# Test Types

int\_var <- c(1L, 6L, 10L)

typeof(int\_var)

[1] "integer"

is.integer(int\_var)

[1] TRUE

is.atomic(int\_var)

[1] TRUE

dbl\_var <- c(1, 2.5, 4.5)

typeof(dbl\_var)

[1] "double"

is.double(dbl\_var)

[1] TRUE

is.atomic(dbl\_var)

[1] TRUE

is.numeric(int\_var)

[1] TRUE

is.numeric(dbl\_var)

[1] TRUE

# Test Dates

origin <- "1970-01-01"

dt\_origin <- as.Date(origin)

dt\_origin\_n <- as.numeric(dt\_origin)

dt\_origin\_n

[1] 0

dt\_origin

[1] "1970-01-01"

birthdays <- c("1963-02-12", "1968-07-13")

birthday\_dates <- as.Date(birthdays)

birthday\_dates

[1] "1963-02-12"

[2] "1968-07-13"

# Test DataFrame

temp <- c(20.37, 18.56, 18.4, 21.96, 29.53, 28.16, 36.38, 36.62, 40.03, 27.59, 22.15, 19.85)

humidity <- c(88, 86, 81, 79, 80, 78, 71, 69, 78, 82, 85, 83)

rain <- c(72, 33.9, 37.5, 36.6, 31.0, 16.6, 1.2, 6.8, 36.8, 30.8, 38.5, 22.7)

month <- c("January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December")

data <- data.frame(month = month, temperature = temp, humidity = humidity, rain = rain)

# Names of the variables (columns)

names(data)

print(head(data), digits=1)

# DataSet

datasets::mtcars

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | mpg | cyl | disp | hp | drat | wt | qsec | vs | am | gear | carb |
| Mazda RX4 | 21 | 6 | 160 | 110 | 3.9 | 2.62 | 16.46 | 0 | 1 | 4 | 4 |
| Mazda RX4 Wag | 21 | 6 | 160 | 110 | 3.9 | 2.875 | 17.02 | 0 | 1 | 4 | 4 |
| Datsun 710 | 22.8 | 4 | 108 | 93 | 3.85 | 2.32 | 18.61 | 1 | 1 | 4 | 1 |
| Hornet 4 Drive | 21.4 | 6 | 258 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 | 3 | 1 |
| Hornet Sportabout | 18.7 | 8 | 360 | 175 | 3.15 | 3.44 | 17.02 | 0 | 0 | 3 | 2 |
| Valiant | 18.1 | 6 | 225 | 105 | 2.76 | 3.46 | 20.22 | 1 | 0 | 3 | 1 |
| Duster 360 | 14.3 | 8 | 360 | 245 | 3.21 | 3.57 | 15.84 | 0 | 0 | 3 | 4 |
| Merc 240D | 24.4 | 4 | 146.7 | 62 | 3.69 | 3.19 | 20 | 1 | 0 | 4 | 2 |
| Merc 230 | 22.8 | 4 | 140.8 | 95 | 3.92 | 3.15 | 22.9 | 1 | 0 | 4 | 2 |
| Merc 280 | 19.2 | 6 | 167.6 | 123 | 3.92 | 3.44 | 18.3 | 1 | 0 | 4 | 4 |
| Merc 280C | 17.8 | 6 | 167.6 | 123 | 3.92 | 3.44 | 18.9 | 1 | 0 | 4 | 4 |
| Merc 450SE | 16.4 | 8 | 275.8 | 180 | 3.07 | 4.07 | 17.4 | 0 | 0 | 3 | 3 |
| Merc 450SL | 17.3 | 8 | 275.8 | 180 | 3.07 | 3.73 | 17.6 | 0 | 0 | 3 | 3 |
| Merc 450SLC | 15.2 | 8 | 275.8 | 180 | 3.07 | 3.78 | 18 | 0 | 0 | 3 | 3 |
| Cadillac Fleetwood | 10.4 | 8 | 472 | 205 | 2.93 | 5.25 | 17.98 | 0 | 0 | 3 | 4 |
| Lincoln Continental | 10.4 | 8 | 460 | 215 | 3 | 5.424 | 17.82 | 0 | 0 | 3 | 4 |
| Chrysler Imperial | 14.7 | 8 | 440 | 230 | 3.23 | 5.345 | 17.42 | 0 | 0 | 3 | 4 |
| Fiat 128 | 32.4 | 4 | 78.7 | 66 | 4.08 | 2.2 | 19.47 | 1 | 1 | 4 | 1 |
| Honda Civic | 30.4 | 4 | 75.7 | 52 | 4.93 | 1.615 | 18.52 | 1 | 1 | 4 | 2 |
| Toyota Corolla | 33.9 | 4 | 71.1 | 65 | 4.22 | 1.835 | 19.9 | 1 | 1 | 4 | 1 |
| Toyota Corona | 21.5 | 4 | 120.1 | 97 | 3.7 | 2.465 | 20.01 | 1 | 0 | 3 | 1 |
| Dodge Challenger | 15.5 | 8 | 318 | 150 | 2.76 | 3.52 | 16.87 | 0 | 0 | 3 | 2 |
| AMC Javelin | 15.2 | 8 | 304 | 150 | 3.15 | 3.435 | 17.3 | 0 | 0 | 3 | 2 |
| Camaro Z28 | 13.3 | 8 | 350 | 245 | 3.73 | 3.84 | 15.41 | 0 | 0 | 3 | 4 |
| Pontiac Firebird | 19.2 | 8 | 400 | 175 | 3.08 | 3.845 | 17.05 | 0 | 0 | 3 | 2 |
| Fiat X1-9 | 27.3 | 4 | 79 | 66 | 4.08 | 1.935 | 18.9 | 1 | 1 | 4 | 1 |
| Porsche 914-2 | 26 | 4 | 120.3 | 91 | 4.43 | 2.14 | 16.7 | 0 | 1 | 5 | 2 |
| Lotus Europa | 30.4 | 4 | 95.1 | 113 | 3.77 | 1.513 | 16.9 | 1 | 1 | 5 | 2 |
| Ford Pantera L | 15.8 | 8 | 351 | 264 | 4.22 | 3.17 | 14.5 | 0 | 1 | 5 | 4 |
| Ferrari Dino | 19.7 | 6 | 145 | 175 | 3.62 | 2.77 | 15.5 | 0 | 1 | 5 | 6 |
| Maserati Bora | 15 | 8 | 301 | 335 | 3.54 | 3.57 | 14.6 | 0 | 1 | 5 | 8 |
| Volvo 142E | 21.4 | 4 | 121 | 109 | 4.11 | 2.78 | 18.6 | 1 | 1 | 4 | 2 |

# Formatting Numbers

x = rnorm(10)

y = rnorm(10)

df = data.frame(x=x, y=y)

df

round(x, 5)

sprintf(x, fmt='%.3f')

a <- 1.234567896658745

options(digits=4)

digits integer int [1:1] 4

a[1] 1.23456789665875

round(a, 3)

[1] 1.235

# Test lm

x <- c(1,2,3,4,5,6,7,8,9,10)

y <- c(2,3,3,5,7,1,2,3,7,9)

df <- data.frame(x=x, y=y)

fm <- lm(y ~ x, data=df)

fm

$'call'

[1] "lm(formula = y ~ x, data = df)"

$'terms'

[1] "y ~ x"

fm$xlevels

# Test Lists

Manually construct a list

list\_data1 <- list("Red", "Green", c(21,32,11), TRUE, 51.23, 119.1)

list\_data1

list\_data2 <- list(c("Jan","Feb","Mar"), matrix(c(3,9,5,1,-2,8), nrow = 2), list("green",12.3))

list\_data2

# Give names to the elements in the list.

names(list\_data2) <- c("1st Quarter", "A\_Matrix", "A Inner list")

list\_data2

list\_data2[1]

list\_data2[3]

list\_data2$'A\_Matrix'

[,1] [,2] [,3]

[1,] 3 5 -2

[2,] 9 1 8

# Test Matrices

mat1 <- matrix(c(5,10,15,20,25,30,35,40,45), nrow = 3)

mat1

[,1] [,2] [,3]

[1,] 5 20 35

[2,] 10 25 40

[3,] 15 30 45

mat2 <- matrix(c(50,55,60,65,70,75,80,85,90), nrow = 3)

mat2

[,1] [,2] [,3]

[1,] 50 65 80

[2,] 55 70 85

[3,] 60 75 90

add\_output <- mat1 + mat2

add\_output

[,1] [,2] [,3]

[1,] 55 85 115

[2,] 65 95 125

[3,] 75 105 135

sub\_output <- mat1 - mat2

sub\_output

[,1] [,2] [,3]

[1,] -45 -45 -45

[2,] -45 -45 -45

[3,] -45 -45 -45

mat3<- matrix(c(2,4,6,8,10,12), nrow = 3,ncol =2)

mat3

[,1] [,2]

[1,] 2 8

[2,] 4 10

[3,] 6 12

mat4 <- matrix(c(14,16,18,20,22,24), nrow = 3,ncol=2)

mat4

[,1] [,2]

[1,] 14 20

[2,] 16 22

[3,] 18 24

final1 <- mat3 \* mat4

final1

[,1] [,2]

[1,] 28 160

[2,] 64 220

[3,] 108 288

final2 <- mat3 / mat4

final2

[,1] [,2]

[1,] 0.142857142857143 0.4

[2,] 0.25 0.454545454545455

[3,] 0.333333333333333 0.5

round(final2, 7)

[,1] [,2]

[1,] 0.1428571 0.4

[2,] 0.25 0.4545455

[3,] 0.3333333 0.5

final2[2,1]

[1] 0.25

final2[4,1]

Error in final2[4, 1] : subscript out of bounds

# Evaluate lm()

library(tidyverse)

library(datarium)

head(marketing)

cor(marketing$sales, marketing$youtube)

[1] 0.782224424861606

smodel <- lm(sales ~ youtube, data = marketing)

#smodel

smodel$'call'

[1] "lm(formula = sales ~ youtube, data = marketing)"

smodel $'terms'

[1] "sales ~ youtube"

# Test S4 class

#

setClass("employee", slots=list(name="character", id="numeric", contact="character"))

[1] "new("classGeneratorFunction", .Data = function (...) "

[2] "new("employee", ...), className = structure("employee", package = ".GlobalEnv"), "

[3] " package = ".GlobalEnv")"

#

obj <- new("employee", name="Steven", id=1002, contact="West Avenue")

obj

An object of class "employee"

Slot "name":

[1] "Steven"

Slot "id":

[1] 1002

Slot "contact":

[1] "West Avenue"

show\_emp2 <- function(object) {

c(object@name, object@id, object@contact)

}

show\_emp2(obj)

[1] "Steven"

[2] "1002"

[3] "West Avenue"

show\_emp2

[1] "function (object) "

[2] "{"

[3] " c(object@name, object@id, object@contact)"

[4] "}"

square <- function(x) {

return (x \* x)

}

square(2)

[1] 4

square(1:10)

[1] 1

[2] 4

[3] 9

[4] 16

[5] 25

[6] 36

[7] 49

[8] 64

[9] 81

[10] 100

square

[1] "function (x) "

[2] "{"

[3] " return(x \* x)"

[4] "}"

summary

[1] "function (object, ...) "

[2] "UseMethod("summary")"

UseMethod("summary")

Error in UseMethod("summary") :

'UseMethod' used in an inappropriate fashion

l = list()

l

list()

l[[1]] = 10

l[[1]]

[1] 10

l[1]

[[1]]

[1] 10