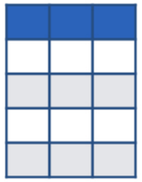
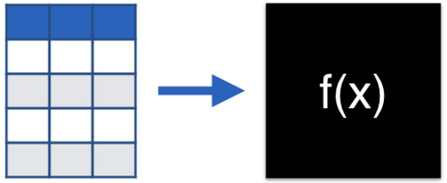
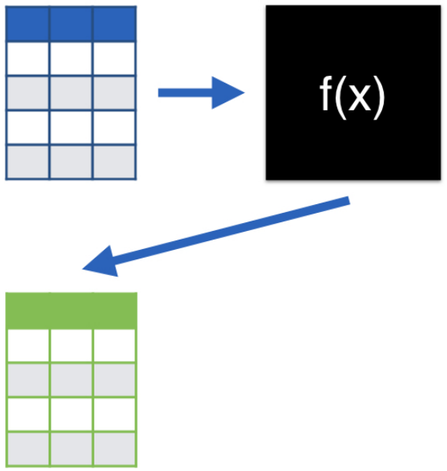
What is Object-Oriented Programming?

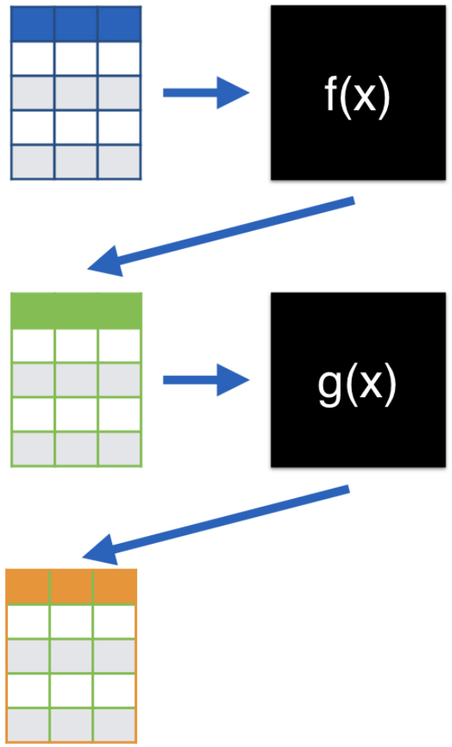
OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

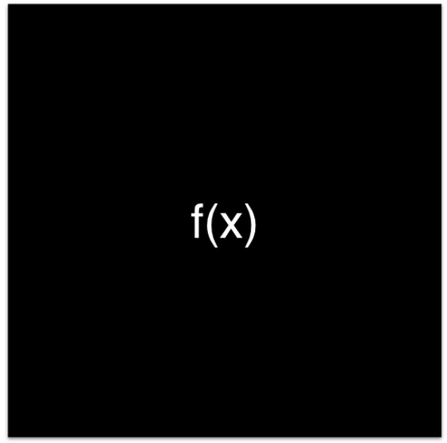
Richie Cotton
Data Evangelist at DataCamp




f(x)

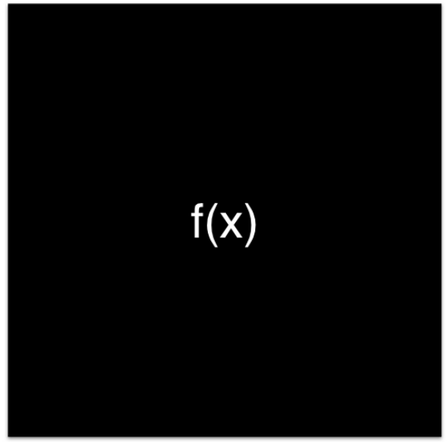

f(x)


f(x)
g(x)

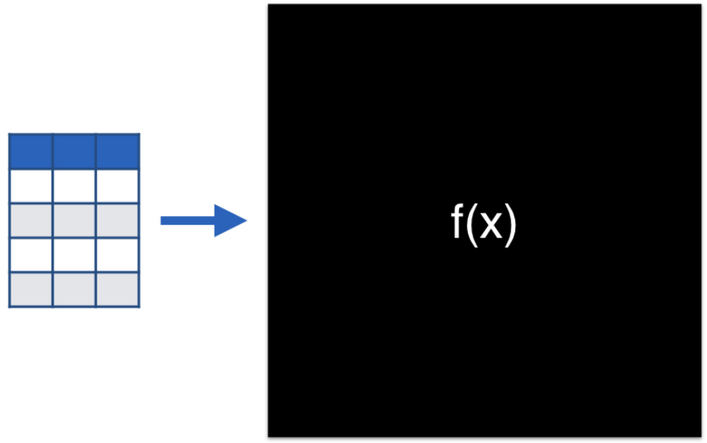



```
                    <- function(        ) {



}
```

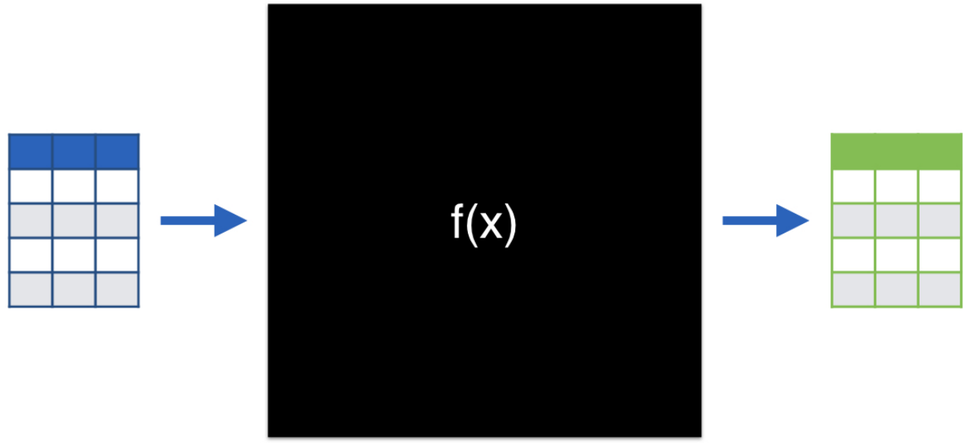
function



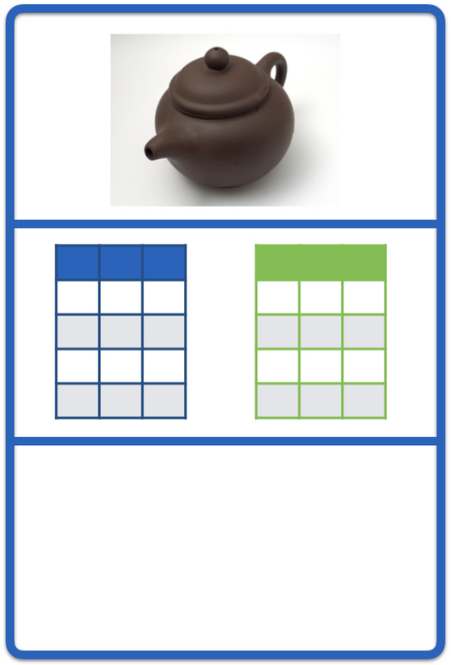

```
calculate_something  <- function(       ) {
  # do something

}
```

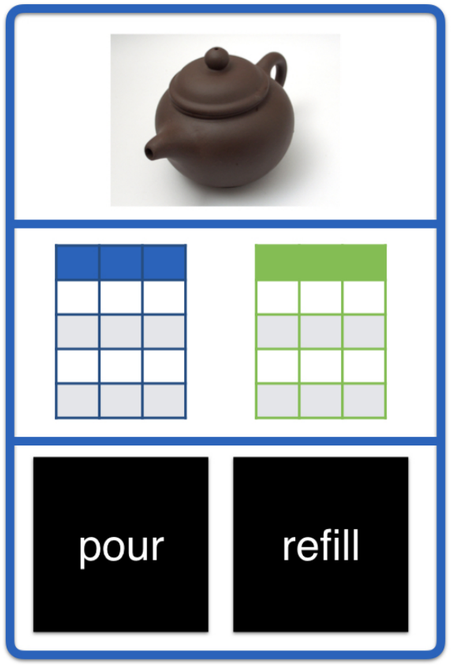



```
calculate_something  <- function(x,  y, z) {
  # do something

}
```




```
calculate_something  <- function(x,  y, z) {
  # do something
  return(the_result)
}
```



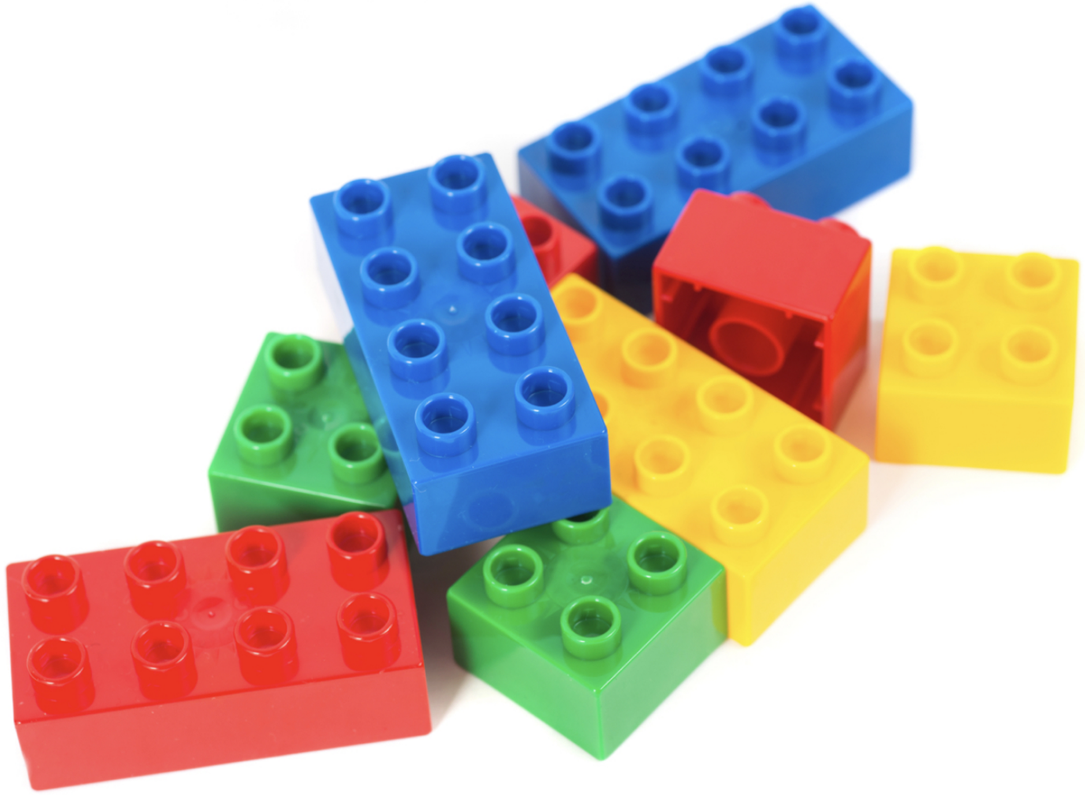

pour
refill

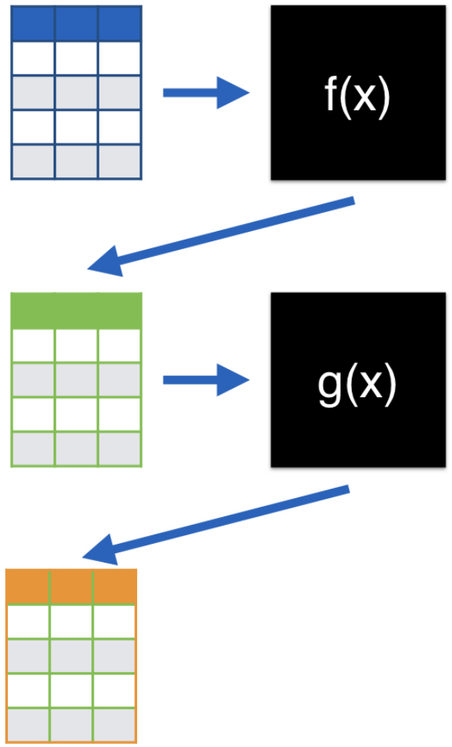
A method is just a function, talked about in an OOP context

logical vector	closure function
integer vector	builtin function
numeric vector	special function
complex vector	environment
character vector	null
raw vector	formula
list	expression
matrix	call
array	pairlist
data.frame	external pointer
factor	

list

environment




f(x)
g(x)

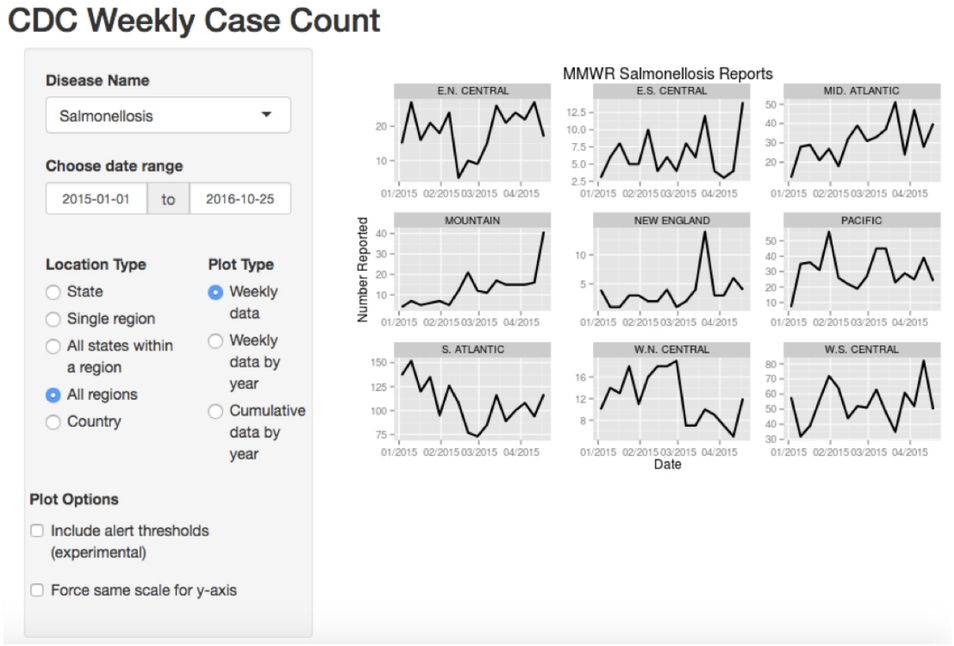

tidyverse
www.rstudio.com


dplyr
dplyr.tidyverse.org

When is OOP a good idea?


Bioconductor
OPEN SOURCE SOFTWARE FOR BIOINFORMATICS


WIKIPEDIA


CDC Weekly Case Count
Disease Name
Salmonellosis
Choose date range
2015-01-01
to
2016-10-25
Location Type
State
Single region
All states within a region
All regions
Country
Plot Type
Weekly data
Weekly data by year
Cumulative data by year
Plot Options
Include alert thresholds (experimental)
Force same scale for y-axis
MMWR Salmonellosis Reports
E.N. CENTRAL
E.S. CENTRAL
MID. ATLANTIC
MOUNTAIN
NEW ENGLAND
PACIFIC
S. ATLANTIC
W.N. CENTRAL
W.S. CENTRAL
Number Reported
Date
01/2015
02/2015
03/2015
04/2015

building tools

use object-oriented programming

analyzing data

use functional programming

Summary

- With **functional programming**, think about the **functions first.**
- With **object-oriented programming** (OOP) think about the **data structures first.**
- **Don't** use OOP for **general purpose data analyses.**
- **Do** use OOP when you have a **limited number of complex objects.**

Let's practice!

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

The Nine Systems

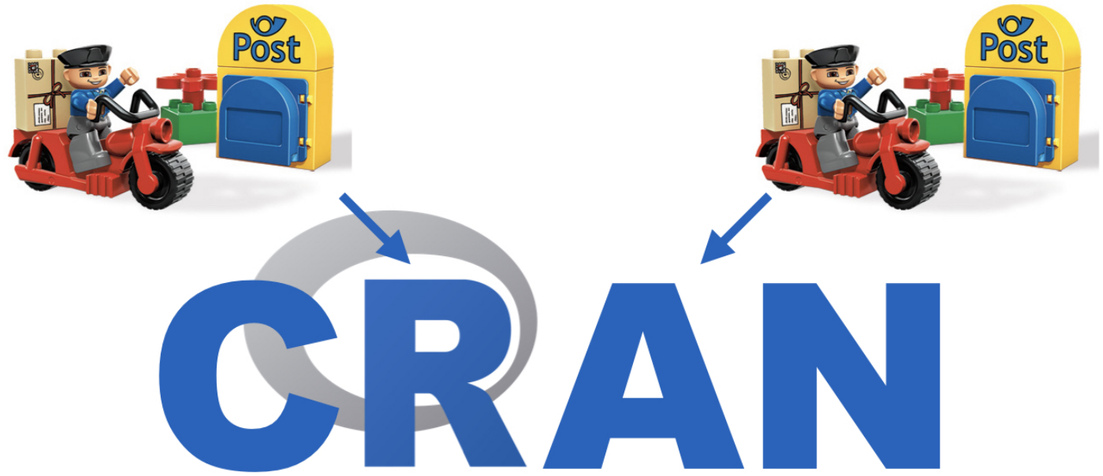
OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

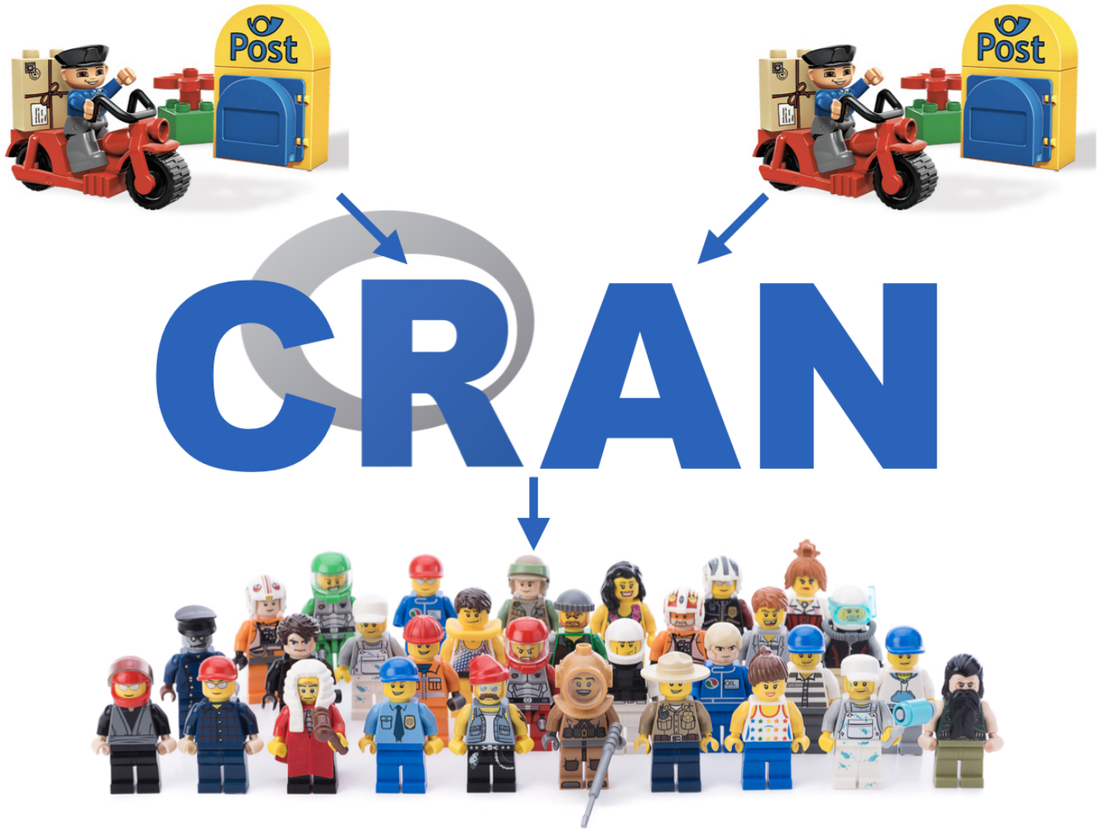
Richie Cotton
Data Evangelist at DataCamp

datacamp


Post


Post


Post
Post
CRAN


Post
Post
CRAN

ReferenceClasses

R.oo

S4

OOP

R5

S3

R6

proto

mutatr

ReferenceClasses

R.oo

S4

OOP

S3

R6

proto

ReferenceClasses

R.oo

S4

S3

R6

proto

ReferenceClasses

R.oo

S4

S3

R6

ReferenceClasses

S4

S3

R6

ReferenceClasses

S4

R6

ReferenceClasses


ReferenceClasses
S4
S3
R6


ReferenceClasses
S4
S3
R6

Summary

- Use **S3 regularly**
- Use **R6** when you need **more power**
- Use **S4** for **Bioconductor**
- **Maybe** use **ReferenceClasses**

Let's practice!

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

How does R Distinguish Variables?

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

Richie Cotton
Data Evangelist at DataCamp

str()

```
str(sleep)
```

```
'data.frame':    20 obs. of  3 variables:
 $ extra: num  0.7 -1.6 -0.2 -1.2 -0.1 ...
 $ group: Factor w/ 2 levels "1","2": 1 1 1 1 1 ...
 $ ID   : Factor w/ 10 levels "1","2","3","4",..: 1 2..
```

```
class(sleep)
```

```
"data.frame"
```

int_mat

```
(int_mat <- matrix(1:12, 3))
```

```
     [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
[3,]    3    6    9   12
```

```
class(int_mat)
```

```
"matrix"
```

```
typeof(int_mat)
```

```
"integer"
```

num_mat

```
(num_mat <- matrix(rnorm(12), 3))
```

```
           [,1]       [,2]        [,3]       [,4]
[1,] -0.2911535 -0.1139933 -0.71290868  0.8640191
[2,] -2.2266419 -1.3604316 -1.90716974  0.4012884
[3,] -0.7504663 -1.2478873  0.01104117 -0.8127333
```

```
class(num_mat)
```

```
"matrix"
```

```
typeof(num_mat)
```

```
"double"
```

mode()

storage.mode()

Summary

- `class()` is your **first choice** for determining the kind of variable
- `typeof()` is also **occasionally useful**
- `mode()` and `storage.mode()` are old functions; **don't use them**

Let's practice!

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

Assigning Classes

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R

Richie Cotton
Data Evangelist at DataCamp

datacamp

```
(x <- rexp(10))
```

```
0.195051 2.191040 0.498703 0.976122 0.299001
0.105187 0.090073 2.328233 3.043201 2.129631
```

```
class(x) <- "random_numbers"
```

```
x
```

```
0.195051 2.191040 0.498703 0.976122 0.299001
0.105187 0.090073 2.328233 3.043201 2.129631
```

```
attr(,"class")
```

```
class(x)
```

```
"random_numbers"
```

```
typeof(x)
```

```
"double"
```

```
is.numeric(x)
```

```
TRUE
```

```
length(x)
```

```
10
```

```
mean(x)
```

```
1.1856
```

Summary

- You can **override** the `class()`
- This **won't** break existing functionality

Let's practice!

OBJECT-ORIENTED PROGRAMMING WITH S3 AND R6 IN R