

These materials adapted by Amelia McNamara from the RStudio CC BY-SA materials Introduction to R (2014) and Master the Tidyverse (2017).

Introduction to R & RStudio: deck 03: Data types

Amelia McNamara

Visiting Assistant Professor of Statistical and Data Sciences
Smith College

January 2018

1. Vectors
2. Matrices
3. Data types
4. Coercion

Your Turn 1

Look at the R object
WorldPhones (by typing its
name in your notebook or the
Console and hitting enter).

What is inside of **WorldPhones**?

A digital timer with a black background and white numbers, displaying 00:30. The numbers are in a stylized, segmented font typical of digital displays.

WorldPhones

	N.Amer	Europe	Asia	S.Amer	Oceania	Africa	Mid.Amer
1951	45939	21574	2876	1815	1646	89	555
1956	60423	29990	4708	2568	2366	1411	733
1957	64721	32510	5230	2695	2526	1546	773
1958	68484	35218	6662	2845	2691	1663	836
1959	71799	37598	6856	3000	2868	1769	911
1960	76036	40341	8220	3145	3054	1905	1008
1961	79831	43173	9053	3338	3224	2005	1076

You can save more than a single number in an object by creating a *vector*, *matrix*, or *array*.

Vectors

Your turn

Which of these are numbers?
Which are words? How can you
tell?

1

"1"

"one"

00:30

Your turn

How many dimensions does a vector have?

1 2 3 4 5 6

A digital timer with a dark grey background and white LED-style digits. The display shows '00:30'.

How many dimensions does a vector have?



vectors

Combine multiple elements into a one dimensional array.

Create with the `c` function (for "concatenate").

```
vec <- c(1, 2, 3, 10, 100)
```

```
vec
```

Your turn

What happens in your Environment
when you run this code?

In your Notebook?

```
vec <- c(1, 2, 3, 10, 100)
```

```
vec
```

A digital timer display with a dark background and white, segmented digits. The digits show '00:30', indicating a time of 30 seconds.

vectors

Combine multiple elements into a one dimensional array.

Create with the `c` function (for "concatenate").

```
vec <- c(1, 2, 3, 10, 100)
```

```
vec
```

```
# 1  2  3 10 100
```



Matrices

Uncommon, but good to know

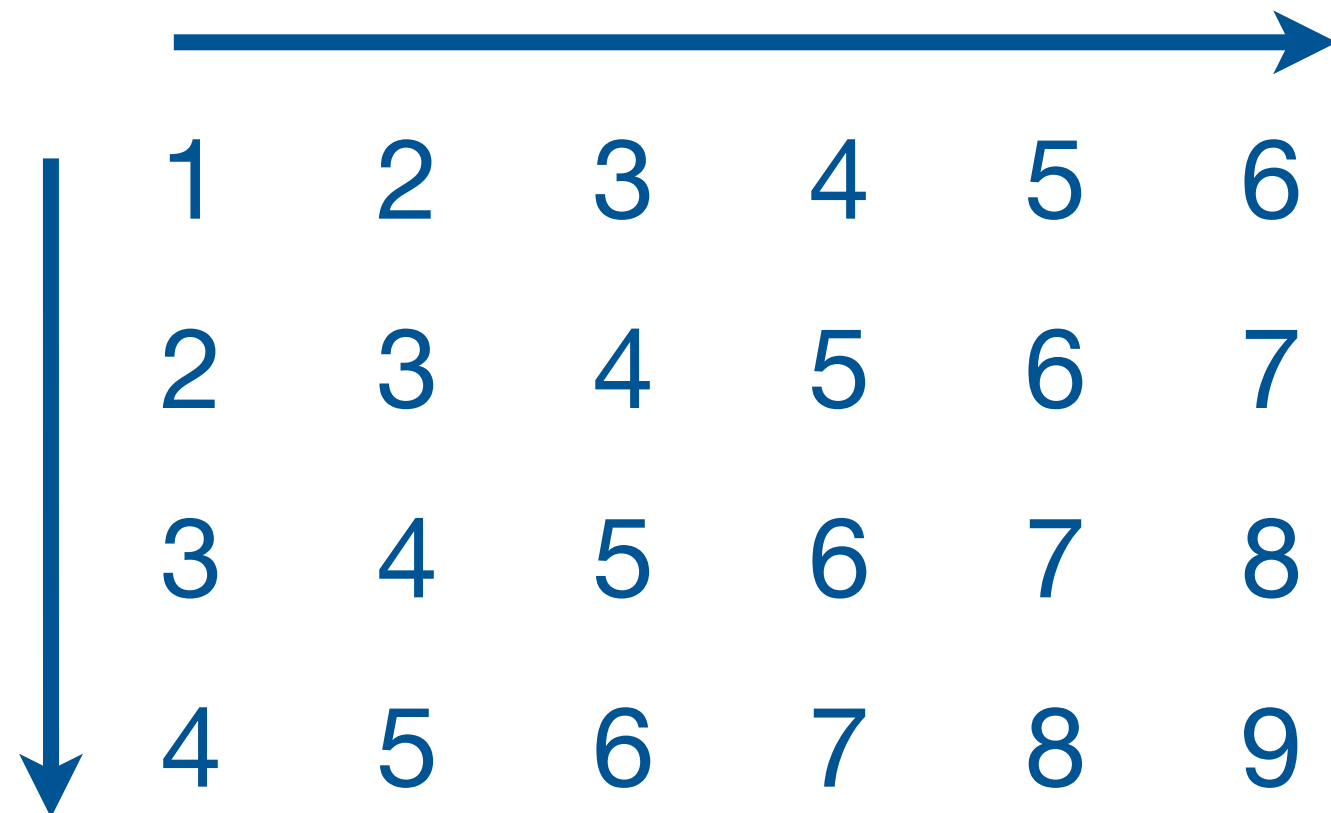
Your turn

How many dimensions does a matrix have?

1	2	3	4	5	6
2	3	4	5	6	7
3	4	5	6	7	8
4	5	6	7	8	9

A digital timer with a black background and white LED-style digits. The display shows '00:30'.

How many dimensions does a matrix have?



1	2	3	4	5	6
2	3	4	5	6	7
3	4	5	6	7	8
4	5	6	7	8	9

Your turn

The matrix below is named **M**.
What is the value of **M**₃₄?

0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23

A digital timer with a black background and white LED-style digits. The display shows '00:30'.

The matrix below is named **M**.
What is the value of **M**₃₄?



0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23

The matrix below is named **M**.
What is the value of **M**₃₄?



0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23

The matrix below is named **M**.
What is the value of **M**₃₄?



0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23

matrices

multiple elements stored in a two dimensional array.

Create with the `matrix` function.

```
mat <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
```

```
mat
```

```
#      [,1] [,2] [,3]
```

```
# [1,]  1   3   5
```

```
# [2,]  2   4   6
```

matrices

Combine multiple elements into a two dimensional array.

Create with the `matrix` function.

```
mat <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
```

```
mat
```

```
#      [,1] [,2] [,3]
```

```
# [1,]  1   3   5
```

```
# [2,]  2   4   6
```



vector of elements
to go in the matrix

```
matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
```

```
#      [,1] [,2] [,3]
```

```
# [1,]  1   3   5
```

```
# [2,]  2   4   6
```

number of rows for
matrix

```
matrix(c(1, 2, 3, 4, 5, 6), nrow = 2)
```

```
#      [,1] [,2] [,3]
```

```
# [1,]  1   3   5
```

```
# [2,]  2   4   6
```

```
matrix(c(1, 2, 3, 4, 5, 6), nrow = 3)
```

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

```
# [1,]    1    4
```

```
# [2,]    2    5
```

```
# [3,]    3    6
```


R as a
calculator
(again)

Math: element-wise

`vec + 4`

`# 5 6 7 14 104`

`vec * 4`

`# 4 8 12 40 400`

`vec * vec`

`# 1 4 9 100 10000`

RStudio

Project: (None)

Environment History Connections

Global Environment

Data

bechdel	1794 obs. of 15 ...
mat	num [1:2, 1:3] 1...

Values

vec	num [1:5] 1 2 3 10...
-----	-----------------------

```
27 matrix(c(1, 2, 3, 4, 5, 6), nrow = 3)
28 ```
29
30 ## Math with vectors and matrices
31
32 ```{r}
33 vec + 4
34 vec * 4
35 vec * vec # element-wise multiplication
36
37 vec %*% vec # matrix multiplication (inner)
38 vec %o% vec # matrix multiplication (outer)
39
40 mat
41 t(mat) # transpose
42 ```
43
44 ## Arrays
45
46 ```{r}
47 array(c(1, 2, 3, 4, 5, 6), dim = c(2, 2, 3))
48 ```
```

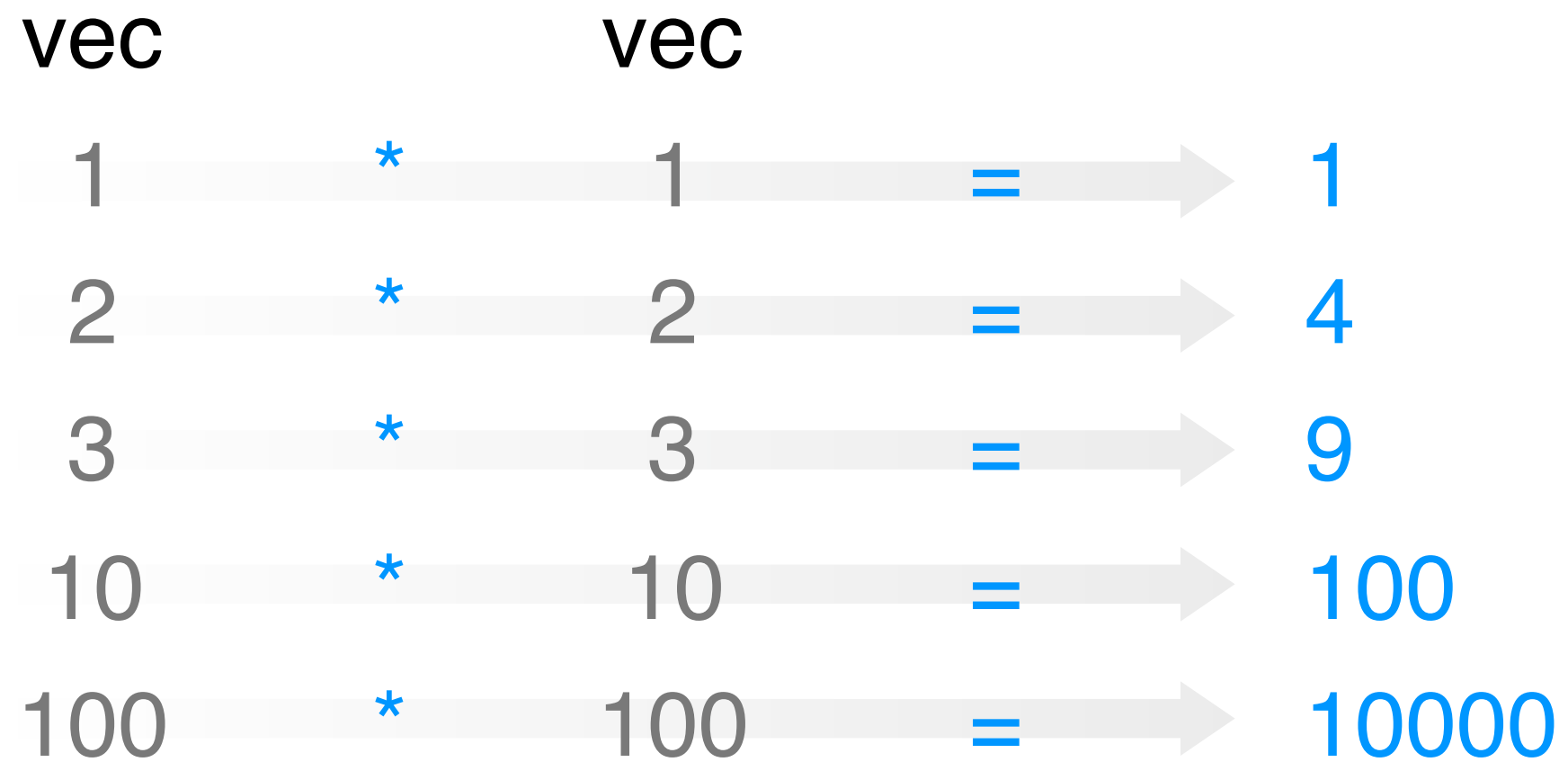
33:8 Chunk 4 R Markdown

Console

Green text indicates a code "comment," another way to document what you're doing. Comments aren't executed by R when you run a line.

vec * vec

1 4 9 100 10000



Matrix multiplication

```
vec %*% vec # inner
```

```
#      [,1]
```

```
# [1,] 10114
```

```
vec %o% vec # outer
```

```
#      [,1] [,2] [,3] [,4] [,5]
```

```
# [1,]    1    2    3   10   100
```

```
# [2,]    2    4    6   20   200
```

```
# [3,]    3    6    9   30   300
```

```
# [4,]   10   20   30  100  1000
```

```
# [5,]  100  200  300 1000 10000
```

mat

```
#      [,1] [,2] [,3]  
# [1,]    1    3    5  
# [2,]    2    4    6
```

t(mat)

```
#      [,1] [,2]  
# [1,]    1    2  
# [2,]    3    4  
# [3,]    5    6
```

arrays

Combine multiple elements into an array that has three or more dimensions.

Create with the `array` function.

```
array(c(1, 2, 3, 4, 5, 6), dim = c(2, 2, 3))
```

arrays

Combine multiple elements into an array that has three or more dimensions.

Create with the `array` function.

```
array(c(1, 2, 3, 4, 5, 6), dim = c(2, 2, 3))
```

Another uncommon
structure

Data types

Warm up

	A	B	C	D
1	date	president	democrat	<u>unemploy</u>
2	Mar 31, 1968	Lyndon Johnson	TRUE	2709
3	Apr 30, 1968	Lyndon Johnson	TRUE	2740
4	May 31, 1968	Lyndon Johnson	TRUE	2938
5	Jun 30, 1968	Lyndon Johnson	TRUE	2883
6	Jul 31, 1968	Lyndon Johnson	TRUE	2768
7	Aug 31, 1968	Lyndon Johnson	TRUE	2686
8	Sep 30, 1968	Lyndon Johnson	TRUE	2689
9	Oct 31, 1968	Lyndon Johnson	TRUE	2715
10	Nov 30, 1968	Lyndon Johnson	TRUE	2685
11	Dec 31, 1968	Lyndon Johnson	TRUE	2718
12	Jan 31, 1969	Richard Nixon	FALSE	2692
13	Feb 28, 1969	Richard Nixon	FALSE	2712
14	Mar 31, 1969	Richard Nixon	FALSE	2758
15	Apr 30, 1969	Richard Nixon	FALSE	2713
16	May 31, 1969	Richard Nixon	FALSE	2816
17	Jun 30, 1969	Richard Nixon	FALSE	2868
18	Jul 31, 1969	Richard Nixon	FALSE	2868
19	Aug 31, 1969	Richard Nixon	FALSE	
20	Sep 30, 1969	Richard Nixon	FALSE	
21	Oct 31, 1969	Richard Nixon	FALSE	
22	Nov 30, 1969	Richard Nixon	FALSE	

What types of data appear in this spreadsheet?

data types

Like Excel, Numbers, etc., R can recognize different types of data.

We'll look at four basic types:

- numbers
- character strings (text)
- logical
- factor

numeric

Any number, no quotes.

Appropriate for math.

1 + 1

3000000

class(0.00001)

"numeric"

character

Any symbols surrounded by quotes.

Appropriate for words, variable names, messages, any text.

```
"hello"
```

```
class("hello")
```

```
# "character"
```

```
"hello" + "world"
```

```
# Error
```

```
nchar("hello")
```

```
# 5
```

```
paste("hello", "world")
```

```
# "hello world"
```

Your turn

Which of these are numbers?
What are the others? How can
you tell?

1

"1"

"one"

A digital timer with a black background and white LED-style digits. The display shows "00:30".

00:30

logical

TRUE or FALSE

R's form of binary data. Useful for logical tests.

$3 < 4$

TRUE

class(TRUE)

"logical"

class(T)

"logical"

factor

R's form of categorical data. Saved as an integer with a set of labels (e.g. levels).

```
fac <- factor(c("a", "b", "c"))
```

```
fac
```

```
# a b c
```

```
# Levels: a b c
```

```
class(fac)
```

```
# factor
```

Use great caution with
factors

Quiz

```
x <- c(1, 2, 3)
```

What is the difference between these?

x

"x"

A digital timer with a dark grey background and white, segmented digits. The display shows '00:30'.

Type	Examples
numeric	0, 1, -2, 3.1415, 0.0005
character	"Amelia", "Agree", "31"
logical	TRUE, FALSE
factor	a c c b Levels: a b c

Your turn 2

Make a vector that contains the number 1, the letter R, and the logical TRUE.

What class of data is the vector?

```
vec <- c(1, "R", TRUE)
class(vec)
# "character"
```

```
vec
# "1"  "R"  "TRUE"
```

```
# What is R doing?
```

Your turn

Another way to see the class of an object is in the Environment pane. Does the Environment agree with what you found using `class()`?

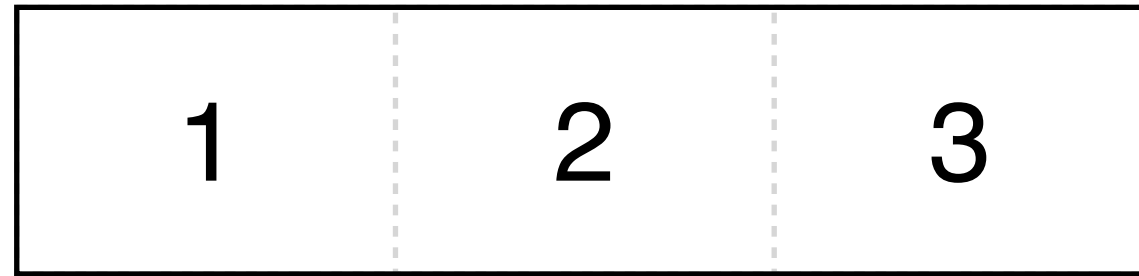
A digital timer with a black background and white numbers, displaying 00:30. The numbers are in a stylized, segmented font typical of digital displays.

00:30

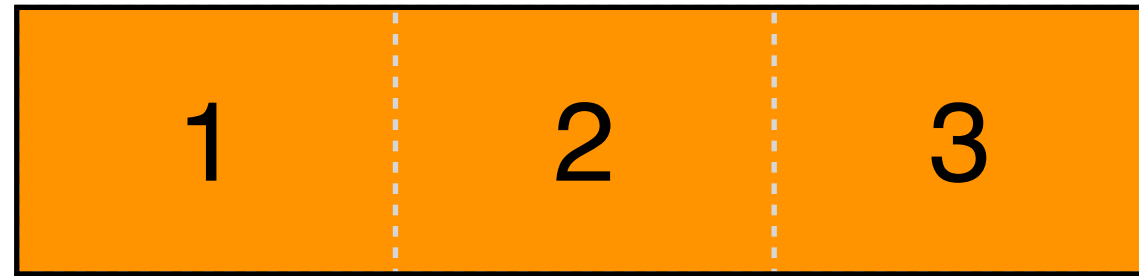
Vector



Vector

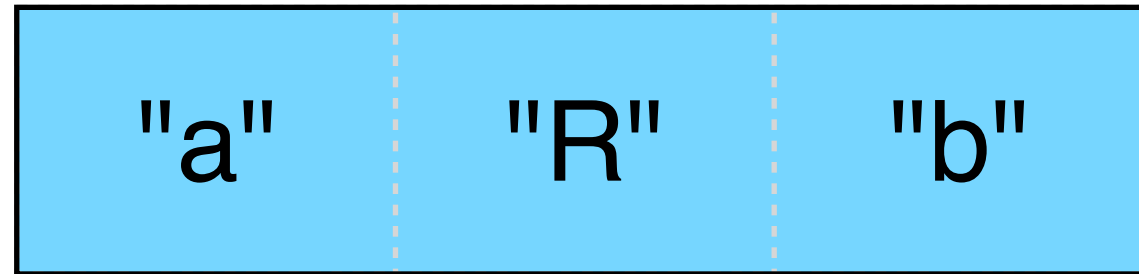


Vector



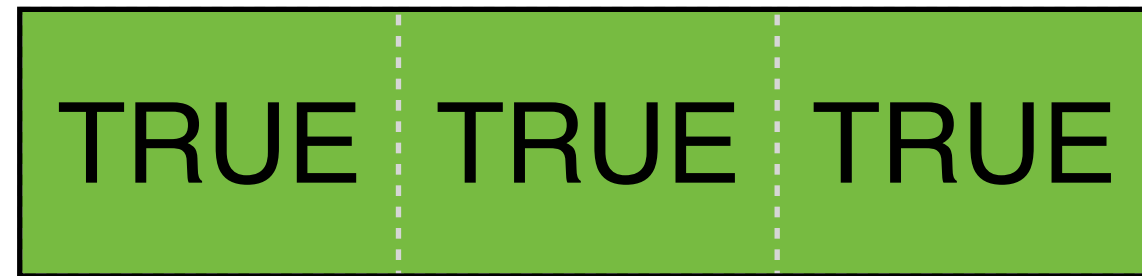
numeric

Vector



character

Vector



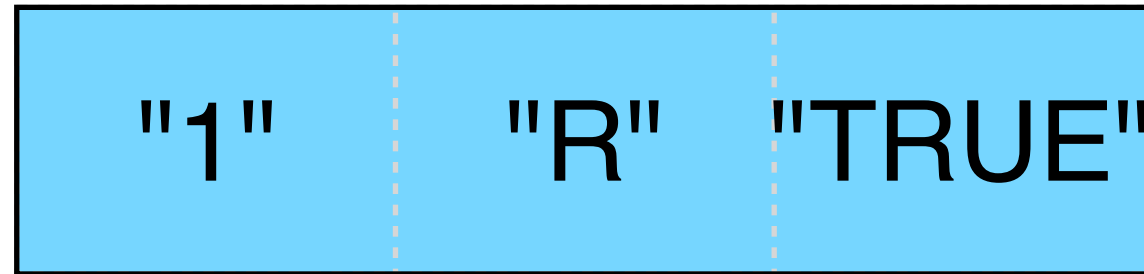
logical

Vector

1	"R"	TRUE
---	-----	------

?

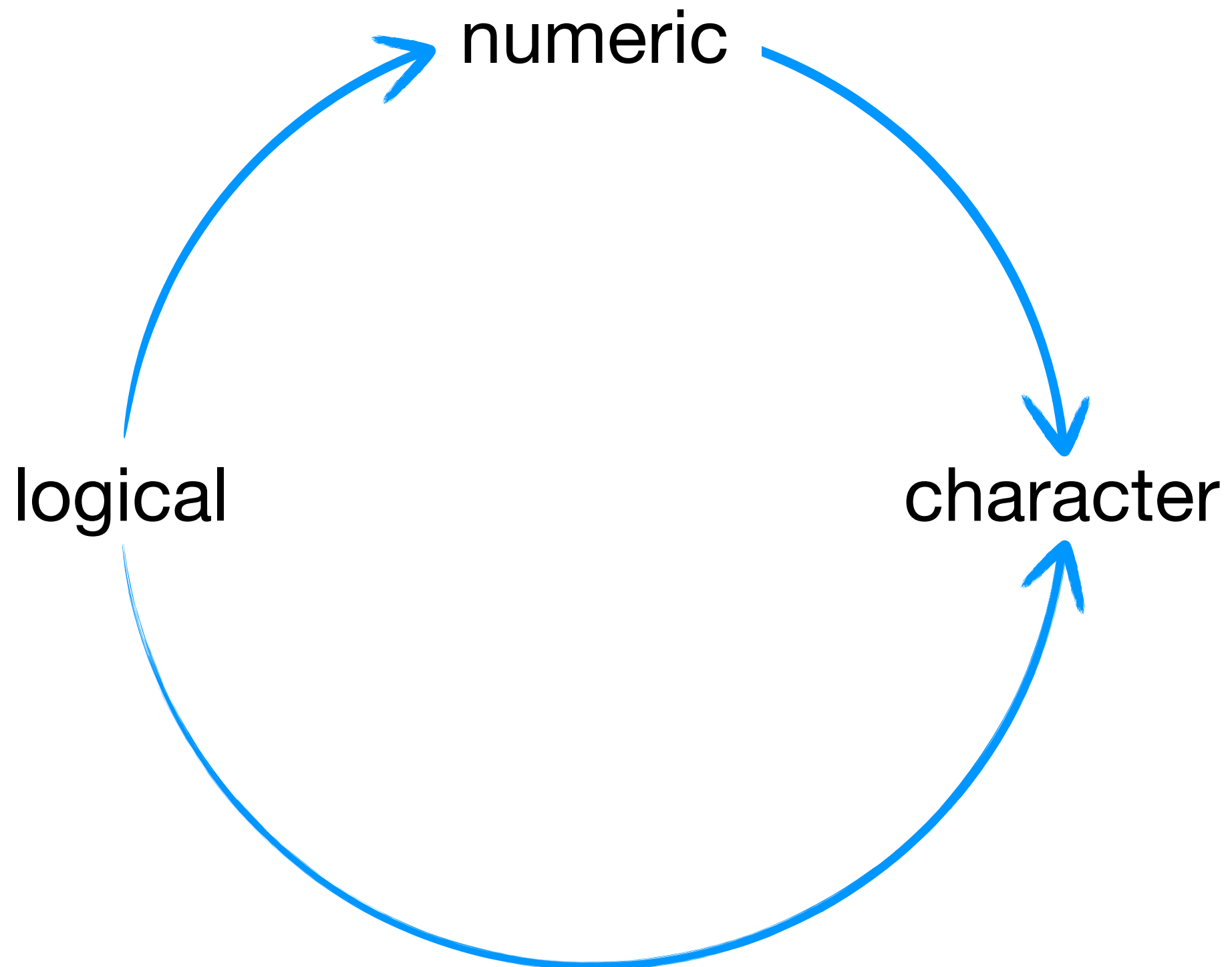
Vector



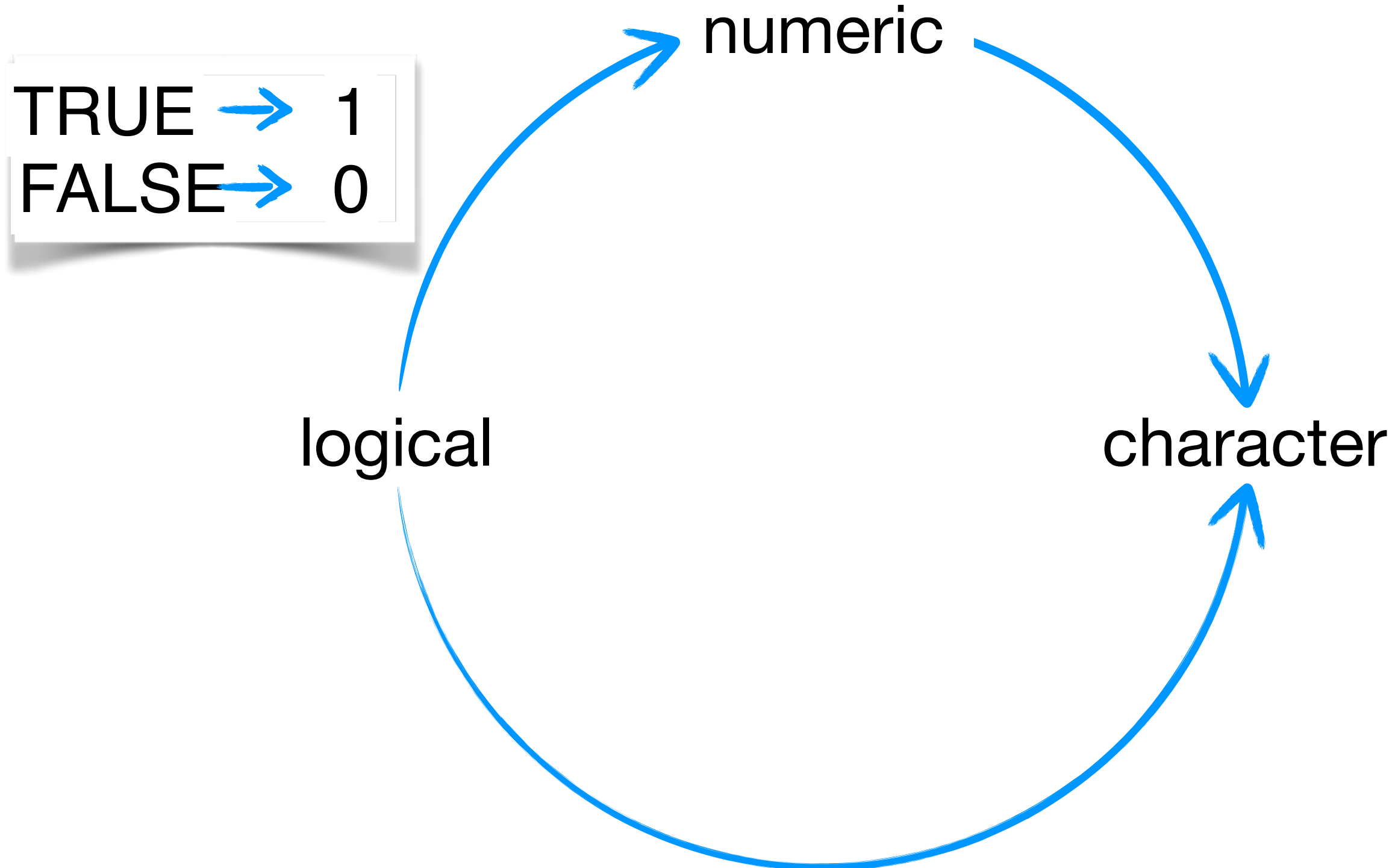
character

Coercion

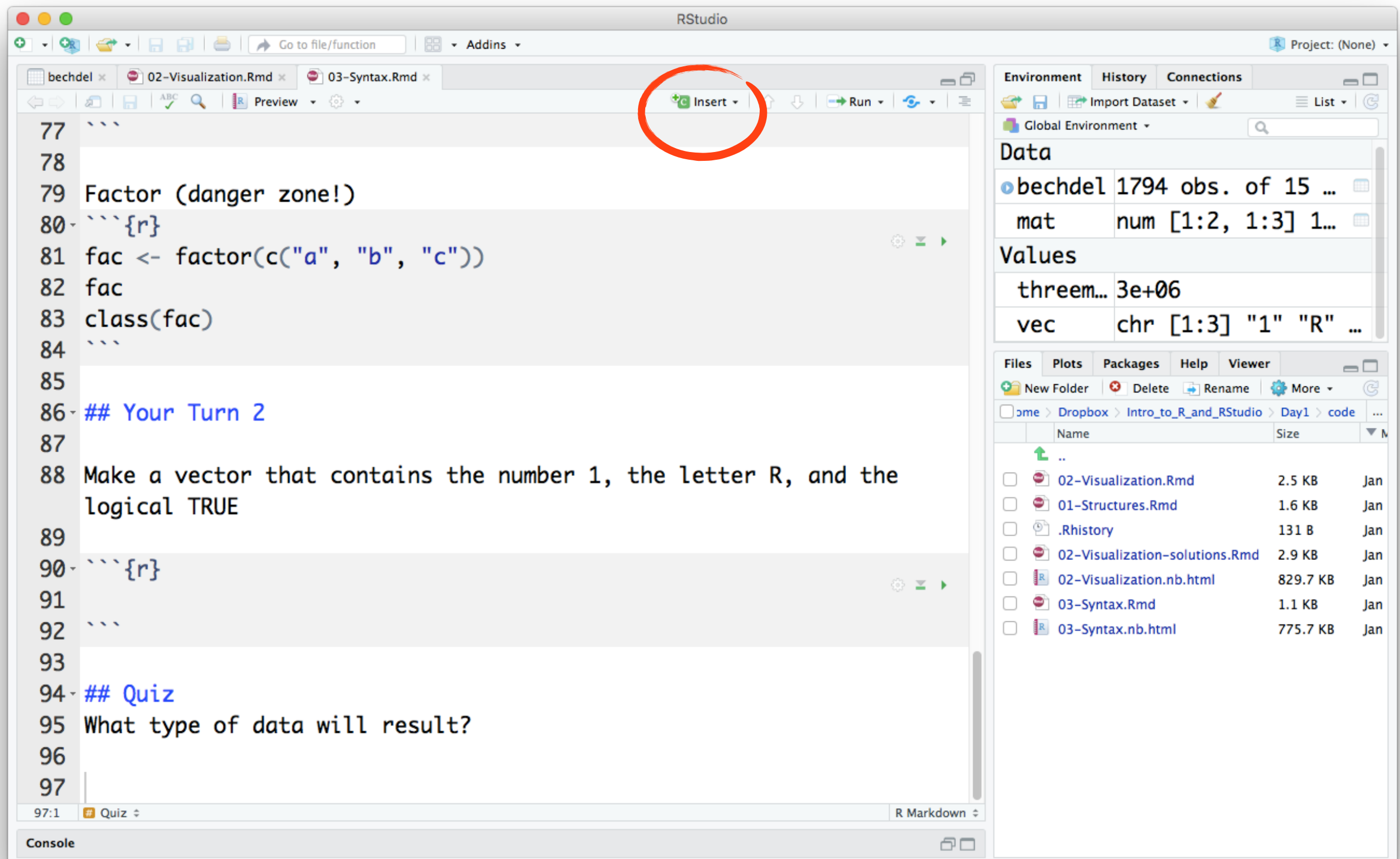
coercion



coercion



I'm going to give you a "quiz", and you might want to create your own chunk to try out some code. Use the Insert button to insert one



The screenshot shows the RStudio interface with the following components:

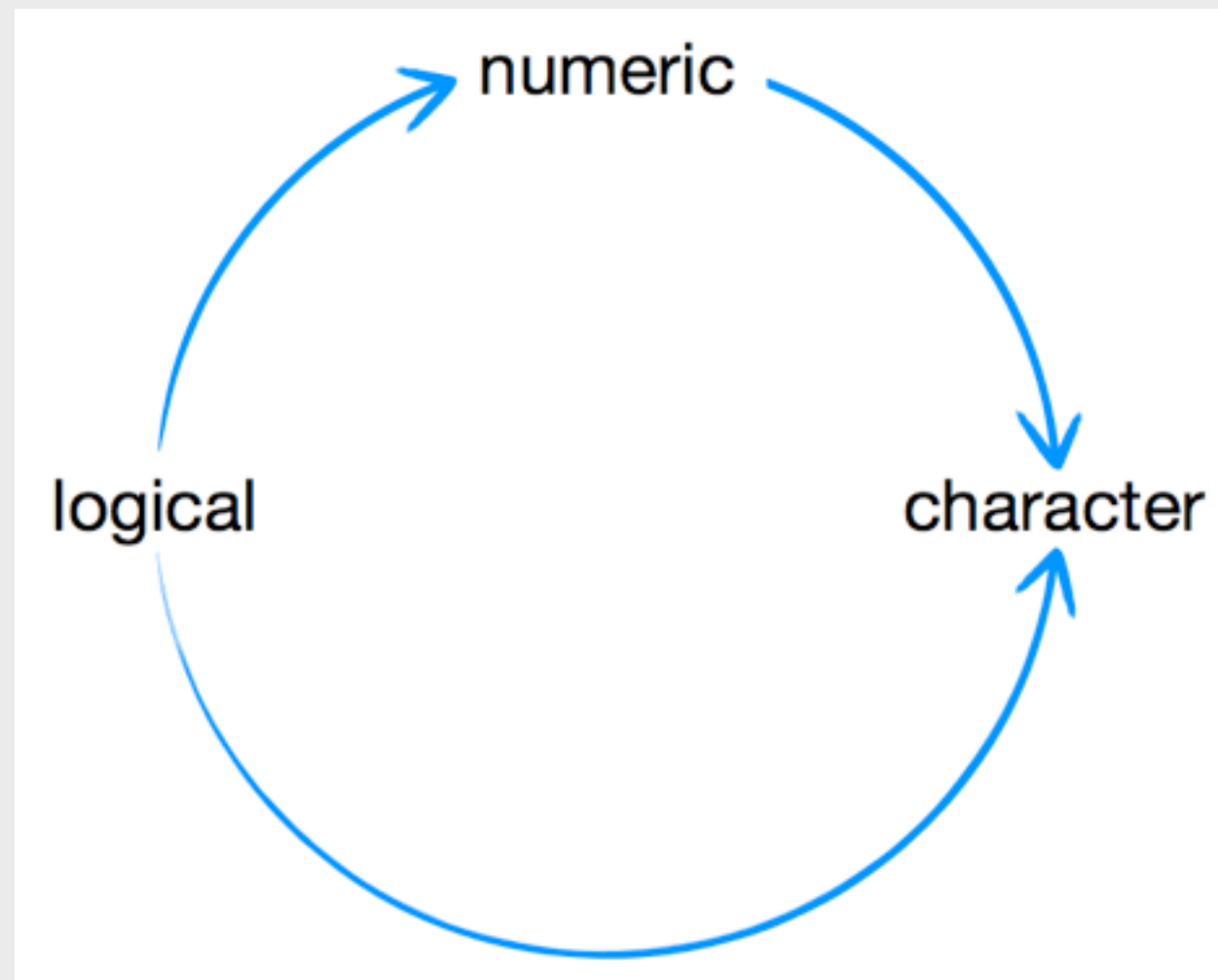
- Code Editor:** Contains R code for creating a factor and a quiz question. The code is as follows:

```
77 ```  
78  
79 Factor (danger zone!)  
80 ```{r}  
81 fac <- factor(c("a", "b", "c"))  
82 fac  
83 class(fac)  
84 ```  
85  
86 ## Your Turn 2  
87  
88 Make a vector that contains the number 1, the letter R, and the  
   logical TRUE  
89  
90 ```{r}  
91  
92 ```  
93  
94 ## Quiz  
95 What type of data will result?  
96  
97
```
- Toolbar:** The 'Insert' button (a green plus icon) is circled in red.
- Environment Panel:** Shows the 'Global Environment' with variables 'bechdel' (1794 obs. of 15 ...) and 'mat' (num [1:2, 1:3] 1...).
- Files Panel:** Shows a list of files in the 'code' directory:

Name	Size	Date
..		
02-Visualization.Rmd	2.5 KB	Jan
01-Structures.Rmd	1.6 KB	Jan
.Rhistory	131 B	Jan
02-Visualization-solutions.Rmd	2.9 KB	Jan
02-Visualization.nb.html	829.7 KB	Jan
03-Syntax.Rmd	1.1 KB	Jan
03-Syntax.nb.html	775.7 KB	Jan

Quiz

What type of data will result?



`c(5, "two")`

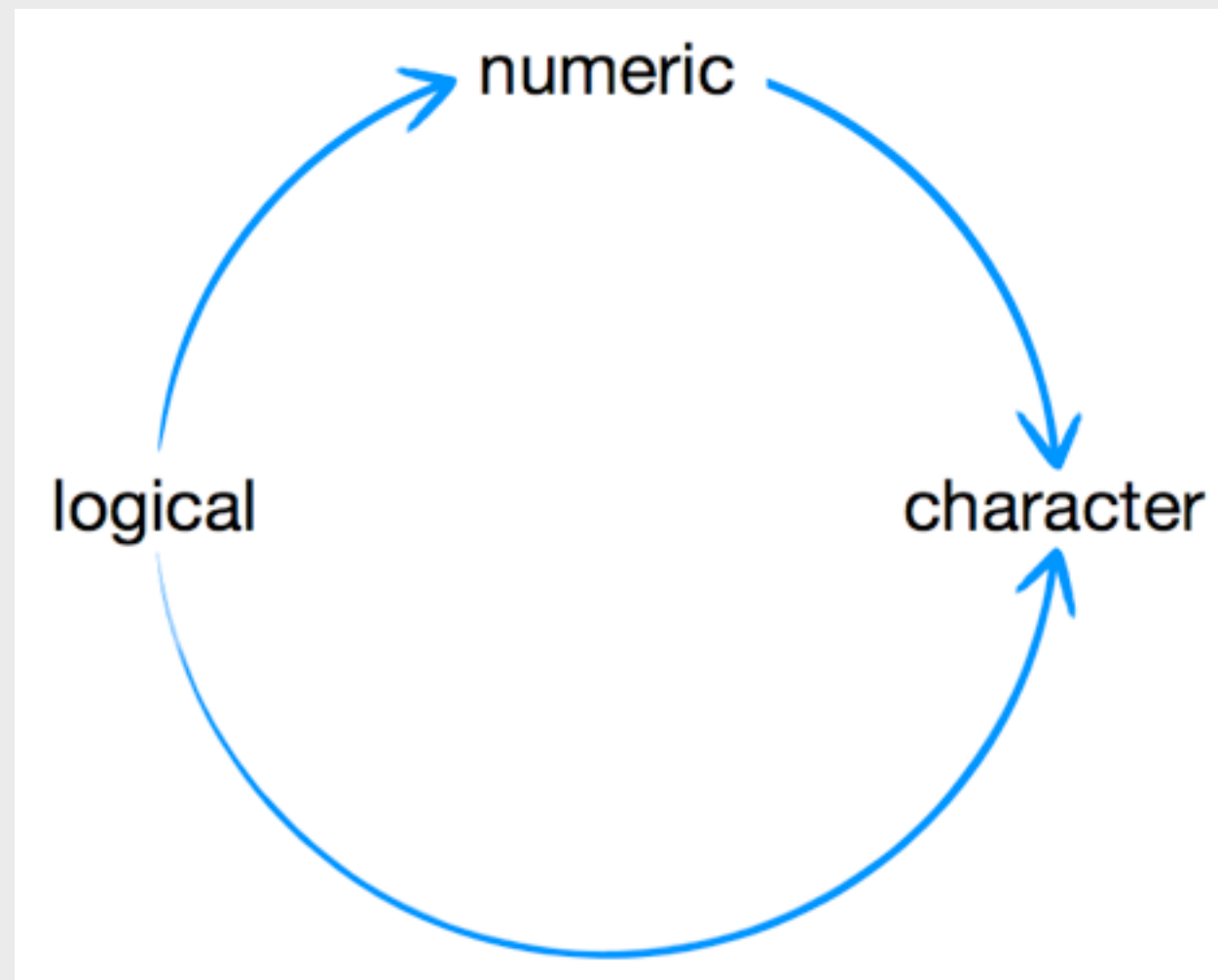
`c(TRUE, "a")`

`c(1, "TRUE")`

`TRUE + 5`

Quiz

What type of data will result?



`c(5, "two")`
character

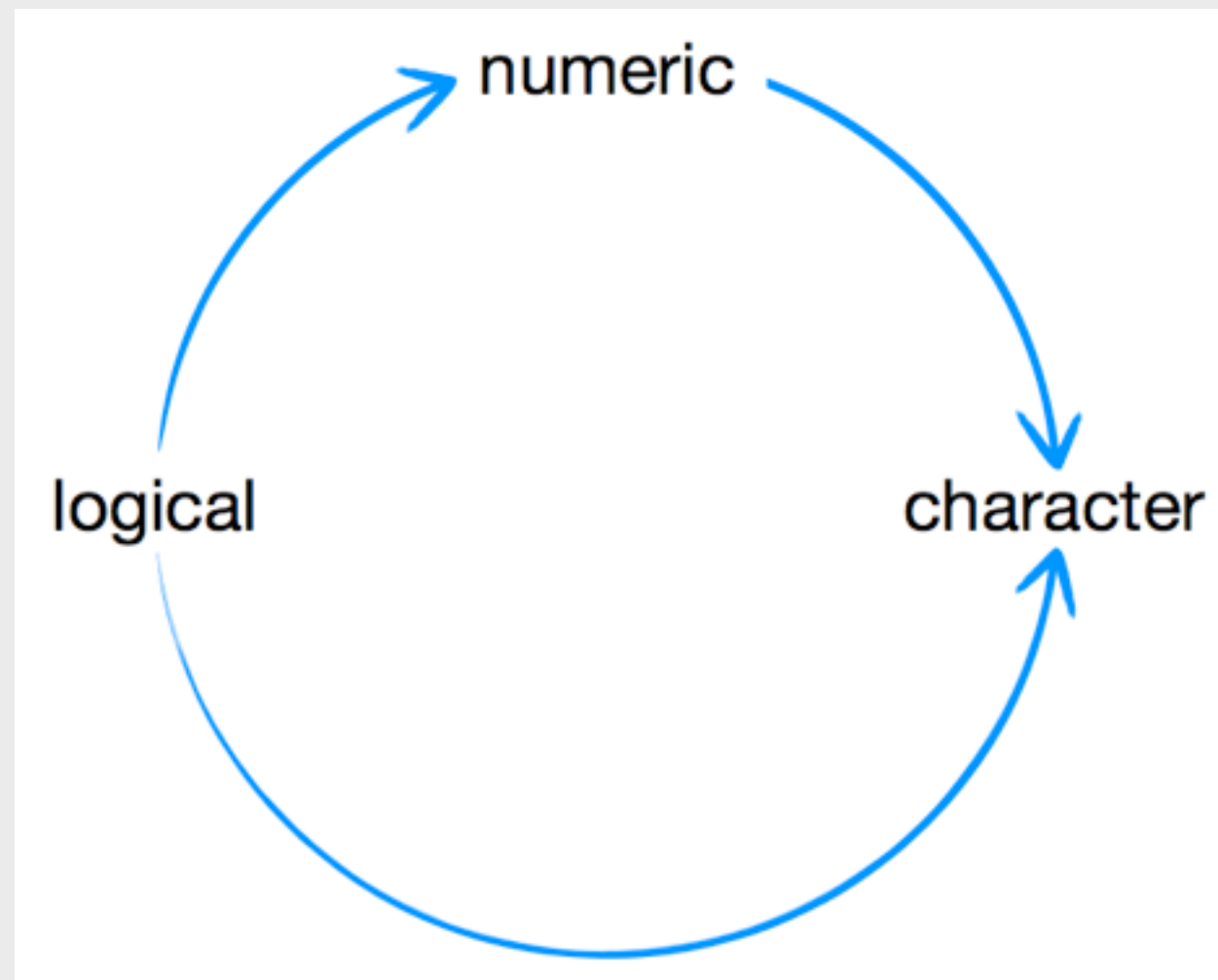
`c(TRUE, "a")`

`c(1, "TRUE")`

`TRUE + 5`

Quiz

What type of data will result?



`c(5, "two")`
character

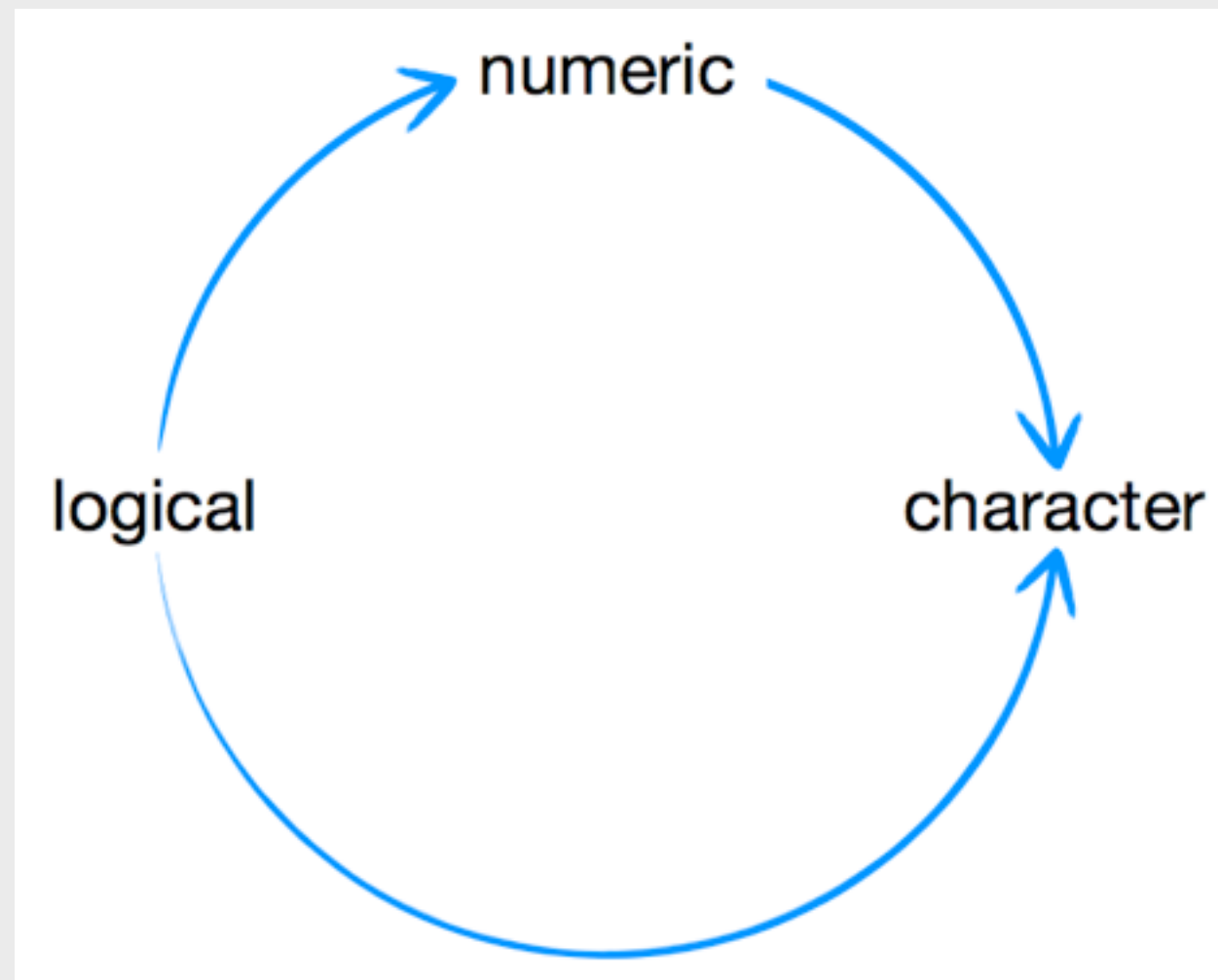
`c(TRUE, "a")`
character

`c(1, "TRUE")`

`TRUE + 5`

Quiz

What type of data will result?



`c(5, "two")`
character

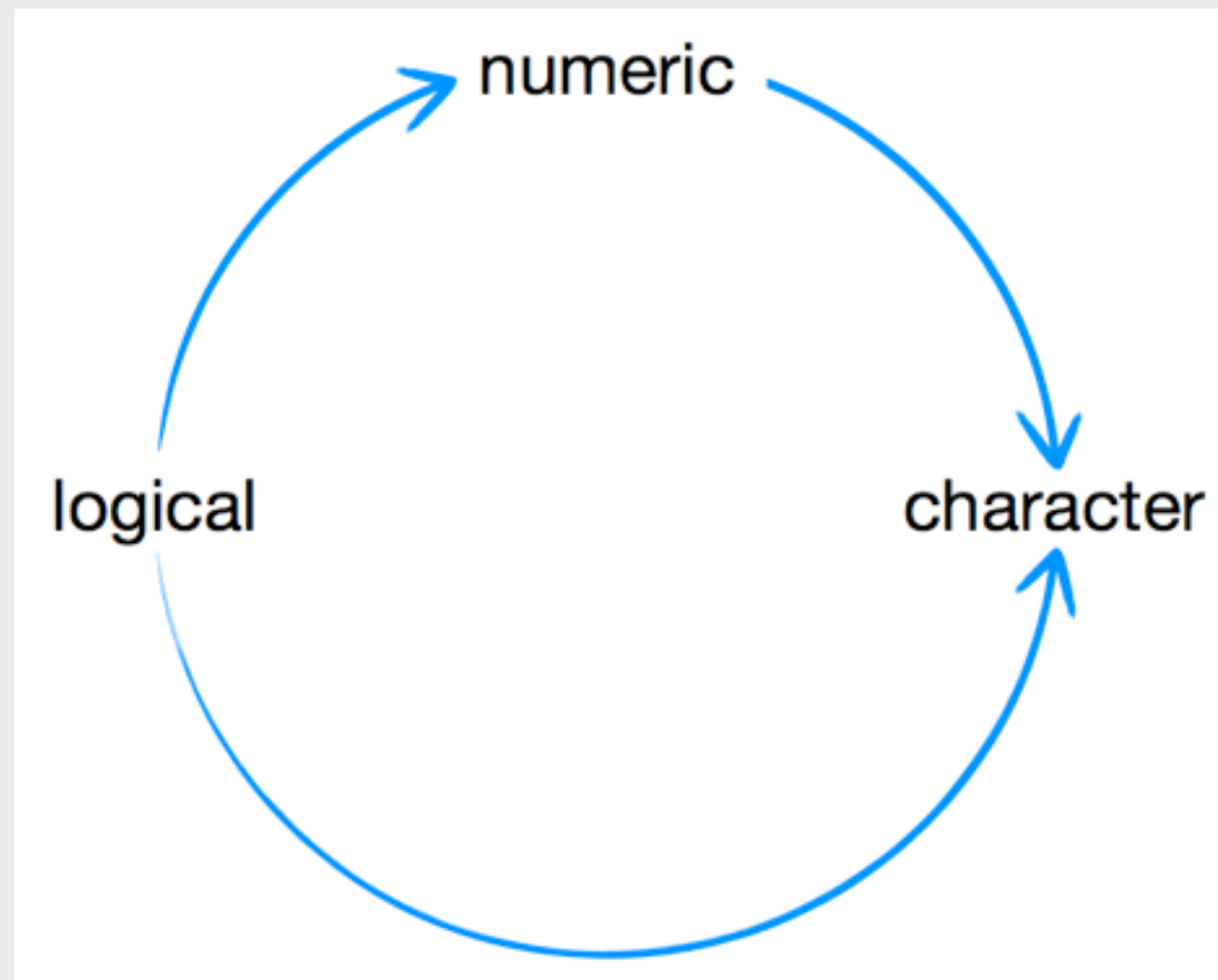
`c(TRUE, "a")`
character

`c(1, "TRUE")`
character

`TRUE + 5`

Quiz

What type of data will result?



`c(5, "two")`
character

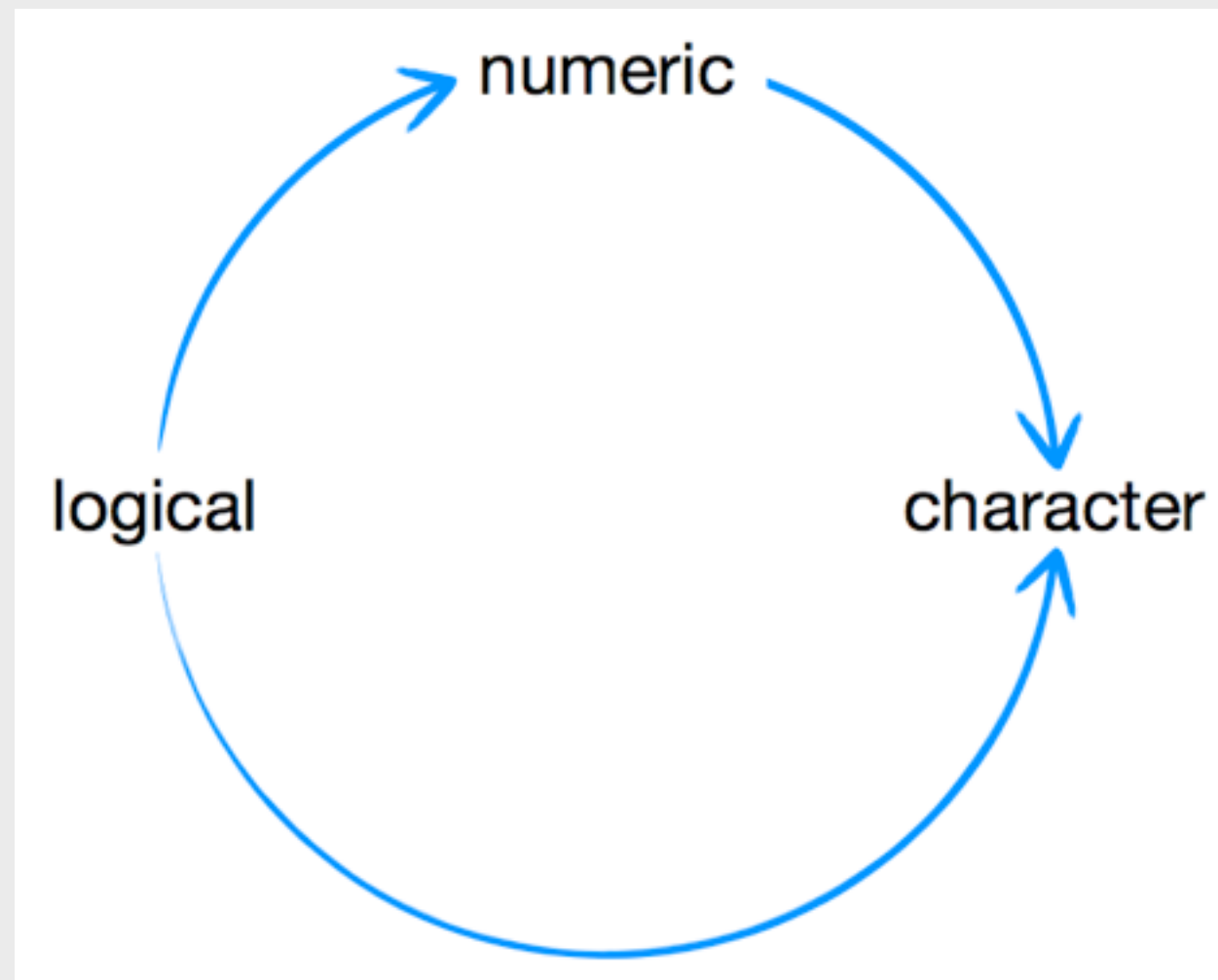
`c(TRUE, "a")`
character

`c(1, "TRUE")`
character

`TRUE + 5`

Quiz

What type of data will result?



`c(5, "two")`
character

`c(TRUE, "a")`
character

`c(1, "TRUE")`
character

`TRUE + 5`
numeric

manual coercion

function	coerces data to
as.numeric	numeric
as.character	character
as.logical	logical
as.factor	factor

`as.numeric("1")`

`as.character(TRUE)`

Matrix

1	"R"	TRUE
2	"S"	FALSE
3	"T"	TRUE

?

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

What if you want different data types in the same object?

Lists and data frames

lists and data frames

lists and *data frames* generalize vectors and matrices to allow multiple types of data

lists

A list is a one dimensional group of R objects.

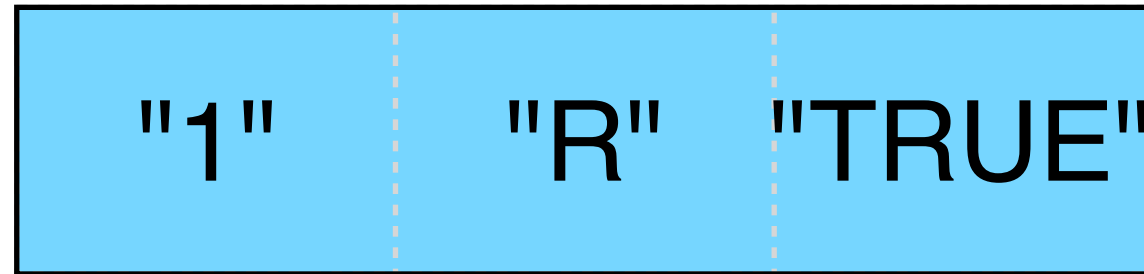
Create lists with `list`

```
lst <- list(1, "R", TRUE)
```

```
class(lst)
```

```
# "list"
```

Vector

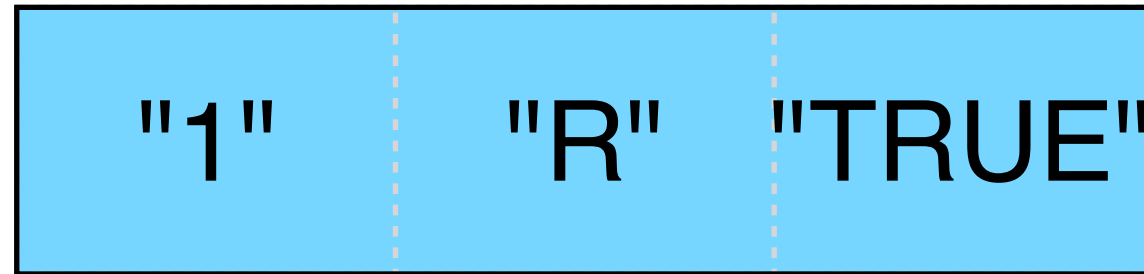


character

List



Vector

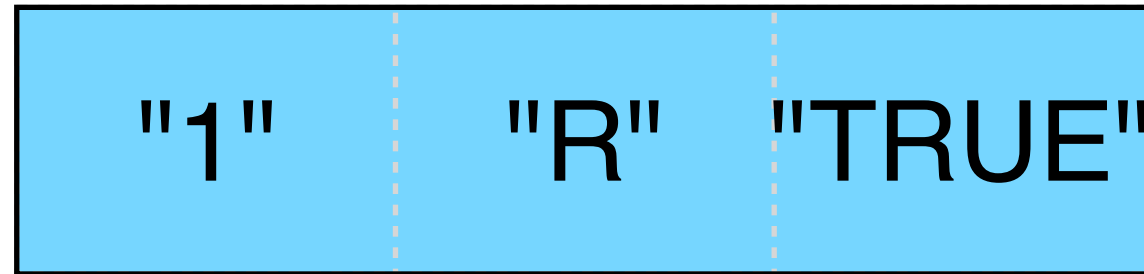


character

List



Vector



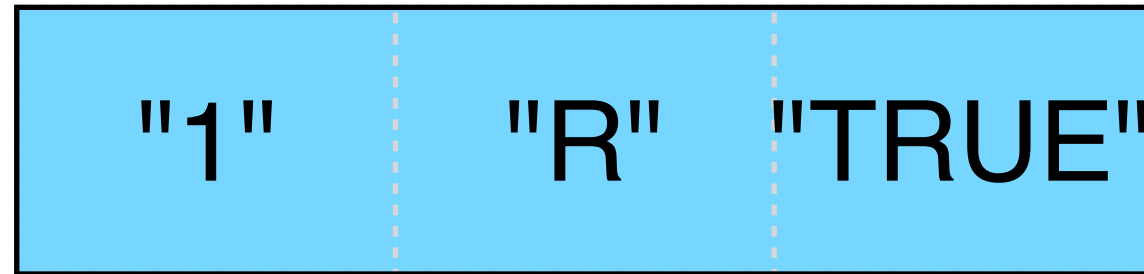
character

List



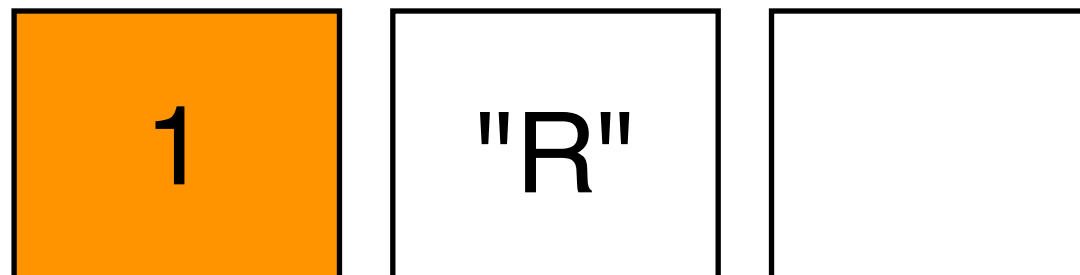
numeric

Vector



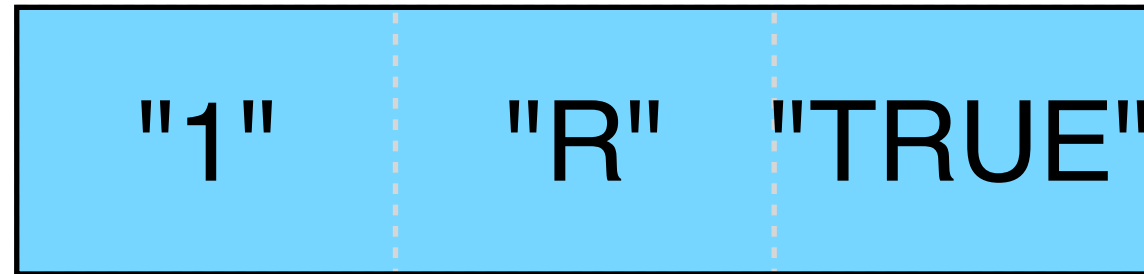
character

List



numeric

Vector



character

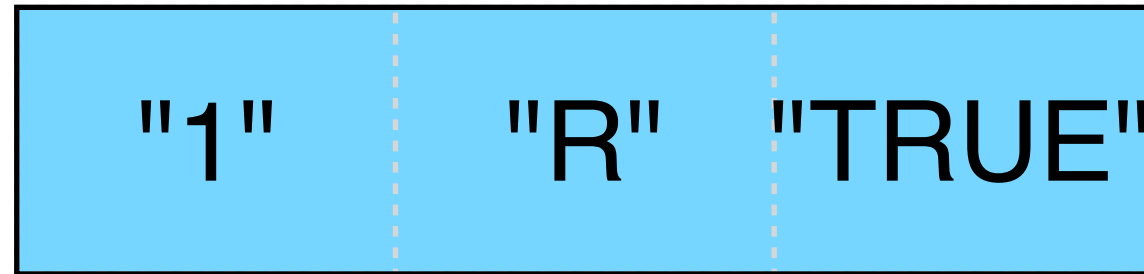
List



numeric

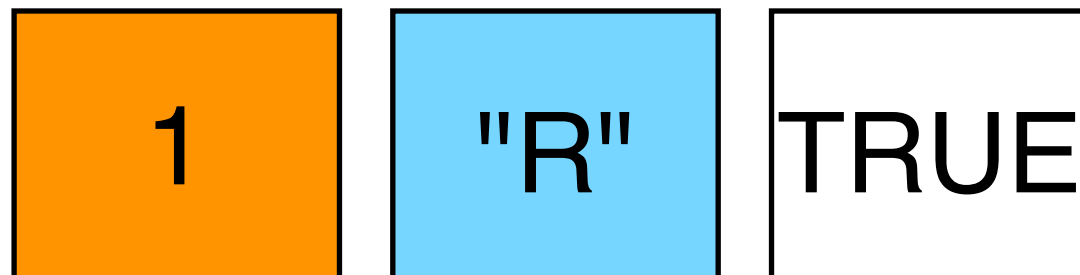
character

Vector



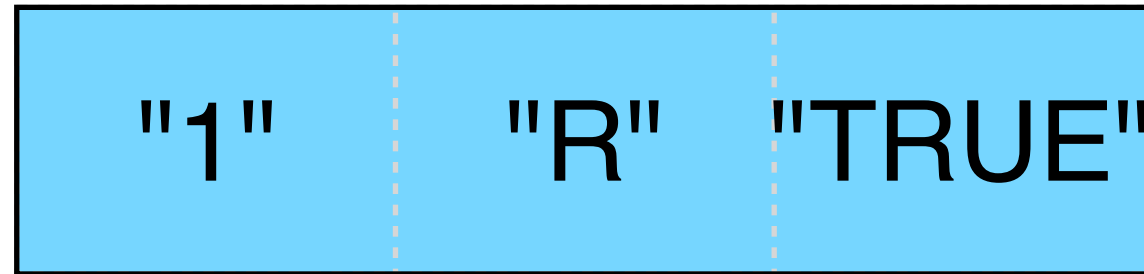
character

List



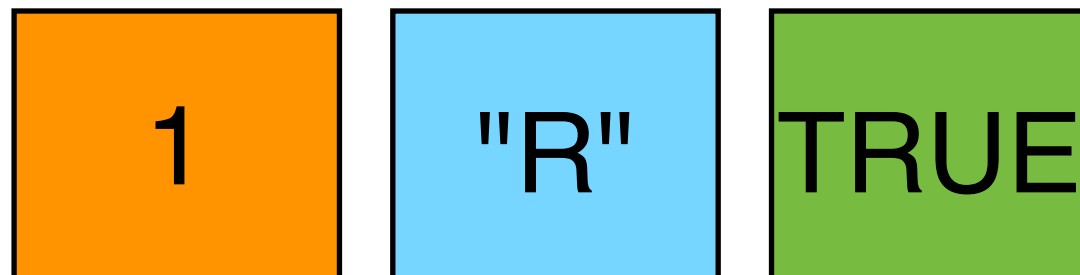
numeric character

Vector



character

List



numeric character logical

The elements of a list can be anything.
Even vectors or other lists.

```
list(c(1, 2), TRUE, c("a", "b", "c"))
```

List

c(1, 2)

TRUE

c("a", "b", "c")

List viewer in RStudio

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for creating a list and a data frame.
- Environment Pane:** Shows the objects created in the Global Environment.
- Files Pane:** Shows the project files and their details.

R Code (Source Editor):

```
94- ## Quiz
95- What type of data will result?
96-
97- ## Lists
98-
99- ```{r}
100- lst <- list(1, "R", TRUE)
101- class(lst)
102- ```
103-
104- ## Data frames
105- ```{r}
106- df <- data.frame(c(1, 2, 3),
107-   c("R","S","T"), c(TRUE, FALSE, TRUE))
108- class(df)
109- ```
110-
111- ## Naming
112- ```{r}
113- nvec <- c(one = 1, two = 2, three = 3)
114- nlst <- list(one = 1, two = 2, many = c(3, 4, 5))
115- ```
```

Environment Pane:

Global Environment

Object	Description
bechdel	1794 obs. of 15 variables
df	3 obs. of 3 variables
lst	List of 3
: num 1	
: chr "R"	
: logi TRUE	
mat	num [1:2, 1:3] 1 2 3 4 5 6
nlst	List of 3

Files Pane:

Name	Size	Modified
..		
02-Visualization.Rmd	2.5 KB	Jan 30, 2018, 5:33 PM
01-Structures.Rmd	1.6 KB	Jan 30, 2018, 5:11 PM
.Rhistory	131 B	Jan 28, 2018, 4:36 PM
02-Visualization-solutions.Rmd	2.9 KB	Jan 30, 2018, 5:11 PM
02-Visualization.nb.html	829.7 KB	Jan 30, 2018, 5:33 PM
03-Syntax.Rmd	1.3 KB	Jan 30, 2018, 6:56 PM

data frame

A data frame is a two dimensional group of R objects.

Each column in a data frame can be a different type

```
df <- data.frame(c(1, 2, 3),  
  c("R","S","T"), c(TRUE, FALSE, TRUE))  
class(df)  
# "data.frame"
```


Your turn

We've already seen a data frame today. What was it called? What kinds of data were in it?

A digital timer with a black background and white LED-style digits. The display shows the time 00:30, with a colon separating the minutes and seconds. The digits are slightly shadowed, giving them a 3D appearance.

00:30

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1		
2		
3		

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1		
2		
3		

numeric

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1	"R"	
2	"S"	
3	"T"	

numeric

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1	"R"	
2	"S"	
3	"T"	

numeric

character

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1	"R"	TRUE
2	"S"	FALSE
3	"T"	TRUE

numeric

character

Matrix

"1"	"R"	"TRUE"
"2"	"S"	"FALSE"
"3"	"T"	"TRUE"

character

data frame

1	"R"	TRUE
2	"S"	FALSE
3	"T"	TRUE

numeric

character

logical

names

You can name the elements of a vector, list, or data frame when you create them.

```
nvec <- c(one = 1, two = 2, three = 3)
```

```
nvec
```

```
# one two three
```

```
#  1  2  3
```



```
nlst <- list(one = 1, two = 2,  
            many = c(3, 4, 5))
```

```
nlst
```

```
# $one
```

```
# [1] 1
```

```
#
```

```
# $two
```

```
# [1] 2
```

```
#
```

```
# $many
```

```
# [1] 3 4 5
```

```
ndf <- data.frame(numbers = c(1, 2, 3),  
  letters = c("R", "S", "T"),  
  logic = c(TRUE, FALSE, TRUE))
```

```
ndf  
#  numbers letters logic  
# 1      1      R TRUE  
# 2      2      S FALSE  
# 3      3      T TRUE
```

Your turn

Use the RStudio data preview to
compare `df` and `ndf`

A digital timer with a dark grey background and white, segmented digits. The display shows the time 00:30.

00:30

 Go to file/function  Addins

bechdel x 02-Visualization.Rmd x 03-Syntax.Rmd* x ndf x df x

 Filter

	numbers	letters	logic
1	1	R	TRUE
2	2	S	FALSE
3	3	T	TRUE

Showing 1 to 3 of 3 entries

Console Terminal x R Markdown x

~/Dropbox/Intro_to_R_and_RStudio/Day1/code/

```
> df <- data.frame(c(1, 2, 3),
```

You can also see the names with names

```
names(ndf)
```

```
# [1] "numbers" "letters" "logic"
```

```
names(nvec)
```

```
# [1] "one" "two" "three"
```

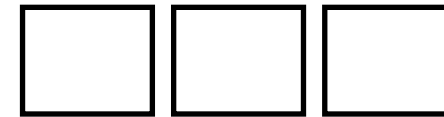
single type

multiple types

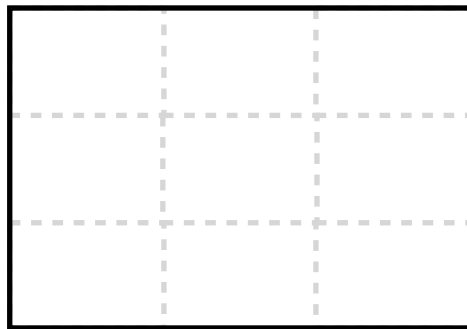
Vector



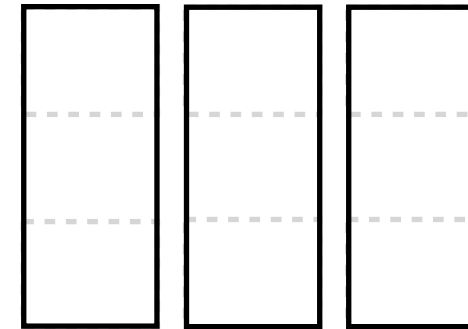
List



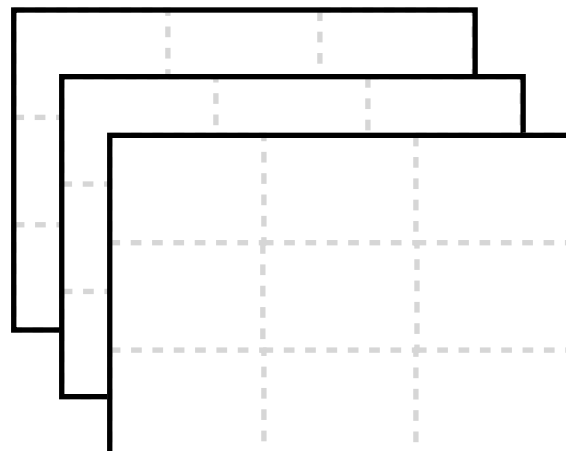
Matrix



Data frame



Array



1D

2D

nD

How R makes a data frame

List

```
c("a","b","c","d")
```

```
c(1, 2, 3, 4)
```

```
c(T, F, T, F)
```

List

c(
"a", "
b", "c
", "d")

c(
1,
2,
3,
4)

c(
T,
F,
T,
F)

List

```
c(  
  "a", "  
  b", "c  
  ", "d")
```

```
c(  
  1,  
  2,  
  3,  
  4)
```

```
c(  
  T,  
  F,  
  T,  
  F)
```

~~List~~

data frame

```
c(
  "a", "
b", "c
", "d")
```

```
c(
  1,
  2,
  3,
  4)
```

```
c(
  T,
  F,
  T,
  F)
```

helper functions for data structures

	create	change to	check	get names	get dimensions
vector	c, vector	as.vector	is.vector	names	length
matrix	matrix	as.matrix	is.matrix	rownames, colnames	dim, nrow, ncol
array	array	as.array	is.array	dimnames	dim
list	list	as.list	is.list	names	length
data frame	data.frame	as.data.frame	is.data.frame	names	dim, nrow, ncol