



INTRODUCTION TO R

Create and Name Vectors

Vector

- Sequence of data elements
- Same basic type
- `character`, `numeric`, `logical`

Create a vector `c()`

```
> c("hearts", "spades", "diamonds", "diamonds", "spades")
[1] "hearts"      "spades"      "diamonds"    "diamonds"    "spades"

> drawn_suits <- c("hearts", "spades", "diamonds",
                  "diamonds", "spades")

> drawn_suits
[1] "hearts"      "spades"      "diamonds"    "diamonds"    "spades"

> is.vector(drawn_suits)
[1] TRUE
```

Create a vector `c()`

```
> remain <- c(11, 12, 11, 13)
> remain
[1] 11 12 11 13
```

Name a vector

names()

```
> remain <- c(11, 12, 11, 13)
> remain
[1] 11 12 11 13

> suits <- c("spades", "hearts", "diamonds", "clubs")
> names(remain) <- suits
> remain
  spades    hearts diamonds    clubs
      11         12        11      13

> remain <- c(spades = 11, hearts = 12,
              diamonds = 11, clubs = 13)

> remain <- c("spades" = 11, "hearts" = 12,
              "diamonds" = 11, "clubs" = 13)
```

Name a vector

names()

```
> remain <- c(11, 12, 11, 13)
> suits <- c("spades", "hearts", "diamonds", "clubs")
> names(remain) <- suits
```

option 1

```
> remain <- c(spades = 11, hearts = 12,
              diamonds = 11, clubs = 13)
```

option 2

```
> remain <- c("spades" = 11, "hearts" = 12,
              "diamonds" = 11, "clubs" = 13)
```

option 3

```
> str(remain)
Named num [1:4] 11 12 11 13
- attr(*, "names")= chr [1:4] "spades" "hearts"
                           "diamonds" "clubs"
```

Single value = vector

```
> my_apples <- 5
> my_oranges <- "six"

> is.vector(my_apples)
[1] TRUE
> is.vector(my_oranges)
[1] TRUE

> length(my_apples)
[1] 1
> length(my_oranges)
[1] 1

> length(drawn_suits)
[1] 5
```

Vectors are homogeneous

- Only elements of the same type
- Atomic vectors <> lists
- Automatic coercion if necessary

Coercion for vectors

```
> drawn_ranks <- c(7, 4, "A", 10, "K", 3, 2, "Q")
```

```
> drawn_ranks  
[1] "7"  "4"  "A"  "10" "K"  "3"  "2"  "Q"
```

```
> class(drawn_ranks)  
[1] "character"
```



INTRODUCTION TO R

Vector Arithmetic

Vector Arithmetic

```
> my_apples <- 5
> my_oranges <- 6
> my_apples + my_oranges
[1] 11
```

my_apples is a vector!
my_oranges is a vector!

Computations are performed **element-wise**

```
> earnings <- c(50, 100, 30)

> earnings * 3
[1] 150 300 90
```

Vector Arithmetic

```
> earnings/10  
[1] 5 10 3  
  
> earnings - 20  
[1] 30 80 10  
  
> earnings + 100  
[1] 150 200 130  
  
> earnings^2  
[1] 2500 10000 900
```

Mathematics naturally extend!

Element-wise

```
> earnings <- c(50, 100, 30)
> expenses <- c(30, 40, 80)
```

```
> earnings - expenses
[1] 20 60 -50
```

```
> earnings + c(10, 20, 30)
[1] 60 120 60
```

```
> earnings * c(1, 2, 3)
[1] 50 200 90
```

**multiplication and division
are done element-wise!**

```
> earnings / c(1, 2, 3)
[1] 50 50 10
```

sum() and >

```
> earnings <- c(50, 100, 30)
> expenses <- c(30, 40, 80)

> bank <- earnings - expenses
> bank
[1] 20 60 -50

> sum(bank)
[1] 30

> earnings > expenses
[1] TRUE TRUE FALSE
```



INTRODUCTION TO R

Subsetting Vectors

Subset by index

```
> remain <- c(spades = 11, hearts = 12,  
              diamonds = 11, clubs = 13)
```

[1] -> take element at index 1

```
> remain[1]  
spades  
11
```

result is a (named) vector too!

```
> remain[3]  
diamonds  
11
```


Subset by name

```
> remain <- c(spades = 11, hearts = 12,  
              diamonds = 11, clubs = 13)
```

```
> remain["spades"]  
spades  
      11
```

```
> remain["diamonds"]  
diamonds  
       11
```

Subset multiple elements

```
> remain <- c(spades = 11, hearts = 12,  
              diamonds = 11, clubs = 13)
```

```
> remain_black <- remain[c(1, 4)]
```

```
> remain_black  
spades  clubs  
    11    13
```

order in selection vector matters!

```
> remain[c(4, 1)]
```

```
clubs spades  
    13    11
```

```
> remain[c("clubs", "spades")]
```

```
clubs spades  
    13    11
```

Subset all but some

```
> remain <- c(spades = 11, hearts = 12,  
              diamonds = 11, clubs = 13)
```

```
> remain[-1]
```

hearts	diamonds	clubs
12	11	13

All but index 1 are returned

```
> remain[-c(1, 2)]
```

diamonds	clubs
11	13

```
> remain[-"spades"]
```

Error in `-"spades"` : invalid argument to unary operator

Subset using logical vector

```
> remain <- c(spades = 11, hearts = 12,  
              diamonds = 11, clubs = 13)  
  
> remain[c(FALSE, TRUE, FALSE, TRUE)]  
hearts  clubs  
    12    13  
  
> selection_vector <- c(FALSE, TRUE, FALSE, TRUE)  
> remain[selection_vector]  
hearts  clubs  
    12    13
```

Subset using logical vector

```
> remain <- c(spades = 11, hearts = 12,
              diamonds = 11, clubs = 13)

> remain[c(TRUE, FALSE)]
  spades diamonds
      11      11

> remain[c(TRUE, FALSE, TRUE, FALSE)]
  spades diamonds
      11      11

> remain[c(TRUE, FALSE, TRUE)]
  spades diamonds clubs
      11      11    13

> remain[c(TRUE, FALSE, TRUE, TRUE)]
  spades diamonds clubs
      11      11    13
```

R recycles c(T, F) to c(T, F, T, F)