# 6\_Functions

#### 2022-10-03

# library(tidyverse)

```
## -- Attaching packages ------- tidyverse 1.3.1 --
## v ggplot2 3.3.5  v purrr  0.3.4
## v tibble 3.1.6  v dplyr  1.0.8
## v tidyr  1.2.0  v stringr 1.4.0
## v readr  2.1.2  v forcats 0.5.1

## -- Conflicts ------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag() masks stats::lag()
```

# 6. Functions

#### 6.1 Introduction

#### Quiz

1. What are the three components of a function?

arguments, body, and environment

2. What does the following code return?

```
x <- 10
f1 <- function(x) {
  function() {
    x + 10
  }
}
f1(1)()</pre>
```

## [1] 11

3. How would you usually write this code?

```
`+`(1, `*`(2, 3))
```

## [1] 7

4. How could you make this call easier to read?

```
mean(, TRUE, x = c(1:10, NA))
## [1] 5.5
```

5. Does the following code throw an error when executed? Why or why not?

```
f2 <- function(a, b) {
   a * 10
}
f2(10, stop("This is an error!"))</pre>
```

```
## [1] 100
```

- 6. What is an infix function? How do you write it? What's a replacement function? How do you write it?
- 7. How do you ensure that cleanup action occurs regardless of how a function exits?

#### 6.2 Function fundamentals

- Functions can be broken down into three components: arguments, body, and environment.
- Functions are objects, just as vectors are objects.

#### 6.2.1 Function components

A function has three parts:

## }

- The formals(), the list of arguments that control how you call the function.
- The body(), the code inside the function.
- The environment(), the data structure that determines how the function finds the values associated with the names.

```
f02 <- function(x, y) {
    # A comment
    x + y
}
formals(f02)

## $x
##
##
##
##
##
##
##
$y

body(f02)</pre>
```

```
environment(f02)
## <environment: R_GlobalEnv>
attr(f02, "srcref")
## function(x, y) {
## # A comment
##
   x + y
## }
6.2.2 Primitive functions
## function (..., na.rm = FALSE) .Primitive("sum")
## .Primitive("[")
typeof(sum)
## [1] "builtin"
typeof(`[`)
## [1] "special"
formals(sum)
## NULL
body(sum)
## NULL
environment(sum)
## NULL
```

6.2.3 First-class functions

```
f01 <- function(x) {</pre>
  sin(1 / x^2)
}
lapply(mtcars, function(x) length(unique(x)))
## $mpg
## [1] 25
##
## $cyl
## [1] 3
##
## $disp
## [1] 27
##
## $hp
## [1] 22
##
## $drat
## [1] 22
##
## $wt
## [1] 29
##
## $qsec
## [1] 30
##
## $vs
## [1] 2
##
## $am
## [1] 2
##
## $gear
## [1] 3
##
## $carb
## [1] 6
Filter(function(x) !is.numeric(x), mtcars)
## data frame with 0 columns and 32 rows
integrate(function(x) sin(x) ^ 2, 0, pi)
## 1.570796 with absolute error < 1.7e-14
funs <- list(</pre>
half = function(x) x / 2,
double = function(x) x * 2
```

```
funs$double(10)
```

## [1] 20

### 6.2.4 Invoking a function

```
args <- list(1:10, na.rm = TRUE)
do.call(mean, args)</pre>
```

## [1] 5.5

#### 6.2.5 Exercises

1. Given a name, like "mean", match.fun() lets you find a function. Given a function, can you find its name? Why doesn't that make sense in R?

It's a lot easier to match a name then it is to match all the code in a function to a name

2. It's possible (although typically not useful) to call an anonymous function. Which of the two approaches below is correct? Why?

```
function(x) 3()

## function(x) 3()

(function(x) 3)()
```

## [1] 3

2nd?

3. A good rule of thumb is that an anonymous function should fit on one line and shouldn't need to use {}. Review your code. Where could you have used an anonymous function instead of a named function? Where should you have used a named function instead of an anonymous function?

Should have used an anonymous on smaller quick functions. Use named functions on larget multi line chunks that need comments

4. What function allows you to tell if an object is a function? What function allows you to tell if a function is a primitive function?

```
is.primitive(sum)
## [1] TRUE
is.function(mean)
## [1] TRUE
  5. This code makes a list of all functions in the base package.
objs <- mget(ls("package:base", all = TRUE), inherits = TRUE)</pre>
funs <- Filter(is.function, objs)</pre>
  a. Which base function has the most arguments?
sort(unlist((lapply(
  lapply(funs, FUN = formals), length
))), decreasing = T) %>% head()
                                                                     formatC
##
                scan
                        format.default
                                                   source
##
                                                        16
                                                                          15
##
             library merge.data.frame
                  13
  b. How many base functions have no arguments? What's special about those functions?
a <- unlist((lapply(</pre>
  lapply(funs, FUN = formals), length
)))
sum(a == 0)
## [1] 253
head(a[a == 0])
                          %%
                       0
is.primitive(`-`)
## [1] TRUE
is.primitive("scan")
## [1] FALSE
```

```
# Are primitive functions
```

c. How could you adapt the code to find all primitive functions?

```
objs <- mget(ls("package:base", all = TRUE), inherits = TRUE)
funs <- Filter(is.primitive, objs)</pre>
```

6. What are the three important components of a function?

Formals, Body, Environment

7. When does printing a function not show the environment it was created in?

Primitives and functions created in the global environment

# 6.3 Function composition

```
square <- function(x) x^2</pre>
deviation <- function(x) x - mean(x)</pre>
x <- runif(100)
sqrt(mean(square(deviation(x))))
## [1] 0.2798161
out <- deviation(x)
out <- square(out)</pre>
out <- mean(out)</pre>
out <- sqrt(out)</pre>
## [1] 0.2798161
library(magrittr)
## Attaching package: 'magrittr'
## The following object is masked from 'package:purrr':
##
##
       set_names
## The following object is masked from 'package:tidyr':
##
##
       extract
```

```
x %>%
  deviation() %>%
  square() %>%
  mean() %>%
  sqrt()
```

## [1] 0.2798161

# 6.4 Lexical scoping

```
x <- 10
g01 <- function() {
   x <- 20
   x
}
g01()</pre>
```

## [1] 20

R's lexical scoping follows four primary rules:

- Name masking
- Functions versus variables
- A fresh start
- Dynamic lookup

# 6.4.1 Name masking

```
x <- 10
y <- 20
g02 <- function() {
    x <- 1
    y <- 2
    c(x, y)
}
g02()</pre>
```

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

```
x <- 2
g03 <- function() {
    y <- 1
    c(x, y)
}
g03()</pre>
```

## [1] 2 1

V

```
## [1] 20
```

```
x <- 1
g04 <- function() {
    y <- 2
    i <- function() {
        z <- 3
        c(x, y, z)
    }
    i()
}
g04()</pre>
```

## [1] 1 2 3

#### 6.4.2 Functions versus variables

```
g07 <- function(x) x + 1
g08 <- function() {
  g07 <- function(x) x + 100
  g07(10)
}
g08()</pre>
```

## [1] 110

```
g09 <- function(x) x + 100
g10 <- function() {
  g09 <- 10
  g09(g09)
}
g10()</pre>
```

## [1] 110

#### 6.4.3 A fresh start

```
g11 <- function() {
  if (!exists("a")) {
    a <- 1
  } else {
    a <- a + 1
  }
  a
}</pre>
```

##	-	Date
##	1	3
##	POSIXt	!
##	3	1
##	!.hexmode	!.octmode
##	2	2
##	!=	\$
##	1	1
##	\$.DLLInfo	\$.package_version
##	3	3
##	\$<-	\$ <data.frame< th=""></data.frame<>
##	1	4
##	- %%	
##	1	1
##	%/%	%in%
##	1	3
##	%o%	%x%
##	3	3
##	&	&&
##	1	1
##	&.hexmode	&.octmode
##	a.neximode	3.000 mode
##	(	*
##	1	1
##		elt
##	*.difftime	1
## ##		
##	length	names
##		1
	getNamespace	Hcbind
##		3
##	Hrbind	.amatch_bounds
##	3	2
##	.amatch_costs	.bincode
##	2	5
##	. C	.cache_class
##	1	1
##	.Call	.Call.graphics
##	1	1
##	.class2	.col
##	1	2
##	.colMeans	.colSums
##	5	5
##	.Date	.decode_numeric_version
##	3	2
##	.Defunct	.deparseOpts
##	4	2
##	.Deprecated	.detach
##	5	2
##	.difftime	.doSortWrap
##	4	5
##	.doTrace	.doWrap
##	3	5
##	.dynLibs	.encode_numeric_version
##	2	2

##	$.\mathtt{expand}_{\mathtt{R}}\mathtt{_{libs}}_{\mathtt{env}}\mathtt{_{var}}$	.External
##	2	1
##	.External.graphics	.External2
##	1	1
##	.First.sys	.fixupGFortranStderr
##	1	1
##	.fixupGFortranStdout	.format.zeros
##	1	6
##	.Fortran	.getNamespace
##	1	2
##	$. {\tt getNamespaceInfo}$	$. { t getRequiredPackages}$
##	3	5
##	.getRequiredPackages2	.gt
##	5	4
##	.gtn	.handleSimpleError
##	3	4
##	.Internal	.isMethodsDispatchOn
##	1 	1
##	.isOpen	.kappa_tri
##	2	6
##	.kronecker	.libPaths
##	6	3
##	.make_numeric_version	$.{\tt makeMessage}$
##	5	4
##	.mapply	$. {\tt maskedMsg}$
##	4	4
##	.mergeExportMethods	$. {\tt mergeImportMethods}$
##	3	4 N - W - H
##	.NotYetImplemented	.NotYetUsed
##	1	3
##	.OptRequireMethods	.packages
##	1	3
##	$.  exttt{packageStartupMessage}$	.POSIXct
##	3	4
##	.POSIX1t	.Primitive
##	4	1
##	.primTrace	.primUntrace
##	1	1
##	.rmpkg	.row
##	2	2
##	.row_names_info	.rowMeans
##	3	5
##	.rowNamesDF<-	.rowSums
##	4	5
##	.S3method	.Script
##	4	5
##	.set_row_names	.signalSimpleWarning
##	2	3
##	.standard_regexps	.subset
##	1	1
##	.subset2	.TAOCP1997init
##	1	2
##	.traceback	$. { t tryResumeInterrupt}$
##	3	1

```
##
                           .valid.factor
##
                                                                               1
                              /.difftime
##
##
                                        3
                                                                               1
##
                                       ::
##
                                        1
                                                                               1
##
                                                                             @<-
##
                                        1
                                                                               1
                                                                          [.AsIs
##
##
                                                                          [.Date
##
                            [.data.frame
##
                                        5
##
                              [.difftime
                                                                         [.Dlist
##
##
                           [.DLLInfoList
                                                                        [.factor
##
##
                               [.hexmode
                                                                        [.listof
##
                                        3
##
                               [.noquote
                                                             [.numeric_version
##
                               [.octmode
                                                                       [.POSIXct
##
##
##
                               [.POSIX1t
                                                                  [.simple.list
##
##
                                 [.table
                                                                     [.warnings
##
                                        6
##
                                       [[
                                                                  [[.data.frame
##
##
                                 [[.Date
                                                                       [[.factor
##
                                                                               3
                     [[.numeric_version
                                                                     [[.POSIXct
##
##
##
                              [[.POSIX1t
                                                                            [[<-
##
                        [[<-.data.frame
                                                                    [[<-.factor
##
##
                   [[<-.numeric_version
                                                                   [[<-.POSIX1t
##
##
                                        4
                                      [<-
                                                                 [<-.data.frame
##
##
                                        1
                                [<-.Date
                                                                   [<-.difftime
##
##
##
                              [<-.factor
                                                           [<-.numeric_version
##
                                                                               5
                             [<-.POSIXct
                                                                    [<-.POSIX1t
##
                                                                               5
##
                                                                               {
##
##
                                        1
                                                                               1
##
                                                                       |.hexmode
##
                                        1
##
                               |.octmode
                                                                              | |
                                        3
##
                                                                               1
##
                                                                               +
                                        1
##
                                                                               1
```

##	+.Date	+.POSIXt
##	3	3
##	<	<-
##	1	1
##	<<-	<=
##	1	1
##	=	==
##	1	1
##	>	>=
##	1	1
##	abbreviate	abs
##	8	1
##	acos	acosh
##	1	1
##	activeBindingFunction	addNA
##	3	3
##	addTaskCallback	agrep
##	4	9
##	agrepl	alist
##	8	2
##	all 1	all.equal 4
##	all.equal.character	all.equal.default
##	air.equar.character	arr.equar.deraurt 4
##	all.equal.environment	all.equal.envRefClass
##	arr.equar.environment	4
##	all.equal.factor	all.equal.formula
##	5	4
##	all.equal.function	all.equal.language
##	5	4
##	all.equal.list	all.equal.numeric
##	6	9
##	all.equal.POSIXt	all.equal.raw
##	7	5
##	all.names	all.vars
##	5	5
##	${\tt allowInterrupts}$	any
##	2	1
##	anyDuplicated	anyDuplicated.array
##	4	6
##	anyDuplicated.data.frame	${\tt anyDuplicated.default}$
##	5	5
##	anyDuplicated.matrix	anyNA
##	6	1
##	anyNA.data.frame	anyNA.numeric_version
##	3	3
##	anyNA.POSIX1t	aperm
##	3	aparm table
##	aperm.default	aperm.table
## ##	5 appoint	6
##	append 4	apply 6
##	Arg	array
##	1	4
πт	1	4

##	a massara d	0.0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## ##	arrayInd 5	as.array 3
##	as.array.default	as.call
##	3	1
##	as.character	as.character.condition
##	1	3
##	as.character.Date	as.character.default
##	3	3
##	as.character.error	as.character.factor
##	3	3
##	as.character.hexmode	as.character.numeric_version
##	3	3
##	as.character.octmode	as.character.POSIXt
##	3	3
##	as.character.srcref	as.complex
##	5	1
## ##	as.data.frame 5	as.data.frame.array 5
##	as.data.frame.AsIs	as.data.frame.character
##	5	4
##	as.data.frame.complex	as.data.frame.data.frame
##	6	4
##	as.data.frame.Date	as.data.frame.default
##	6	3
##	as.data.frame.difftime	as.data.frame.factor
##	6	6
##	as.data.frame.integer	as.data.frame.list
##	6	10
##	as.data.frame.logical	as.data.frame.matrix
##	6	7
##	as.data.frame.model.matrix	as.data.frame.noquote
## ##	as.data.frame.numeric	as.data.frame.numeric_version
##	as.uata.irame.numeric	as.data.frame.humeric_version
##	as.data.frame.ordered	as.data.frame.POSIXct
##	6	6
##	as.data.frame.POSIXlt	as.data.frame.raw
##	5	6
##	as.data.frame.table	as.data.frame.ts
##	8	3
##	as.data.frame.vector	as.Date
##	6	3
##	as.Date.character	as.Date.default
##	6	3
##	as.Date.factor	as.Date.numeric
##	3	4
## ##	as.Date.POSIXct	as.Date.POSIX1t 3
##	as.difftime	as.double
##	as.dilitime	as.double
##	as.double.difftime	as.double.POSIX1t
##	4	3
##	as.environment	as.expression
##	1	3

##	as.expression.default	as.factor
##	3	2
##	as.function	as.function.default
##	3	4
##	as.hexmode	as.integer
##	2	1
##	as.list 3	as.list.data.frame
##		3
## ##	as.list.Date	as.list.default 3
##	as.list.difftime	as.list.environment
##	3	as.list.envilonment
##	as.list.factor	as.list.function
##	3	3
##	as.list.numeric_version	as.list.POSIXct
##	3	3
##	as.list.POSIXlt	as.logical
##	3	1
##	as.logical.factor	as.matrix
##	3	3
##	as.matrix.data.frame	as.matrix.default
##	4	3
##	as.matrix.noquote	as.matrix.POSIX1t
##	3	3
##	as.name	as.null
##	2	3
##	as.null.default	as.numeric
##	3	1
##	as.numeric_version	as.octmode
##	2	2
##	as.ordered	as.package_version
##	2	2
##	as.pairlist	as.POSIXct
##	2	4
##	as.POSIXct.Date	as.POSIXct.default
##	3	A DOGIN - DOGINA
##	as.POSIXct.numeric	as.POSIXct.POSIXlt
##	5	4
## ##	as.POSIX1t 4	as.POSIX1t.character
##	as.POSIX1t.Date	as.POSIXlt.default
##	as.rusixit.bate	as.rusixit.delauit 5
##	as.POSIXlt.factor	as.POSIXlt.numeric
##	as.Fusikit.iactor	as.robixit.numeric
##	as.POSIX1t.POSIXct	as.qr
##	4	2
##	as.raw	as.single
##	1	3
##	as.single.default	as.symbol
##	3	2
##	as.table	as.table.default
##	3	3
##	as.vector	as.vector.factor
##	3	3
	_	_

##	asin	asinh
##	1	1
##	asNamespace	asplit
##	3	3
##	asS3	asS4
##	4	4
##	assign	atan
##	7	1
##	atan2	atanh
##	3	1
##	attach	attachNamespace
##	5	6
##	attr	attr.all.equal
##	1	6
## ##	attr<-	attributes
##	1 attributes<-	1 autoload
##	attributes 1	5 auto10au
##	autoloader	backsolve
##	4	backsolve 6
##	baseenv	basename
##	1	2
##	besselI	besselJ
##	4	3
##	besselK	besselY
##	4	3
##	beta	bindingIsActive
##	3	3
##	bindingIsLocked	bindtextdomain
##	3	3
##	bitwAnd	bitwNot
##	3	2
##	bitw0r	${ t bitwShiftL}$
##	3	3
##	bitwShiftR	bitwXor
##	3	3
##	body	body<-
##	2	4
##	bquote	break
##	4	1
##	browser	browserCondition
##	1	2
##	browserSetDebug	browserText
## ##	2	2
##	builtins 2	bу 6
##		
##	by.data.frame 6	by.default 6
##	bzfile	c
##	5	1
##	c.Date	c.difftime
##	3	3
##	c.factor	c.noquote
##	3	3
	· ·	0

##	<pre>c.numeric_version</pre>	c.POSIXct
##	3	3
##	c.POSIX1t	c.warnings
## ##	3	3
##	call 1	callCC 2
##	capabilities	casefold
##	3	3
##	cat	cbind
##	7	3
##	cbind.data.frame	ceiling
##	3	1
##	char.expand	character
##	4	2
##	charmatch	charToRaw
##	4	2
##	chartr	check_tzones
##	4 -h-D-+-	2
## ##	chkDots 4	chol 3
##	chol.default	chol2inv
##	6	4
##	choose	class
##	3	1
##	class<-	clearPushBack
##	1	2
##	close	close.connection
##	3	4
##	close.srcfile	close.srcfilealias
##	3	3
##	closeAllConnections	col
##	1	3
##	colMeans	colnames
## ##	4	4
##	colnames<-	colSums 4
##	commandArgs	comment
##	2	2
##	comment<-	complex
##	3	6
##	computeRestarts	conditionCall
##	2	2
##	conditionCall.condition	${\tt condition} {\tt Message}$
##	2	2
##	${\tt condition Message.condition}$	conflictRules
##	2	4
##	conflicts	Conj
##	3	1
##	contributors	cos
##	1	1
## ##	cosh	cospi
## ##	1 crossprod	1 Cstack_info
##	crossprod 3	cstack_info
##	3	1

##	cummax	cummin
##	1	1
##	cumprod	cumsum
##	1	1
##	curlGetHeaders	cut
##	6	3
## ##	cut.Date 7	cut.default 9
##	cut.POSIXt	data.class
##	7	data.class
##	data.frame	data.matrix
##	7	3
##	date	debug
##	1	5
##	debuggingState	debugonce
##	2	5
##	default.stringsAsFactors	delayedAssign
##	1	5
##	deparse	deparse1
##	6	5
##	det	detach
##	3	6
##	determinant	determinant.matrix
##	4	4
##	dget	diag
##	3	5
##	diag<-	diff
##	3	3
## ##	diff.Date 5	diff.default 5
##	diff.difftime	diff.POSIXt
##	3	5
##	difftime	digamma
##	5	1
##	dim	dim.data.frame
##	1	2
##	dim<-	dimnames
##	1	1
##	dimnames.data.frame	dimnames<-
##	2	1
##	dimnames <data.frame< th=""><th>dir</th></data.frame<>	dir
##	3	9
##	dir.create	dir.exists
##	5	2
##	dirname	do.call
##	2	5
##	dontCheck	double
##	2	2
##	dput	dQuote
##	4	dronlovels
##	drop 2	droplevels 3
##	droplevels.data.frame	droplevels.factor
##	dropieveis.data.irame 5	dropieveis.lactor
πĦ	5	4

##	dump	duplicated
##	7	4
## ##	duplicated.array 6	duplicated.data.frame 5
##	duplicated.default	duplicated.matrix
##	duplicated.deladit	dupilcated.matrix
##	duplicated.numeric_version	duplicated.POSIX1t
##	4	4
##	duplicated.warnings	dyn.load
##	4	5
##	dyn.unload	dynGet
##	2	5
##	eapply	eigen
##	6	5
##	emptyenv	enc2native
##	1	1
## ##	enc2utf8 1	encodeString
##	Encoding	6 Encoding<-
##	incouring 2	3
##	endsWith	enquote
##	3	2
##	env.profile	environment
##	2	2
##	environment<-	${\tt environmentIsLocked}$
##	1	2
##	environmentName	errorCondition
##	2	5
##	eval	eval.parent
##	4	3
## ##	evalq 4	exists 7
##	exp	expand.grid
##	1	4
##	expm1	expression
##	1	1
##	extSoftVersion	factor
##	1	7
##	factorial	fifo
##	2	5
##	file	file.access
##	7	3
##	file.append	file.choose
##	3	2
##	file.copy	file.create
## ##	7 file.exists	3 file.info
##	2	3
##	file.link	file.mode
##	3	2
##	file.mtime	file.path
##	2	3
##	file.remove	file.rename
##	2	3

```
##
                              file.show
                                                                   file.size
##
##
                          file.symlink
                                                                      Filter
##
                                   Find
                                                                find.package
##
                                      5
                          findInterval
                                                             findPackageEnv
##
##
                           findRestart
                                                                       floor
##
                                      3
                                  flush
                                                           flush.connection
##
                                    for
                                                                       force
##
##
                          forceAndCall
                                                                     formals
##
##
                             formals<-
                                                                      format
##
                           format.AsIs
                                                          format.data.frame
##
##
                           format.Date
                                                             format.default
##
##
##
                       format.difftime
                                                              format.factor
                        format.hexmode
                                                                 format.info
##
##
                     format.libraryIQR
                                                     format.numeric_version
##
                        format.octmode
                                                         format.packageInfo
                        format.POSIXct
                                                              format.POSIX1t
##
##
##
                           format.pval
                                                      format.summaryDefault
##
                                formatC
                                                                    formatDL
##
##
##
                          forwardsolve
                                                                    function
##
                                      6
                                                                           1
##
                                  gamma
                                                                          gc
##
                                      1
                                                                           4
                               gc.time
                                                                      gcinfo
##
##
                                                                  gctorture2
                              gctorture
##
                                                                        get0
                                    get
##
                                                               getCallingDLL
                     getAllConnections
##
##
                        getCallingDLLe
                                                              getConnection
##
             getDLLRegisteredRoutines getDLLRegisteredRoutines.character
##
##
     getDLLRegisteredRoutines.DLLInfo
##
                                                                  getElement
##
```

${\tt getExportedVal}$	5 5	## ##
getLoadedDL		##
S	_	##
${\tt getNamespaceExpor}$	getNamespace	##
	‡ 2	##
${ t getNamespaceIn}$	8	##
		##
${ t getNamespaceUse}$	8	##
mo+No+irroCrypholIn		##
getNativeSymbolIn	8	## ##
getRversi		##
gotitvoisi		##
${ t getTaskCallbackNam}$		##
C		##
gettex	# gettext	##
	‡ 3	##
	8	##
		##
globale	0 0	##
		##
gregex	8 18 1	## ##
gre		##
gre	8 1	##
groupi	·	##
		##
gzo	‡ gsub	##
	<b>#</b> 8	##
	8	##
		##
iconvli		##
÷ Q -+ Q - 1 1 -		##
icuSetColla		## ##
identi	<del>-</del>	##
1401101		##
ifel		##
		##
importIntoE	‡ Im	##
	‡ 1	##
inheri		##
		##
interacti		##
		##
interse		##
in+Toll+		## ##
intToUt		##
invisib		##
111/1010		##
${\tt invokeRestartInteractive}$		##
	<b>‡</b> 3	##

##	is.array	is.atomic
##	1	ia sharatan
## ##	is.call	is.character 1
##	is.complex	is.data.frame
##	1	2
##	is.double	is.element
##	1	3
##	is.environment	is.expression
##	1	1
##	is.factor	is.finite
##	2	1
## ##	is.function 1	is.infinite 1
##	is.integer	is.language
##	1s.integer	15.1anguage
##	is.list	is.loaded
##	1	4
##	is.logical	is.matrix
##	1	1
##	is.na	is.na.data.frame
##	1	2
##	is.na.numeric_version	is.na.POSIX1t
##	2	2
##	is.na<-	is.na <default< th=""></default<>
## ##	3 is.na <factor< th=""><th>is not numeric version</th></factor<>	is not numeric version
##	15.11a\\ .11aCto1	is.na <numeric_version 3<="" th=""></numeric_version>
##	is.name	is.nan
##	1	1
##	is.null	is.numeric
##	1	1
##	is.numeric.Date	is.numeric.difftime
##	2	2
##	is.numeric.POSIXt	is.numeric_version
##	2	2
##	is.object	is.ordered
## ##	is.package_version	ig pairligh
##	is.package_version 2	is.pairlist 1
##	is.primitive	is.qr
##	2	2
##	is.R	is.raw
##	1	1
##	is.recursive	is.single
##	1	1
##	is.symbol	is.table
##	1	2
##	is.unsorted	is.vector
##	4	3
## ##	isa 3	isatty 2
##	isBaseNamespace	isdebugged
##	13Dasenamespace 2	15debugged 3
	2	ŭ

##	isFALSE	${\tt isIncomplete}$
##	2	2
## ##	isNamespace 2	isNamespaceLoaded 2
##	ISOdate	ISOdatetime
##	8	8
##	isOpen	isRestart
##	3	2
##	isS4	isSeekable
##	1	2
##	isSymmetric	<pre>isSymmetric.matrix</pre>
##	3	5
##	isTRUE	jitter
##	2	4
##	julian	julian.Date
##	3	4
## ##	julian.POSIXt 4	kappa 3
##	kappa.default	kappa.lm
##	kappa.dordari	3
##	kappa.qr	kronecker
##	3	6
##	l10n_info	La.svd
##	1	4
##	La_library	La_version
##	1	1
##	labels	labels.default
##	3	3
##	lapply	lazyLoad
## ##	4	4
##	lazyLoadDBexec 4	lazyLoadDBfetch 1
##	lbeta	lchoose
##	3	3
##	length	length.POSIX1t
##	1	2
##	length<-	length <date< th=""></date<>
##	1	3
##	<pre>length<difftime< pre=""></difftime<></pre>	length <factor< th=""></factor<>
##	3	3
##	length <posixct< th=""><th>length<posix1t< th=""></posix1t<></th></posixct<>	length <posix1t< th=""></posix1t<>
##	3	3
##	lengths	levels
## ##	3 levels.default	2 levels<-
##	revers.deraurt 2	1
##	levels <factor< th=""><th>lfactorial</th></factor<>	lfactorial
##	3	2
##	lgamma	libcurlVersion
##	1	1
##	library	library.dynam
##	14	7
##	library.dynam.unload	licence
##	5	1

##	licongo	list
##	license 1	1
##	list.dirs	list.files
##	4	9
##	list2DF	list2env
##	3	6
##	load	loadedNamespaces
##	4	1
##	${\tt loadingNamespaceInfo}$	loadNamespace
##	1	7
##	local	lockBinding
##	3	3
##	lockEnvironment	log
##	3	1
##	log10	log1p
##	1	1
##	log2	logb
##	1	3
##	logical	lower.tri
##	2	3
##	ls 7	make.names 4
## ##	make.unique	
##	make.unique 3	makeActiveBinding 4
##	Map	mapply
##	3	шаррту 6
##	margin.table	marginSums
##	3	3
##	mat.or.vec	match
##	3	5
##	match.arg	match.call
##	4	5
##	match.fun	Math.data.frame
##	3	3
##	Math.Date	Math.difftime
##	3	3
##	Math.factor	Math.POSIXt
##	3	3
##	matrix	max
##	6	1
##	max.col	mean
##	3	3
## ##	mean.Date	mean.default 5
##	mean.difftime	mean.POSIXct
##	mean.dilitime	mean.i obiket
##	mean.POSIX1t	mem.maxNSize
##	mean.robikit	mem.maxN512e
##	mem.maxVSize	memCompress
##	2	3
##	memDecompress	memory.profile
##	4	1
##	merge	merge.data.frame
##	4	14

##	merge.default	message
##	4	4
##	mget	min
##	6	1
##	missing	Mod
##	1	1
##	mode	mode<-
##	2	3
##	months	months.Date
##	3	3
##	months.POSIXt	mostattributes<-
##	3	3
##	names	names.POSIX1t
##	1	2
##	names<-	names <posix1t< td=""></posix1t<>
##	1	3
##	namespaceExport	${\tt namespaceImport}$
##	3	5
##	${\tt namespaceImportClasses}$	
		namespaceImportFrom
##	5	8
##	${\tt namespaceImportMethods}$	nargs
##	5	1
##	nchar	ncol
##	5	2
##	NCOL	Negate
##	2	2
##	new.env	next
##	4	1
##	NextMethod	ngettext
##	4	5
##	nlevels	noquote
##	2	3
##		${\tt normalizePath}$
	norm	
##	3	4
##	nrow	NROW
##	2	2
##	nullfile	numeric
##	1	2
##	numeric_version	numToBits
##	3	2
##	numToInts	nzchar
##	2	1
##	objects	oldClass
##	7	1
##	oldClass<-	OlsonNames
##	1	2
##	on.exit	
		open
##	1	3
##	open.connection	open.srcfile
##	5	4
##	open.srcfilealias	open.srcfilecopy
##	4	4
##	Ops.data.frame	Ops.Date
##	3	3

##	Ops.difftime	Ops.factor
##	3	3
##	Ops.numeric_version 3	Ops.ordered
## ##	Ops.POSIXt	3 options
##	3	options 2
##	order	ordered
##	5	3
##	outer	package_version
##	5	3
##	packageEvent	${\tt packageHasNamespace}$
##	3	3
##	packageNotFoundError	packageStartupMessage
##	4	4
##	packBits	pairlist
## ##	3	2
##	parent.env 2	parent.env<- 3
##	parent.frame	parse
##	2	8
##	parseNamespaceFile	paste
##	4	5
##	paste0	path.expand
##	4	2
##	path.package	pcre_config
##	3	_ 1
##	pipe	plot
## ##	4	4
##	pmatch 5	pmax 3
##	pmax.int	pmin
##	3	3
##	pmin.int	polyroot
##	3	2
##	pos.to.env	pretty
##	1	3
##	pretty.default	prettyNum
##	9	14
##	print	print.AsIs
##	3	3
## ##	print.by 4	print.condition 3
##	print.connection	print.data.frame
##	print.connection 3	princ.data.rrame
##	print.Date	print.default
##	4	11
##	<pre>print.difftime</pre>	print.Dlist
##	4	3
##	print.DLLInfo	<pre>print.DLLInfoList</pre>
##	3	3
##	print.DLLRegisteredRoutines	print.eigen
##	3	3
##	print.factor	print.function
##	6	4

##	<pre>print.hexmode</pre>	print.libraryIQR
##	3	3
##	print.listof	print.NativeRoutineList
##	3	3
## ##	print.noquote 5	print.numeric_version 4
##	print.octmode	print.packageInfo
##	3	3
##	print.POSIXct	print.POSIX1t
##	6	6
##	<pre>print.proc_time</pre>	print.restart
##	3	3
##	<pre>print.rle</pre>	<pre>print.simple.list</pre>
##	5	3
##	print.srcfile	print.srcref
##	3	4
##	print.summary.table	print.summary.warnings
##	4	3
## ##	print.summaryDefault 4	print.table 9
##	print.warnings	prmatrix
##	print.warnings	8
##	proc.time	prod
##	1	1
##	prop.table	proportions
##	3	3
##	provideDimnames	psigamma
##	5	3
##	pushBack	pushBackLength
##	5	2
##	q	qr
##	4	3
##	qr.coef	qr.default
## ##	3	5
##	qr.fitted 4	qr.Q 4
##	qr.qty	qr.qy
##	3	3
##	qr.R	qr.resid
##	3	3
##	qr.solve	qr.X
##	4	4
##	quarters	quarters.Date
##	3	3
##	quarters.POSIXt	quit
##	3	4
##	quote	R.home
##	1 P. Vi	2
##	R.Version	R_system_version
## ##	1	rango default
## ##	range 1	range.default 4
##	rank	rapply
##	4	7
	-	•

##	raw	rawConnection
##	2	3
##	${\tt rawConnectionValue}$	rawShift
##	2	3
##	rawToBits	rawToChar
##	2	3
##	rbind	rbind.data.frame
##	3	6
##	rcond	Re
##	5	1
## ##	read.dcf 5	readBin 7
##	readChar	readline
##	1 eadonaí 4	
##	readLines	readRDS
##	7	3
##	readRenviron	Recall
##	2	2
##	Reduce	reg.finalizer
##	6	4
##	regexec	regexpr
##	7	7
##	registerS3method	registerS3methods
##	5	4
##	regmatches	regmatches<-
##	4	5
##	remove	removeTaskCallback
## ##	6	2
##	rep 1	rep.Date 3
##	rep.difftime	rep.factor
##	3	3
##	rep.int	rep.numeric_version
##	3	3
##	rep.POSIXct	rep.POSIX1t
##	3	3
##	rep_len	repeat
##	3	1
##	replace	replicate
##	4	4
##	require	requireNamespace
##	10	4
## ##	restartDescription 2	restartFormals 2
##	retracemem	return
##	retracemem 1	1
##	returnValue	rev
##	2	2
##	rev.default	rle
##	2	2
##	rm	RNGkind
##	6	4
##	RNGversion	round
##	2	1

шш		1 DOGTV+
##	round.Date	round.POSIXt
##	3	3
##	row	row.names
##	3	2
##	row.names.data.frame	row.names.default
##	2	2
##	row.names<-	row.names <data.frame< th=""></data.frame<>
##	3	3
##	row.names <default< th=""><th>rowMeans</th></default<>	rowMeans
##	3	4
##	rownames	rownames<-
##	4	3
##	rowsum	rowsum.data.frame
##	5	6
##	rowsum.default	rowSums
##	6	4
##	sample	sample.int
##	5	6
##	sapply	save
##	6	11
##	save.image	saveRDS
##	6	7
##	scale	scale.default
##	4	4
##	scan	search
##	23	1
##	searchpaths	seek
##	1	3
##	seek.connection	seq
##	6	2
##	seq.Date	seq.default
##	7	7
##	seq.int	seq.POSIXt
##	1	7
##	seq_along	seq_len
##	1	1
##	sequence	sequence.default
##	3	5
##	serialize	serverSocket
##	7	2
##	set.seed	setdiff
##	5	4
##	setequal	setHook
##	4	4
##	${\tt setNamespaceInfo}$	$\mathtt{setSessionTimeLimit}$
##	4	3
##	setTimeLimit	setwd
##	4	2
##	shell	shell.exec
##	9	2
##	showConnections	shQuote
##	2	3
##	sign	signalCondition
##	1	2
π#	1	2

##	signif	simpleCondition
##	1	3
## ##	simpleError 3	simpleMessage 3
##	simpleWarning	simplify2array
##	3	3 simpiliyzariay
##	sin	single
##	1	2
##	sinh	sink
##	1	5
##	sink.number	sinpi
##	2	1
##	slice.index	socketAccept
##	3	7
##	socketConnection	socketSelect
##	9	4
## ##	socketTimeout 3	solve 4
##	solve.default	solve.qr
##	Solve derault	301ve.qr 4
##	sort	sort.default
##	4	5
##	sort.int	sort.list
##	7	6
##	sort.POSIX1t	source
##	5	17
##	split	split.data.frame
##	5	5
##	split.Date	split.default
##	5	7
## ##	split.POSIXct 5	split<- 6
##	split <data.frame< td=""><td>split<default< td=""></default<></td></data.frame<>	split <default< td=""></default<>
##	Spiit .data.iiame	Spiit delaut
##	sprintf	sqrt
##	3	1
##	sQuote	srcfile
##	3	4
##	srcfilealias	srcfilecopy
##	3	5
##	srcref	standardGeneric
##	3	1
##	startsWith	stderr
##	3	1
## ##	stdin 1	stdout
##		1
##	stop 4	stopifnot 5
##	storage.mode	storage.mode<-
##	2	1
##	str2expression	str2lang
##	2	2
##	strftime	strptime
##	6	4

##	strrep	strsplit
##	3	6
##	strtoi	strtrim
##	3	3
##	structure	strwrap
##	3	8
##	sub	subset
##	8	3
##	subset.data.frame 6	subset.default
## ##	subset.matrix	4 substitute
##	Subset.matrix	substitute 1
##	substr	substr<-
##	4	5
##	substring	substring<-
##	4	5
##	sum	summary
##	1	3
##	summary.connection	summary.data.frame
##	3	5
##	Summary.data.frame	summary.Date
##	3	4
##	Summary.Date	summary.default
##	3	5
##	Summary.difftime	summary.factor
##	3	4
##	Summary.factor	summary.matrix
##	3	3
##	Summary.numeric_version	Summary.ordered
##	3	3
##	summary.POSIXct	Summary.POSIXct
## ##	4	Summany DOSTVI+
##	summary.POSIX1t 4	Summary.POSIX1t 3
##	summary.proc_time	summary.srcfile
##	3	3
##	summary.srcref	summary.table
##	4	3
##	summary.warnings	suppressMessages
##	3	3
##	suppressPackageStartupMessages	suppressWarnings
##	2	3
##	${\tt suspendInterrupts}$	svd
##	2	5
##	sweep	switch
##	7	1
##	sys.call	sys.calls
##	2	1
##	Sys.chmod	Sys.Date
##	4	1
##	sys.frame	sys.frames
## ##	2	1
## ##	sys.function 2	Sys.getenv
##	2	4

##	Sys.getlocale	Sys.getpid
##	2	1
##	Sys.glob	Sys.info
##	3	1
##	Sys.junction	sys.load.image
## ##	3 Sys.localeconv	3
##	Sys. Totaleconv	sys.nframe 1
##	sys.on.exit	sys.parent
##	1	2
##	sys.parents	Sys.readlink
##	1	2
##	sys.save.image	Sys.setenv
##	2	2
##	${\tt Sys.setFileTime}$	Sys.setlocale
##	3	3
##	Sys.sleep	sys.source
##	2	7
##	sys.status	Sys.time
## ##	1	1
##	Sys.timezone 2	Sys.umask 2
##	Sys.unsetenv	Sys.which
##	by s. unsevenv	2 2
##	system	system.file
##	11	5
##	system.time	system2
##	3	12
##	t	t.data.frame
##	2	2
##	t.default	table
##	2	6
##	tabulate	tan
##	3	1
##	tanh	tanpi
## ##	1	1
##	tapply 7	taskCallbackManager 4
##	tcrossprod	tempdir
##	3	2
##	tempfile	textConnection
##	4	6
##	textConnectionValue	tolower
##	2	2
##	topenv	toString
##	3	3
##	toString.default	toupper
##	4	2
##	trace	traceback
##	9	3
## ##	tracemem 1	tracingState 2
##	transform	transform.data.frame
##	3	3
ııπ	3	5

##	transform.default	trigamma
##	3	1
##	trimws	trunc
##	4	1
##	trunc.Date	trunc.POSIXt
##	3	4
##	truncate	truncate.connection
##	3	3
##	try	tryCatch
##	4	4
##	tryInvokeRestart	typeof
##	3	2
##	unclass	undebug
##	1	3
##	union	unique
## ##	4	4
##	unique.array 6	unique.data.frame 5
##	unique.default	unique.matrix
##	unique.deraurt	dilique.matrix
##	unique.numeric_version	unique.POSIX1t
##	unique.numeric_version 4	unique.1051x10
##	unique.warnings	units
##	unique: warnings	2
##	units.difftime	units<-
##	2	3
##	units <difftime< td=""><td>unix.time</td></difftime<>	unix.time
##	3	2
##	unlink	unlist
##	5	4
##	${\tt unloadNamespace}$	unlockBinding
##	2	3
##	unname	unserialize
##	3	3
##	unsplit	untrace
##	4	4
##	untracemem	unz
##	1	5
##	upper.tri	url
##	3	7
##	UseMethod	utf8ToInt
##	1	2
##	validEnc	validUTF8
##	2	2
##	vapply	vector
##	6	3
##	Vectorize	warning
##	5	6
##	warningCondition	warnings
##	5	2
##	weekdays	weekdays.Date
##	3	3
##	weekdays.POSIXt	which
##	3	4

##	which.max	which.min	
##	2	2	
##	while	with	
##	1	4	
##	with.default	${\tt withAutoprint}$	
##	4	10	
##	withCallingHandlers	within	
##	3	4	
##	within.data.frame	within.list	
##	4	5	
##	${\tt withRestarts}$	withVisible	
##	3	2	
##	write	write.dcf	
##	6	8	
##	writeBin	writeChar	
##	6	6	
##	writeLines	xor	
##	5	3	
##	xpdrows.data.frame	xtfrm	
##	4	1	
##	xtfrm.AsIs	xtfrm.data.frame	
##	2	2	
##	xtfrm.Date	xtfrm.default	
##	2	2	
##	xtfrm.difftime	xtfrm.factor	
##	2	2	
##	xtfrm.numeric_version	xtfrm.POSIXct	
##	2	2	
##	xtfrm.POSIX1t	xzfile	
##	2	5	
##	zapsmall		
##	3		
g11()			
##		Date	
##	1	3	
##	POSIXt	!	
##	3	1	
##	!.hexmode	!.octmode	
##	2	2	
##	!=	\$	
##	1	1	
##	\$.DLLInfo	\$.package_version	
##	3	3	
##	\$<-	\$ <data.frame< td=""><td></td></data.frame<>	
##	1	4	
##	<b>%%</b>	% <b>*</b> %	
##	1	1	
##	%/%	%in%	
##	1	3	
##	%0%	%x%	
##	3	3	
##	&	&&	

1	1	##
&.octmode	&.hexmode	##
3	3	##
*	(	##
1	1	##
elt	*.difftime	##
1	3	##
names	length	##
1		##
Hcbind	5 1	##
3		##
.amatch_bounds	<b></b>	##
2		##
.bincode	<del>-</del>	##
5		##
.cache_class		##
1		##
.Call.graphics		##
1		##
.col		##
2		##
.colSums		##
5		##
.decode_numeric_version		##
denoracionta		## ##
.deparseOpts 2		##
.detach		##
. detach	1	##
.doSortWrap		##
5		##
.doWrap		##
5		##
.encode_numeric_version		##
2		##
.External		##
1	_	##
.External2		##
1		##
$. { t fixup GFortran Stdern}$	.First.sys	##
1		##
.format.zeros	.fixupGFortranStdout	##
6		##
$. \verb"getNamespace"$	.Fortran	##
2	1	##
$. { t getRequiredPackages}$	$. {\tt getNamespaceInfo}$	##
5	3	##
.gt	$. {\tt getRequiredPackages2}$	##
4	5	##
$. {\tt handleSimpleError}$	.gtn	##
4		##
$. { t isMethodsDispatchOn}$		##
1		##
.kappa_tri	.isOpen	##

##	2	6
##	.kronecker	.libPaths
##	6	3
##	.make_numeric_version	.makeMessage
##	5	4
##	.mapply	$. {\tt maskedMsg}$
##	4	4
##	$. {\tt mergeExportMethods}$	$.{\tt mergeImportMethods}$
##	3	4
##	.NotYetImplemented	.NotYetUsed
##	1	3
##	.OptRequireMethods	.packages
##	1	3
##	$. {\tt packageStartupMessage}$	.POSIXct
##	3	4
##	.POSIX1t	.Primitive
##	4	1
##	.primTrace	.primUntrace
##	1	1
##	.rmpkg	.row
##	2	2
##	.row_names_info	rowMeans.
##	3	5
##	.rowNamesDF<-	.rowSums
##	4	5
##	.S3method	.Script
##	4	5
##	.set_row_names	$. { t signal Simple Warning}$
##	2	3
##	.standard_regexps	.subset
##	1	1
##	.subset2	.TAOCP1997init
##	1	2
##	.traceback	$. { t tryResumeInterrupt}$
##	3	. orynesumernoerrupo
##	.valid.factor	/
##	2	1
##	/.difftime	:
##	3	
##	::	:::
##	1	
##	0	0<-
##	1	1
##	Ţ	[.AsIs
##	1	4
##	[.data.frame	[.Date
##	j.data.irame 5	4. Date
##	c.difftime	[.Dlist
##	L.dilitime 4	(.DIISt 4
##		[.factor
	[.DLLInfoList	
## ##	3 [.hexmode	4 [.listof
	n	Λ
## ##	3 [.noquote	4 [.numeric_version

```
##
##
                               [.octmode
                                                                       [.POSIXct
##
##
                               [.POSIX1t
                                                                  [.simple.list
                                        5
##
                                 [.table
##
                                                                     [.warnings
##
                                        6
                                       [[
                                                                  [[.data.frame
##
##
                                        1
                                 [[.Date
##
                                                                       [[.factor
##
                                                                               3
##
                     [[.numeric_version
                                                                     [[.POSIXct
##
                              [[.POSIX1t
                                                                            [[<-
##
##
                                                                               1
                        [[<-.data.frame
                                                                    [[<-.factor
##
##
                   [[<-.numeric_version
                                                                   [[<-.POSIX1t
##
##
                                      [<-
                                                                 [<-.data.frame
##
##
                                        1
                                                                   [<-.difftime
##
                                [<-.Date
##
                                                           [<-.numeric_version
                              [<-.factor
##
##
                             [<-.POSIXct
                                                                    [<-.POSIX1t
##
##
                                                                               5
                                                                               {
##
                                        1
                                                                               1
##
                                                                       |.hexmode
##
                                        1
                                                                               3
##
                               |.octmode
                                                                              \Pi
##
                                        3
                                                                               1
##
                                        1
##
##
                                  +.Date
                                                                       +.POSIXt
##
                                                                               3
##
                                        <
                                        1
                                                                               1
##
##
##
                                        1
##
                                                                               1
##
                                                                              >=
                                        1
                                                                               1
##
                              abbreviate
                                                                             abs
##
                                        8
                                                                               1
##
                                     acos
                                                                           acosh
##
                                                                               1
##
                  activeBindingFunction
                                                                           addNA
##
                                                                               3
                        addTaskCallback
##
                                                                           agrep
##
                                                                               9
##
                                  agrepl
                                                                           alist
```

##	8	2
##	all	all.equal
##	1	4
##	all.equal.character	all.equal.default
##	5	4
##	all.equal.environment	all.equal.envRefClass
##	6	4
##	all.equal.factor	all.equal.formula
## ##	5 all.equal.function	4
##	arr.equar.runctron 5	all.equal.language 4
##	all.equal.list	all.equal.numeric
##	6	9
##	all.equal.POSIXt	all.equal.raw
##	7	5
##	all.names	all.vars
##	5	5
##	allowInterrupts	any
##	2	1
##	anyDuplicated	anyDuplicated.array
## ##	4 anyDuplicated.data.frame	6 anyDuplicated.default
##	anyDupiicated.data.irame	anybupiicated.derauit
##	anyDuplicated.matrix	anyNA
##	6	1
##	anyNA.data.frame	anyNA.numeric_version
##	3	3
##	${ t any {\tt NA.POSIXlt}}$	aperm
##	3	4
##	aperm.default	aperm.table
##	5	6
##	append 4	apply 6
## ##		o array
##	Arg 1	4
##	arrayInd	as.array
##	5	3
##	as.array.default	as.call
##	3	1
##	as.character	as.character.condition
##	1	3
##	as.character.Date	as.character.default
## ##	as.character.error	3 as.character.factor
##	as.character.error	as.character.ractor
##	as.character.hexmode	as.character.numeric_version
##	3	3
##	as.character.octmode	as.character.POSIXt
##	3	3
##	as.character.srcref	as.complex
##	5	1
##	as.data.frame	as.data.frame.array
##	5	5
##	as.data.frame.AsIs	as.data.frame.character

##	5	4
##	as.data.frame.complex	as.data.frame.data.frame
##	6	4
##	as.data.frame.Date	as.data.frame.default
##	6	3
##	as.data.frame.difftime	as.data.frame.factor
##	6	6
##	as.data.frame.integer	as.data.frame.list
##	6	10
##	as.data.frame.logical	as.data.frame.matrix
##	6	7
##	as.data.frame.model.matrix	as.data.frame.noquote
##	6	6
##	as.data.frame.numeric	as.data.frame.numeric_version
##	6	6
##	as.data.frame.ordered	as.data.frame.POSIXct
##	6	6
##	as.data.frame.POSIX1t	as.data.frame.raw
##	5	6
##	as.data.frame.table 8	as.data.frame.ts
## ##	as.data.frame.vector	3 as.Date
##	as.data.frame.vector	as.Date
##	as.Date.character	as.Date.default
##	as.bate.character	as.bate.deraurt
##	as.Date.factor	as.Date.numeric
##	3	4
##	as.Date.POSIXct	as.Date.POSIXlt
##	4	3
##	as.difftime	as.double
##	5	1
##	as.double.difftime	as.double.POSIX1t
##	4	3
##	as.environment	as.expression
##	1	3
##	as.expression.default	as.factor
##	3	2
##	as.function	as.function.default
##	3	4
##	as.hexmode	as.integer
##	2	1
##	as.list	as.list.data.frame
##	3	3
## ##	as.list.Date	as.list.default 3
##	as.list.difftime	
##	as.list.difftime 3	as.list.environment 5
##	as.list.factor	as.list.function
##	as.list.lactor	as.list.lunction
##	as.list.numeric_version	as.list.POSIXct
##	as.list.numeric_version	3
##	as.list.POSIXlt	as.logical
##	3	1
##	as.logical.factor	as.matrix
	0	

##	3	3
##	as.matrix.data.frame	as.matrix.default
##	4	3
##	as.matrix.noquote	as.matrix.POSIX1t
##	3	3
##	as.name	as.null
##	2	3
##	as.null.default	as.numeric
##	3	1
##	as.numeric_version	as.octmode
##	2	2
##	as.ordered	as.package_version
##	2	2
##	as.pairlist	as.POSIXct
##	2	4
##	as.POSIXct.Date	as.POSIXct.default
##	3	4
##	as.POSIXct.numeric	as.POSIXct.POSIXlt
##	5	4
##	as.POSIX1t	as.POSIXlt.character
##	4	7
##	as.POSIXlt.Date	as.POSIXlt.default
##	3	5
##	as.POSIX1t.factor	as.POSIXlt.numeric
##	3	5
##	as.POSIXlt.POSIXct	as.qr
##	4	2
##	as.raw	as.single
##	1	3
##	as.single.default	as.symbol
##	3	2
##	as.table	as.table.default
##	3	3
##	as.vector	as.vector.factor
##	3	3
##	asin	asinh
##	1	1
##	asNamespace	asplit
##	aswamespace 3	3
##	asS3	asS4
##	4	4
##		
##	assign 7	atan 1
##	atan2	atanh
	3	1
		attachNamespace
##	attach	<del>-</del>
## ##	E	
## ## ##	5	6
## ## ## ##	attr	attr.all.equal
## ## ## ##	attr 1	attr.all.equal
## ## ## ## ##	attr 1 attr<-	attr.all.equal 6 attributes
## ## ## ## ## ##	attr 1 attr<- 1	attr.all.equal 6 attributes 1
## ## ## ## ## ##	attr 1 attr<- 1 attributes<-	attr.all.equal 6 attributes 1 autoload
## ## ## ## ## ##	attr 1 attr<- 1	attr.all.equal 6 attributes 1

##	4	6
##	baseenv	basename
##	1	2
##	besselI	besselJ
##	4	3
##	besselK	besselY
##	4	3
##	beta	bindingIsActive
##	3	3
##	bindingIsLocked	bindtextdomain
##	3	3
##	bitwAnd	bitwNot
##	3	2
##	bitw0r	bitwShiftL
##	3	3
##	bitwShiftR	bitwXor
##	3	3
##	body	body<-
##	2	4
##	bquote	break
##	4	1
##	browser	browserCondition
##	1	2
##	browserSetDebug	browserText
##	2	2
##	builtins	by
##	2	6
##	by.data.frame	by.default
##	6	6
##	bzfile	C
##	5	1
##	c.Date	c.difftime
##	3	3
##	c.factor	c.noquote
##	3	3
##	c.numeric_version	c.POSIXct
##	3	3
##	c.POSIX1t	c.warnings
##	3	3
##	call	callCC
##	1	2
##	capabilities	casefold
##	capabilities 3	3
##	cat	cbind
##	7	3
##	cbind.data.frame	ceiling
##	cbind.data.irame	terring 1
##	char.expand	character
##	char.expand 4	2
## ##	charmatch	charToRaw
## ##	charmaten 4	char rokaw 2
## ##	chartr	check_tzones
## ##	chartr 4	check_tzones
## ##		
##	chkDots	chol

##	4	3
##	chol.default	chol2inv
##	6	4
##	choose	class
##	3	1
##	class<-	clearPushBack
##	1	2
##	close	close.connection
##	3	4
##	close.srcfile	close.srcfilealias
##	3	3
##	closeAllConnections	col
##	1	3
##		
	colMeans 4	colnames
##		4
##	colnames<-	colSums
##	3	4
##	commandArgs	comment
##	2	2
##	comment<-	complex
##	3	6
##	computeRestarts	conditionCall
##	2	2
##	${\tt conditionCall.condition}$	${\tt condition} {\tt Message}$
##	2	2
##	${\tt condition} {\tt Message.condition}$	conflictRules
##	2	4
##	conflicts	Conj
##	3	1
##	contributors	cos
##	1	1
##	cosh	cospi
##	1	1
##	crossprod	 Cstack_info
##	3	1
##	cummax	cummin
##	1	1
##		
##	cumprod 1	cumsum 1
##	curlGetHeaders	
		cut
##	6	3
##	cut.Date	cut.default
##	7	9
##	cut.POSIXt	data.class
##	7	2
##	data.frame	data.matrix
##	7	3
##	date	debug
##	1	5
##	${\tt debuggingState}$	debugonce
##	2	5
##	default.stringsAsFactors	${\tt delayedAssign}$
##	1	5
##	deparse	deparse1
	•	•

##	6	5
##	det	detach
##	3	6
##	determinant	determinant.matrix
##	4	4
##	dget	diag
##	3	5
##	diag<-	diff
##	3	3
##	diff.Date	diff.default
##	5	5
##	diff.difftime	diff.POSIXt
##	3	5
##	difftime	digamma
##	5	1
##	dim	dim.data.frame
##	1	2
##	dim<-	dimnames
##	1	1
##	dimnames.data.frame	dimnames<-
##	2	1
##	dimnames <data.frame< td=""><td>dir</td></data.frame<>	dir
##	3	9
##	dir.create	dir.exists
##	5	2
##	dirname	do.call
##	2	5
##	dontCheck	double
##	2	2
##	dput	dQuote
##	4	3
##	drop	droplevels
##	2	3
##	droplevels.data.frame	droplevels.factor
##	5	4
##	dump	duplicated
##	7	4
##	duplicated.array	duplicated.data.frame
##	6	5
##	duplicated.default	duplicated.matrix
##	6	6
##	duplicated.numeric_version	duplicated.POSIX1t
##	4	4
##	duplicated.warnings	dyn.load
##	4	5
##	dyn.unload	dynGet
##	2	5
##	eapply	eigen
##	6	5
##	emptyenv	enc2native
##	1	1
##	enc2utf8	encodeString
##	1	6
##	Encoding	Encoding<-
	<u> </u>	· ·

##	2	3
##	endsWith	enquote
##	3	2
##	env.profile	environment
##	2	2
##	environment<-	environmentIsLocked
##	1	2
##	environmentName	errorCondition
##	2	5
##	eval	eval.parent
##	4	3
##	evalq	exists
##	4	7
##	exp	expand.grid
##	1	4
##	expm1	expression
##	1	1
##	extSoftVersion	factor
##	1	7
##	factorial	fifo
##	2	5
##	file	file.access
##	7	3
##	file.append	file.choose
##	3	2
##	file.copy	file.create
##	7	3
##	file.exists	file.info
##	2	3
##	file.link	file.mode
##	3	111e.mode 2
##	file.mtime	
	111e.mtime	file.path
##	_	5:12
##	file.remove	file.rename
##	2	3
##	file.show	file.size
##	7	2
##	file.symlink	Filter
##	3	3
##	Find	find.package
##	5	5
##	findInterval	findPackageEnv
##	6	2
##	findRestart	floor
##	3	1
##	flush	flush.connection
##	2	2
##	for	force
##	1	2
##	forceAndCall	formals
##	1	3
##	formals<-	format
##	4	3
##	format.AsIs	format.data.frame

##	4	4
##	format.Date	format.default
##	3	17
##	format.difftime	format.factor
##	3	3
##	format.hexmode	format.info
##	5	4
##	format.libraryIQR	format.numeric_version
##	3	3
##	format.octmode	format.packageInfo
##	4	3
##	format.POSIXct	format.POSIX1t
##	6	5
##	format.pval	format.summaryDefault
##	6	4 farmat DI
##	formatC	formatDL
## ##	16 forwardsolve	6 function
##	101 wardsolve	1
##	gamma	
##	1	gc 4
##	gc.time	gcinfo
##	1	2
##	gctorture	gctorture2
##	2	4
##	get	get0
##	6	6
##	getAllConnections	getCallingDLL
##	1	3
##	${\tt getCallingDLLe}$	getConnection
##	2	2
##	getDLLRegisteredRoutines	${\tt getDLLRegisteredRoutines.character}$
##	3	3
##	<pre>getDLLRegisteredRoutines.DLLInfo</pre>	getElement
##	3	3
##	geterrmessage	getExportedValue
##	1	3
## ##	getHook 2	getLoadedDLLs 1
##	=	_
##	getNamespace 2	${ t getNamespaceExports} \ 2$
##	${\tt getNamespaceImports}$	${\tt getNamespaceInfo}$
##	getwamespace1mpo1ts 2	getNameSpaceInio
##	getNamespaceName	getNamespaceUsers
##	2	2
##	getNamespaceVersion	$\tt getNativeSymbolInfo$
##	2	5
##	getOption	getRversion
##	3	1
##	getSrcLines	${\tt getTaskCallbackNames}$
##	4	1
##	gettext	gettextf
##	3	4
##	getwd	gl

##	1	6
##	globalCallingHandlers	globalenv
##	2	1
##	gregexec	gregexpr
##	7	7
##	grep	grepl
##	9	7
##	grepRaw	grouping
##	9	2
##	gsub	gzcon
##	8	5
##	gzfile	I
##	5	2
##	iconv	iconvlist
##	7	1
##	icuGetCollate	icuSetCollate
##	2	2
## ##	identical 9	identity 2
##	if	ifelse
##	1	4
##	Im	importIntoEnv
##	1	5
##	infoRDS	inherits
##	2	4
##	integer	interaction
##	2	5
##	interactive	intersect
##	1	4
##	intToBits	intToUtf8
##	2	4
##	inverse.rle	invisible
##	3	1
##	invokeRestart	invokeRestartInteractively
##	3	
##	is.array	is.atomic
## ##	1 is.call	1 is.character
##	is.caii	is.character
##	is.complex	is.data.frame
##	13.00mpicx	2
##	is.double	is.element
##	1	3
##	is.environment	is.expression
##	1	1
##	is.factor	is.finite
##	2	1
##	is.function	is.infinite
##	1	1
##	is.integer	is.language
##	1	1
##	is.list	is.loaded
##	1	
##	is.logical	is.matrix

##	1	1
##	is.na	is.na.data.frame
##	1	2
##	is.na.numeric_version	is.na.POSIX1t
##	2	2
##	is.na<-	is.na <default< th=""></default<>
##	3	3
##	is.na <factor< th=""><th>is.na<numeric_version< th=""></numeric_version<></th></factor<>	is.na <numeric_version< th=""></numeric_version<>
##	3	3
## ##	is.name 1	is.nan 1
##	is.null	is.numeric
##	15.11111	1s.numeric
##	is.numeric.Date	is.numeric.difftime
##	2	2
##	is.numeric.POSIXt	is.numeric_version
##	2	2
##	is.object	is.ordered
##	1	2
##	is.package_version	is.pairlist
##	2	1
##	is.primitive	is.qr
##	2	2
##	is.R	is.raw
##	. 1	1
## ##	is.recursive 1	is.single 1
##	is.symbol	is.table
##	13.3ymb01	2
##	is.unsorted	is.vector
##	4	3
##	isa	isatty
##	3	2
##	isBaseNamespace	isdebugged
##	2	3
##	isFALSE	isIncomplete
##	2	2
##	isNamespace	isNamespaceLoaded
##	2	7001
## ##	ISOdate 8	ISOdatetime 8
##	o isOpen	o isRestart
##	1sopen 3	isnestart 2
##	isS4	isSeekable
##	1	2
##	isSymmetric	isSymmetric.matrix
##	3	5
##	isTRUE	jitter
##	2	4
##	julian	julian.Date
##	3	4
##	julian.POSIXt	kappa
##	4	3
##	kappa.default	kappa.lm

##	6	3
##	kappa.qr	kronecker
##	3	6
##	l10n_info	La.svd
##	1	4
##	La_library	La_version
##	1	1
##	labels	labels.default
##	3	3
##	lapply	lazyLoad
##	4	4
##	lazyLoadDBexec	lazyLoadDBfetch
##	4 lbeta	1
## ##	Theta 3	lchoose 3
##	length	length.POSIX1t
##	1 rength	rength.rusixit
##	length<-	length <date< td=""></date<>
##	1	3
##	length <difftime< td=""><td>length<factor< td=""></factor<></td></difftime<>	length <factor< td=""></factor<>
##	3	3
##	length <posixct< td=""><td>length<posix1t< td=""></posix1t<></td></posixct<>	length <posix1t< td=""></posix1t<>
##	3	3
##	lengths	levels
##	3	2
##	levels.default	levels<-
##	2	1
##	levels <factor< td=""><td>lfactorial</td></factor<>	lfactorial
##	3	2
##	lgamma	libcurlVersion
##	1	1
##	library	library.dynam
##	14	7
##	library.dynam.unload	licence
##	5	1
##	license	list
##	1	1
##	list.dirs	list.files
##	4	9
##	list2DF	list2env
##	3	6
##	load	loadedNamespaces
##	4	l and Namagna an
## ##	loadingNamespaceInfo 1	loadNamespace 7
##	local	lockBinding
##	3	3
##	lockEnvironment	
##	3	log 1
##	log10	log1p
##	10010	10819
##	log2	logb
##	1	3
##	logical	lower.tri
	1001041	10,01.011

3	2	##
make.names	ls	##
4	7	##
makeActiveBinding	make.unique	##
4	3	##
mapply	Map	##
		##
marginSums	margin.table	##
3	_	##
match		##
į.		##
match.call		##
3		##
Math.data.frame		##
3		##
Math.difftime		##
		##
Math.POSIX		##
ilduli i dbix		##
max		##
ind2		##
		##
mear 3		##
		##
mean.default		##
mean.POSIXct		##
		##
mem.maxNSize		##
2		##
memCompress		##
3		##
memory.profile	1	##
		##
merge.data.frame	· O·	##
14		##
message	8	##
4		##
mir	8	##
-		##
Mod	0	##
<u> </u>		##
mode<-		##
3	2	##
months.Date	months	##
3	3	##
mostattributes<-	months.POSIXt	##
3	3	##
names.POSIX1t	names	##
2	1	##
names <posix1t< td=""><td>names&lt;-</td><td>##</td></posix1t<>	names<-	##
3	1	##
namespaceImport	namespaceExport	##
	3	##
${\tt namespaceImportFrom}$	namespaceImportClasses	##

##	5	8
##	${\tt namespaceImportMethods}$	nargs
##	5	1
##	nchar	ncol
##	5	2
##	NCOL	Negate
##	2	2
##	new.env	next
##	4	1
##	NextMethod	ngettext
##	4	5
##	nlevels	noquote
##	2	3
##	norm	normalizePath
##	3	4
##	nrow	NROW
##	2	2
##	nullfile	numeric
##	1	2
##	numeric_version	numToBits
##	3	2
##	numToInts	nzchar
##	2	1
##	objects	oldClass
##	7	1
##	oldClass<-	OlsonNames
##	1	2
##	on.exit	open
##	1	3
##	open.connection	open.srcfile
##	5	4
##	open.srcfilealias	open.srcfilecopy
##	4	4
##	Ops.data.frame	Ops.Date
##	3	3
##	Ops.difftime	Ops.factor
##	3	3
##	Ops.numeric_version	Ops.ordered
##	3	3
##	Ops.POSIXt	options
##	3	2
##	order	ordered
##	5	3
##	outer	package_version
##	5	package_version 3
##	packageEvent	packageHasNamespace
##	packageEvent 3	packagenasnamespace 3
##	packageNotFoundError	
##	packageworrounderror 4	$ exttt{packageStartupMessage} \ 4$
##		
	$ exttt{packBits}$	pairlist
##		2
##	parent.env	parent.env<-
##	2	3
##	parent.frame	parse

##	2	8
##	${\tt parseNamespaceFile}$	paste
##	4	5
##	paste0	path.expand
##	4	2
##	path.package	pcre_config
##	3	1
##	pipe	plot
##	4	4
##	pmatch	pmax
##	5	. 3
##	pmax.int	pmin
##	3	3
##	pmin.int	polyroot
##	3	2
##	pos.to.env	pretty
##	1	3
##	pretty.default	prettyNum
##	9	14
##	print	print.AsIs
##	3	3
##	print.by	<pre>print.condition</pre>
##	4	3
##	print.connection	print.data.frame
##	3	8
##	print.Date	print.default
##	4	11
##	print.difftime	print.Dlist
##	4	3
##	print.DLLInfo	print.DLLInfoList
##	3	3
##	print.DLLRegisteredRoutines	print.eigen
##	3	3
##	print.factor	print.function
##	6	4
##	<pre>print.hexmode</pre>	print.libraryIQR
##	3	3
##	print.listof	print.NativeRoutineList
##	3	3
##	print.noquote	print.numeric_version
##	5	4
##	print.octmode	print.packageInfo
##	3	3
##	print.POSIXct	print.POSIX1t
##	6	6
##	<pre>print.proc_time</pre>	print.restart
##	3	3
##	print.rle	<pre>print.simple.list</pre>
##	5	3
##	<pre>print.srcfile</pre>	print.srcref
##	3	4
##	<pre>print.summary.table</pre>	<pre>print.summary.warnings</pre>
##	4	3
##	<pre>print.summaryDefault</pre>	print.table

##	4	9
##	print.warnings	prmatrix
##	5	8
##	proc.time	prod
##	1	1
##	prop.table	proportions
##	3	3
##	provideDimnames	psigamma
##	5	3
##	pushBack	pushBackLength
##	5	2
##	q	qr
##	4	3
## ##	qr.coef 3	qr.default 5
## ##		
##	qr.fitted 4	qr.Q 4
##	qr.qty	
##	41.4ty 3	qr.qy 3
##	qr.R	qr.resid
##	3	3
##	qr.solve	qr.X
##	4	4
##	quarters	quarters.Date
##	3	3
##	quarters.POSIXt	quit
##	3	4
##	quote	R.home
##	1	2
##	R.Version	$R_system_version$
##	1	3
##	range	range.default
##	1	4
##	rank	rapply
##	4	7
##	raw	rawConnection
##	2	3
##	rawConnectionValue	rawShift
##	2	3
##	rawToBits	rawToChar
##	2	3
##	rbind	rbind.data.frame
## ##	3	6
## ##	rcond 5	Re 1
##	read.dcf	readBin
## ##	read.dci 5	reaubin 7
##	readChar	readline
##	1 eadonar 4	readille 2
##	readLines	readRDS
##	readLines 7	3
##	$\overset{'}{ ext{readRenviron}}$	Recall
##	2	2
##	Reduce	reg.finalizer
		100.1111111111

4	<b>#</b> 6	##
regexpr	regexec	##
7	‡ 7	##
registerS3methods	registerS3method	##
4	<b>‡</b> 5	##
regmatches<-	regmatches	##
5	‡ 4	##
${\tt removeTaskCallback}$	‡ remove	##
2	<b>‡</b>	##
rep.Date	‡ rep	##
3		##
rep.factor	1	##
3		##
rep.numeric_version	1	##
3		##
rep.POSIX1t	1	##
3		##
repeat	1 =	##
1		##
replicate	1	##
4		##
requireNamespace	1	##
4		##
restartFormals	1	##
2		## ##
return 1		##
rev		##
2		##
rle		##
2		##
- RNGkind		##
4		##
round		##
1		##
round.POSIXt		##
3	<b>‡</b> 3	##
row.names	‡ row	##
2	‡ 3	##
row.names.default	row.names.data.frame	##
2	‡ 2	##
row.names <data.frame< th=""><th>t row.names&lt;-</th><th>##</th></data.frame<>	t row.names<-	##
3	<b>‡</b> 3	##
rowMeans	row.names <default< td=""><td>##</td></default<>	##
4	‡ 3	##
rownames<-	rownames	##
3	‡ 4	##
rowsum.data.frame	‡ rowsum	##
6	<b>‡</b> 5	##
rowSums	t rowsum.default	##
4		##
sample.int	1	##
6		##
save	# sapply	##

##	6	11
##	save.image	saveRDS
##	6	7
##	scale	scale.default
##	4	4
##	scan	search
##	23	1
##	searchpaths	seek
##	1	3
##	seek.connection	seq
##	6	2
##	seq.Date	seq.default
##	7	7
##	seq.int	seq.POSIXt
##	1	7
##	seq_along	seq_len
##	1	1
##	sequence	sequence.default
##	3	5
##	serialize	serverSocket
##	7	2
##	set.seed	setdiff
##	5	4
##	setequal	setHook
##	4	4
##	${\tt setNamespaceInfo}$	${\tt setSessionTimeLimit}$
##	4	3
##	setTimeLimit	setwd
##	4	2
##	shell	shell.exec
##	9	2
##	${\tt showConnections}$	shQuote
##	2	3
##	sign	${\tt signalCondition}$
##	1	2
##	signif	${\tt simpleCondition}$
##	1	3
##	simpleError	${ t simple Message}$
##	3	3
##	simpleWarning	simplify2array
##	3	3
##	sin	single
##	1	2
##	sinh	sink
##	1	5
##	sink.number	sinpi
##	2	1
##	slice.index	socketAccept
##	3	7
##	${\tt socketConnection}$	socketSelect
##	9	4
##	socketTimeout	solve
##	3	4
##	solve.default	solve.qr

##	6	4
##	sort	sort.default
##	4	5
##	sort.int	sort.list
##	7	6
##	sort.POSIX1t	source
##	5	17
##	split	split.data.frame
##	5	5
##	split.Date	split.default
##	5	7
##	split.POSIXct	split<-
##	5	6
##	split <data.frame< th=""><th>split<default< th=""></default<></th></data.frame<>	split <default< th=""></default<>
##	6	6
##	sprintf	sqrt
##	3	1
##	sQuote	srcfile
##	3	4
##	srcfilealias	srcfilecopy
##	3	5
##	srcref	standardGeneric
##	3	1
##	startsWith	stderr
##	3	1
##	stdin	stdout
##	1	1
##	stop	stopifnot
##	4	5
##	storage.mode	storage.mode<-
##	2	1
##	str2expression	str2lang
##	2	2
##	strftime	strptime
##	6	4
##	strrep	strsplit
##	3	6
##	strtoi	strtrim
##	3	3
##	structure	strwrap
##	3	8
##	sub	subset
##	8	3
##	subset.data.frame	subset.default
##	6	4
##	subset.matrix	substitute
##	Subset.matrix	substitute 1
##	substr	substr<-
##	substi 4	5
##		
##	substring 4	substring<- 5
## ##		
##	sum 1	summary 3
##	1	
##	summary.connection	summary.data.frame

##	3	5
##	Summary.data.frame	summary.Date
##	3	4
##	Summary.Date	summary.default
##	3	5
##	Summary.difftime	summary.factor
##	3	4
##	Summary.factor	summary.matrix
##	3	3
##	Summary.numeric_version	Summary.ordered
##	3	3
##	summary.POSIXct	Summary.POSIXct
##	4	3
##	summary.POSIX1t	Summary.POSIX1t
##	4	3
##	summary.proc_time	summary.srcfile
##	3	3
## ##	summary.srcref 4	summary.table 3
##	summary.warnings	suppressMessages
##	3	Suppressivessages
##	suppressPackageStartupMessages	suppressWarnings
##	2	3
##	suspendInterrupts	svd
##	2	5
##	sweep	switch
##	7	1
##	sys.call	sys.calls
##	2	1
##	Sys.chmod	Sys.Date
##	4	1
##	sys.frame	sys.frames
##	2	1
##	sys.function	Sys.getenv
##	2	4
##	Sys.getlocale	Sys.getpid
##	2	1
##	Sys.glob	Sys.info
##	3	1
##	Sys.junction	sys.load.image
##	3	3
##	Sys.localeconv	sys.nframe
## ##	1 sys.on.exit	1 sys.parent
##	sys.on.exit	sys.parent
##	sys.parents	Sys.readlink
##	1	2
##	sys.save.image	Sys.setenv
##	2	2
##	Sys.setFileTime	Sys.setlocale
##	3	3
##	Sys.sleep	sys.source
##	2	7
##	sys.status	Sys.time
	-	

##	1	1
##	Sys.timezone	Sys.umask
##	2	2
##	Sys.unsetenv	Sys.which
##	2	2
##	system	system.file
##	11	5
##	system.time	system2
##	3	12
##	t	t.data.frame
##	2	2
##	t.default	table
##	2	6
##	tabulate	tan
##	3	1
##	tanh	tanpi
##	1	1
##	tapply	taskCallbackManager
##	7	4
##	tcrossprod	tempdir
##	3	2
##	tempfile	textConnection
##	4	6
##	textConnectionValue	tolower
##	2	2
##	topenv	toString
##	3	3
##	toString.default	toupper
##	4	2
##	trace	traceback
##	9	3
##	tracemem	tracingState
##	1	2
##	transform	transform.data.frame
##	3	3
##	transform.default	trigamma
##	3	1
##	trimws	trunc
##	4	1
##	trunc.Date	trunc.POSIXt
##	3	4
##	truncate	truncate.connection
##	3	3
##	try	tryCatch
##	4	4
##	tryInvokeRestart	typeof
##	3	2
##	unclass	undebug
##	1	3
##	union	unique
##	4	4
##	unique.array	unique.data.frame
##	6	5
##	unique.default	unique.matrix
	uniquo.uciauit	diiiquo.matiix

##	6	6
##	unique.numeric_version	unique.POSIX1t
##	4	4
##	unique.warnings	units
##	4	2
##	units.difftime	units<-
##	2	3
##	units <difftime< th=""><th>unix.time</th></difftime<>	unix.time
##	3	2
##	unlink	unlist
##	5	4
##	${\tt unloadNamespace}$	unlockBinding
##	2	3
##	unname	unserialize
##	3	3
##	unsplit	untrace
##	4	4
##	untracemem	unz
##	1	5
##	upper.tri	url
##	3	7
##	UseMethod	utf8ToInt
##	1	2
##	validEnc	validUTF8
##	2	2
##	vapply	vector
##	6	3
##	Vectorize	warning
##	5	6
##	warningCondition	warnings
##	5	2
##	weekdays	weekdays.Date
##	3	3
##	weekdays.POSIXt	which
##	3	4
##	which.max	which.min
##	2	2
##	while	with
##	1	4
##	with.default	withAutoprint
##	4	10
##	withCallingHandlers	within
##	3	4
##	within.data.frame	within.list
##	4	5
##	withRestarts	withVisible
##	3	2
##	write	write.dcf
##	6	8
##	writeBin	writeChar
##	wiltebin 6	wiltechar 6
##	writeLines	
##	writeLines 5	xor 3
##		xtfrm
##	xpdrows.data.frame	Xtirm

```
##
                            xtfrm.AsIs
##
                                                            xtfrm.data.frame
##
##
                            xtfrm.Date
                                                               xtfrm.default
##
                        xtfrm.difftime
                                                                xtfrm.factor
##
##
                                                               xtfrm.POSIXct
                 xtfrm.numeric_version
##
##
                         xtfrm.POSIX1t
                                                                      xzfile
##
##
                                                                            5
##
                               zapsmall
##
```

# 6.4.4 Dynamic lookup

```
g12 <- function() x + 1
x <- 15
g12()

## [1] 16

x <- 20
g12()

## [1] 21

codetools::findGlobals(g12)

## [1] "+" "x"

environment(g12) <- emptyenv()
g12()</pre>
```

## Error in x + 1: could not find function "+"

# 6.4.5 Exercises

1. What does the following code return? Why? Describe how each of the three c's is interpreted.

```
c <- 10
c(c = c)
```

## c ## 10

Creates a vector with one named element "c" with a value of 10

2. What are the four principles that govern how R looks for values?

Name masking, Functions versus variables, A fresh start, Dynamic lookup

3. What does the following function return? Make a prediction before running the code yourself.

```
f <- function(x) {
   f <- function(x) {
      f <- function() {
        x ^ 2
    }
    f() + 1
   }
   f(x) * 2
}</pre>
```

## [1] 202

202

# 6.5 Lazy evaluation

```
h01 <- function(x) {
   10
}
h01(stop("This is an error!"))</pre>
```

## [1] 10

#### 6.5.1 Promises

```
y <- 10
h02 <- function(x) {
  y <- 100
  x + 1
}</pre>
```

```
## [1] 11
```

```
h02(y <- 1000)
```

## [1] 1001

```
## [1] 1000
double <- function(x) {</pre>
 message("Calculating...")
 x * 2
}
h03 <- function(x) {
c(x, x)
h03(double(20))
## Calculating...
## [1] 40 40
6.5.2 Default arguments
h04 \leftarrow function(x = 1, y = x * 2, z = a + b) {
 a <- 10
 b <- 100
 c(x, y, z)
h04()
## [1] 1 2 110
h05 \leftarrow function(x = ls()) {
 a <- 1
 Х
}
# ls() evaluated inside h05:
h05()
## [1] "a" "x"
# ls() evaluated in global environment:
h05(ls())
                                 "c"
                                                                       "f"
## [1] "a"
                    "args"
                                              "deviation" "double"
## [7] "f01"
                    "f02"
                                 "f1"
                                              "f2"
                                                                       "g01"
                                                          "funs"
## [13] "g02"
                                                          "g08"
                    "g03"
                                 "g04"
                                              "g07"
                                                                       "g09"
                                              "h01"
                                                          "h02"
## [19] "g10"
                    "g11"
                                 "g12"
                                                                       "h03"
## [25] "h04"
                    "h05"
                                 "objs"
                                              "out"
                                                          "square"
                                                                       "x"
## [31] "y"
```

#### 6.5.3 Missing arguments

```
h06 \leftarrow function(x = 10) {
 list(missing(x), x)
}
str(h06())
## List of 2
## $ : logi TRUE
## $ : num 10
str(h06(10))
## List of 2
## $ : logi FALSE
## $ : num 10
args(sample)
## function (x, size, replace = FALSE, prob = NULL)
## NULL
sample <- function(x, size = NULL, replace = FALSE, prob = NULL) {</pre>
  if (is.null(size)) {
    size <- length(x)</pre>
  }
  x[sample.int(length(x), size, replace = replace, prob = prob)]
`%||%` <- function(lhs, rhs) {</pre>
 if (!is.null(lhs)) {
    lhs
  } else {
    rhs
}
sample <- function(x, size = NULL, replace = FALSE, prob = NULL) {</pre>
 size <- size %||% length(x)</pre>
 x[sample.int(length(x), size, replace = replace, prob = prob)]
```

# 6.5.4 Exercises

1. What important property of && makes x\_ok() work?

```
x_ok <- function(x) {
  !is.null(x) && length(x) == 1 && x > 0
}
x_ok(NULL)
```

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

```
x_ok(1)
```

## [1] TRUE

```
x_{ok}(1:3)
```

# ## [1] FALSE

Processed from left to right. Prevents error later on by requiring the first tests to work.

```
x_ok <- function(x) {
  !is.null(x) & length(x) == 1 & x > 0
}
x_ok(NULL)
```

## logical(0)

```
x_{ok}(1)
```

## [1] TRUE

```
x_ok(1:3)
```

# ## [1] FALSE FALSE FALSE

Runs each logical test, no control flow. Provide values to tests that should not accept them

2. What does this function return? Why? Which principle does it illustrate?

```
f2 <- function(x = z) {
  z <- 100
  x
}
f2()</pre>
```

# ## [1] 100

100, lazy evaluation

3. What does this function return? Why? Which principle does it illustrate?

```
y <- 10
f1 <- function(x = {y <- 1; 2}, y = 0) {
  c(x, y)
}
f1()</pre>
```

# ## [1] 2 1

у

## [1] 10

- (2, 1), 10. name masking
  - 4. In hist(), the default value of xlim is range(breaks), the default value for breaks is "Sturges", and

```
range("Sturges")
```

```
## [1] "Sturges" "Sturges"
```

Explain how hist() works to get a correct xlim value.

Sturges is a function called which calculates bins for you

5. Explain why this function works. Why is it confusing?

```
show_time <- function(x = stop("Error!")) {
  stop <- function(...) Sys.time()
  print(x)
}
show_time()</pre>
```

```
## [1] "2022-10-10 08:47:40 PDT"
```

The default value for x is run the function stop with the argument "Error!". Which should stop and prevent an error message. The stop function is then redefined in the function to provide the system time instead. They evaluation is then applied to the value of x. Last x is print given its environmental value which is the system time. It's confusing because we are masking functions and lazy evaluating calls.

6. How many arguments are required when calling library()?

None