# Basics of R

Laika Jean Paculba

March 15, 2023

use <- operator to store an object # can be used to put a comment

my_sum <- 3 + 3 my_sum + 3

my_sum # show the value of my_sum new_sum <- my_sum + 3 #assign my_sum to new_sum and show the value

## integers can be signified by adding an 'L' to the end

my_integer <- 1L my_double <- 6.38

## typeof() function - used to see the type of a singe scalar value

typeof(my_integer) typeof(my_double)

## Numeric Data

can be in integer form or double (decimal) form.

my_integer <- 1L my_double <- 6.38 typeof(my_integer) typeof(my_double)

my_integer <- 9L typeof(my_integer) my_double <- 7.98 typeof(double)

## Character Data

text data surrounded by single or double quotes

my_character <- "This is text" typeof(my_character)

## Logic Data

takes the form TRUE or FALSE

my_logical <- TRUE typepf(my_logical) *verify if its indeed logical* my_logical <- TRUE typeof(TRUE)

my_double_vector <- c(2.3, 6.8, 4.5, 65, 6) #define str(my_double_vector) #verify type of(my_double_vector)

## Homogenous Data Structures

### Vectors

one-dimensional structures containing data of the same type and are notated by using c().

*define a vector of the factor which is not* categories <- factor(c("A", "B", "V", "A", "C")) # factor - defines it as an integer str(categories) # verify

#character vector ranking <- c("Medium", "High", "Low") str(ranking) #turn it to order factor ranking_factors <- ordered( ranking, levels = c("Medium", "High", "Low") ) str(ranking_factors) # contents and type of the vector which is ordered type

((ranking <- c("Medium", "High", "Low")) ranking_order <- order(ranking, levels(c("Medium", "High", "Low")))

ranking_order <- order(ranking, levels(c("Medium", "High", "Low"))) str(ranking_order) # contents and type of the vector which is ordered type))))

link(categories)

simple numeric sequence vectors (my_sequence <- 1:10) # (my_sequence_2 <- seq(from=1, to=10)) same above, flexible than above (seq_five <- seq(from=5, to=500, by=5)) #example

coercion - result of poor design

## numeric sequence vector

vec <- 1:5 str(vec) #create a new vector containing vec and the character hello new_vec <- c(vec, "hello") # creates a vector forom the precious vector # numeric value have been coerced str(new_vec) # the new elements become a type elements vec[1]+ vec[2] # result is 3 new_vec[1] + new_vec[2] # result to error # keep in mind to know what is the type of data

mix <- c(TRUE, 6) str(mix) # result = num [1:2] 1 6 new_categories <- c(categories, 1) str(new_categories)

matrices - two dimensional you can convert a vector to matrix

#create a 2x2 matrix with the first four integers (m <- matrix( c(1, 2, 3, 4), nrow = 2, ncol = 2 )) (m <- matrix(c(1,2,3,4), nrow=2, ncol=3)) (m <- matrix(c(1,2,3,4), nrow=3, ncol=3))

Arrays - are n-dimensional data structure with the same data type and are not used extensively by most r users # not really im portant