- function(arguments) {
  # function body
  
  This is where Part 2 happens; Where we do something with our input arguments
}
Let's see (in RStudio) how we would create a function we can use