RProgramming

tubako

8/30/2020

# R Console Input and Evaluation

### Entering input

x <- 1  
print(x)

## [1] 1

x

## [1] 1

msg <- "hello"  
  
## x <- ##Incomplete expression

### Evaluation

x <- 5 ## nothing happens  
x ## auto-printing occurs

## [1] 5

print(x) ## explicit printing

## [1] 5

## [1] indicates that x is a vector and 5 is the first element.

### Printing

## The : operator is used to create integer sequences.  
x <- 1:20   
x

## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

# Data Types

### R Objects and Attributes

#### Objects

R has 5 basic classes of objects

* character
* numeric (real numbers)
* integer
* complex
* logical (T/F)

The most basic object is a vector

* A vector can only contain objects of the same class
* BUT: The one exception is a *list*, which is represented as a vector but can contain objects of different classes (indeed, that’s usually why we use them)

Empty vectors can be created with the

vector()

## logical(0)

function.

#### Numbers

* Numbers in R are generally treated as numeric objects (i.e. double precision real numbers)
* If you explicitly want an integer, you need to specify the **L** suffix
* Ex: Entering 1 gives you a numeric object; entering 1**L** explicitly gives you an integer.
* There is also a special number **Inf** which represents infinity: e.g. 1/0; **Inf** can be used in ordinary calculations; e.g. 1/**Inf** is 0
* The value **NaN** represents an undefined value ("not a number); e.g. 0/0; **NaN** can also be thought of as a missing value

#### Attributes

R objects can have attributes

* names, dimensions
* dimensions (e.g. matrices, arrays)
* class
* length
* other user-defined attributes/metadata

Attributes of an object can be accessed using the

##attributes()

function.

### Vectors and Lists