WARM-UPS

Enough chit-chatting, time to code!

PRIMARY DATA TYPES

What are the primary data types most often used in data analysis and in R?

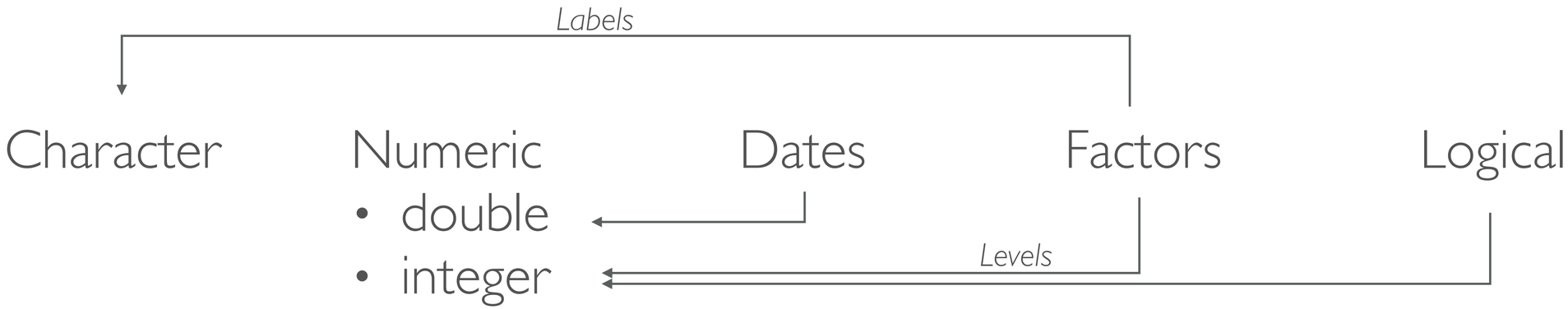
PRIMARY DATA TYPES

What are the primary data types most often used in data analysis and in R?

Character	Numeric	Dates	Factors	Logical
	• double • integer			

PRIMARY DATA TYPES

What are the primary data types most often used in data analysis and in R?



DATA TYPE EXERCISES

Create these vectors:

- `dbl_var <- c(1, 2.5, 4.5, NaN, Inf)`
- `int_var <- c(1L, 6L, 10L, NA)`
- `log_var <- c(TRUE, FALSE, T, F)`
- `chr_var <- c(“these are”, “some strings”)`
- `fct_var <- factor(c("male", "female", "male", "female"))`

Brainstorm with your neighbors all the ways to test for the type of contents in these vectors

SOLUTIONS

```
typeof        -> be careful, what happens when you apply this to a factor?
class         -> be careful, what happens when you apply this to a double?
is.numeric
is.double
is.integer
is.logical
is.character
is.factor

# Bonus
is.na
is.nan
is.infinite
```

Plenty more, just type **`is.`** *+ tab for a full list*

COERCION EXERCISES

Discuss with your neighbor …what happens when you do the following? Use your mind not your computer to answer:

- `typeof(c(int_var, dbl_var))`
- `typeof(c(int_var, log_var))`
- `typeof(c(log_var, chr_var))`
- `typeof(c(chr_var, int_var))`
- `typeof(c(chr_var, fct_var))`

SOLUTIONS

```
> typeof(c(int_var, dbl_var))
[1] “double"

> typeof(c(int_var, log_var))
[1] “integer"

> typeof(c(log_var, chr_var))
[1] “character"

> typeof(c(chr_var, int_var))
[1] “character"

> typeof(c(chr_var, fct_var))
[1] "character"
```

COERCION EXERCISES

Now, how would you make these conversions:

- *convert* `int_var` *to a double*
- *convert* `log_var` *to a character*
- *convert* `fct_var` *to a character*
- *convert* `log_car` *to a factor*

SOLUTIONS

```
as.double(int_var)
as.character(log_var)
as.character(fct_var)
as.factor(log_var)
```

EXERCISES

How would you apply the following function to compute the sum of squares for each list item?

```
sum_of_squares <- function(x) {
      sum(x ^ 2)
}

lst <- list(c(3, 4, 8), c(4, 7, 1))
```

SOLUTIONS

```
library(tidyverse)

sum_of_squares <- function(x) {
    sum(x ^ 2)
}
lst <- list(c(3, 4, 8), c(4, 7, 1))

map_dbl(lst, sum_of_squares)
[1] 89 66
```

EXERCISES

What is the output of the following code?

```
c(25, 10, 15, 12) %>% rank()
```

SOLUTIONS

```
c(25, 10, 15, 12) %>% rank()
[1] 4 1 3 2
```

EXERCISES

`id` *is the primary key in* `df1`*. Examine the data frame* `df2`*. Which variable is the secondary key? How would you join by these keys?*

```
> df1
  id name       phone
1  4 Jane 03566658397
2  5 John 05521214979
3  6  Joe 06334794646
```

```
> df2
customer_id order_id paid
1           4      A10  Yes
2           5      A11  Yes
3           4      A12   No
4           6      A13   No
```

SOLUTIONS

```
> df1
  id name       phone
1  4 Jane 03566658397
2  5 John 05521214979
3  6  Joe 06334794646
```

```
> df2
  customer_id order_id paid
1           4      A10  Yes
2           5      A11  Yes
3           4      A12   No
4           6      A13   No
```

```
df1 %>% left_join(df2, by = c("id" = "customer_id")
```


LET'S GET STARTED!