Vectors

Raphael Carvalho

15/05/2019

Vectors

First, create a numeric vector num_vect that contains the values 0.5, 55, -10, and 6.

```
num_vect <- c(0.5, 55, -10, 6)
```

Now, create a variable called tf that gets the result of num_vect < 1, which is read as ‘num_vect is less than 1’.

```
tf <- num_vect < 1
```

What do you think tf will look like?

```
print("a vector of 4 logical values")
```

```
## [1] "a vector of 4 logical values"
```

Let’s try another. Type num_vect >= 6 without assigning the result to a new variable.

```
num_vect >= 6
```

```
## [1] FALSE  TRUE FALSE  TRUE
```

(3 > 5) & (4 == 4)

```
print(1: FALSE)
```

```
## [1] 1 0
```

(TRUE == TRUE) | (TRUE == FALSE)

```
print(2: TRUE)
```

```
## [1] 2 1
```

((111 >= 111) | !(TRUE)) & ((4 + 1) == 5)

```
print(2: TRUE)
```

```
## [1] 2 1
```

Create a character vector that contains the following words: “My”, “name”, “is”. Remember to enclose each word in its own set of double quotes, so that R knows they are character strings. Store the vector in a variable called my_char.

```
my_char <- c("My", "name", "is")
```

Type paste(my_char, collapse = " “) now. Make sure there’s a space between the double quotes in the `collapse` argument. You’ll see why in a second.

```
paste(my_char, collapse = " ")
```

```
## [1] "My name is"
```

To add (or 'concatenate') your name to the end of my_char, use the c() function like this: c(my_char, "your_name_here"). Place your name in double quotes where I've put "your_name_here". Try it now, storing the result in a new variable called my_name.

```
my_name <- c(my_char, "Raphael")
```

In the simplest case, we can join two character vectors that are each of length 1 (i.e. join two words). Try paste("Hello", "world!", sep = " "), where the `sep` argument tells R that we want to separate the joined elements with a single space.

```
paste("Hello", "world!", sep = " ")
```

```
## [1] "Hello world!"
```

For a slightly more complicated example, we can join two vectors, each of length 3. Use paste() to join the integer vector 1:3 with the character vector c("X", "Y", "Z"). This time, use sep = "" to leave no space between the joined elements.

```
 paste(c(1:3), c("X", "Y", "Z"), sep = "")
```

```
## [1] "1X" "2Y" "3Z"
```

Vector recycling! Try paste(LETTERS, 1:4, sep = "-"), where LETTERS is a predefined variable in R containing a character vector of all 26 letters in the English alphabet.

```
 paste(LETTERS, 1:4, sep = "-")
```

```
##  [1] "A-1" "B-2" "C-3" "D-4" "E-1" "F-2" "G-3" "H-4" "I-1" "J-2" "K-3"
## [12] "L-4" "M-1" "N-2" "O-3" "P-4" "Q-1" "R-2" "S-3" "T-4" "U-1" "V-2"
## [23] "W-3" "X-4" "Y-1" "Z-2"
```