



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) option 1
> suits <- c("spades", "hearts", "diamonds", "clubs")
> names(remain) <- suits

> remain <- c(spades = 11, hearts = 12, option 2
             diamonds = 11, clubs = 13)

> remain <- c("spades" = 11, "hearts" = 12, option 3
             "diamonds" = 11, "clubs" = 13)

> 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  $\langle \rangle$  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

**Let's practice!**