Working with Data in the Tidyverse

Renato Erazo

18/12/2020

## ================== 01 Explore your data

Read a CSV file

In this exercise, you’ll use read\_csv() twice. The first time you will only specify the filename, but you’ll notice a problem with the imported data. The second time you’ll use a new argument called skip to fix the problem. Remember to use ?read\_csv to read more about arguments like skip and how to use them.

The data you’ll work with is from “The Great British Bake-Off.” The file “bakeoff.csv” contains data for each episode of the show, organized by series and baker.

This course touches on a lot of concepts you may have forgotten, so if you ever need a quick refresher, download the Tidyverse Cheat Sheet and keep it handy! Instructions 2/4 25 XP

2  
3  
4  
  
Use read\_csv() to read in "bakeoff.csv", and assign it to a new dataset bakeoff using the assignment operator (<-).

# Load readr  
library(readr)

## Warning: package 'readr' was built under R version 4.0.3

# Create bakeoff from "bakeoff.csv"  
bakeoff <- read\_csv("bakeoff.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## series = col\_double(),  
## episode = col\_double(),  
## baker = col\_character(),  
## signature = col\_character(),  
## technical = col\_double(),  
## showstopper = col\_character(),  
## result = col\_character(),  
## uk\_airdate = col\_date(format = ""),  
## us\_season = col\_double(),  
## us\_airdate = col\_date(format = "")  
## )

Exercise Read a CSV file

In this exercise, you’ll use read\_csv() twice. The first time you will only specify the filename, but you’ll notice a problem with the imported data. The second time you’ll use a new argument called skip to fix the problem. Remember to use ?read\_csv to read more about arguments like skip and how to use them.

The data you’ll work with is from “The Great British Bake-Off.” The file “bakeoff.csv” contains data for each episode of the show, organized by series and baker.

This course touches on a lot of concepts you may have forgotten, so if you ever need a quick refresher, download the Tidyverse Cheat Sheet and keep it handy! Instructions 4/4 25 XP

4  
  
Adapt your code to read in "bakeoff.csv" again, but this time, use the skip argument to skip the first line before reading the data. Print again to view it. Remember you can use ?read\_csv

# Load readr  
library(readr)  
  
# Create bakeoff but skip first row  
bakeoff <- read\_csv("bakeoff.csv", skip = 1)

## Warning: Missing column names filled in: 'X9' [9], 'X10' [10]

## Warning: Duplicated column names deduplicated: '1' => '1\_1' [2]

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## `1` = col\_double(),  
## `1\_1` = col\_double(),  
## Annetha = col\_character(),  
## `Light Jamaican Black Cake  
## with Strawberries and Cream` = col\_character(),  
## `2` = col\_double(),  
## `Chocolate with Cigarellos, Fresh Fruit, and Cream` = col\_character(),  
## IN = col\_character(),  
## `2010-08-17` = col\_date(format = ""),  
## X9 = col\_double(),  
## X10 = col\_date(format = "")  
## )

# Print bakeoff  
bakeoff

## # A tibble: 548 x 10  
## `1` `1\_1` Annetha `Light Jamaican~ `2` `Chocolate with~ IN   
## <dbl> <dbl> <chr> <chr> <dbl> <chr> <chr>  
## 1 1 1 David "Chocolate Oran~ 3 "Black Forest F~ IN   
## 2 1 1 Edd "Caramel Cinnam~ 1 <NA> IN   
## 3 1 1 Jasmin~ "Fresh Mango an~ NA <NA> IN   
## 4 1 1 Jonath~ "Carrot Cake wi~ 9 "Three-layer wi~ IN   
## 5 1 1 Louise "Carrot and Ora~ NA "Never Fail Cho~ IN   
## 6 1 1 Miranda "Triple Layered~ 8 "Three Tiered C~ IN   
## 7 1 1 Ruth "Lemon Drizzle ~ NA "Classic Chocol~ IN   
## 8 1 1 Lea "Cranberry and ~ 10 "Chocolate, Ras~ OUT   
## 9 1 1 Mark "Sticky Marmala~ NA "Heart-shaped C~ OUT   
## 10 1 2 David "Cheddar Cheese~ 8 "Choux Pastry S~ IN   
## # ... with 538 more rows, and 3 more variables: `2010-08-17` <date>, X9 <dbl>,  
## # X10 <date>

Exercise Assign missing values

The read\_csv() function also has an na argument, which allows you to specify value(s) that represent missing values in your data. The default values for the na argument are c("“,”NA“), so both are recoded as missing (NA) in R. When you read in data, you can add additional values like the string”UNKNOWN" to a vector of missing values using the c() function to combine multiple values into a single vector.

The is.na() function is also helpful for identifying rows with missing values for a variable. Instructions 2/4 25 XP

2  
3  
4  
  
Add a filter() line after the pipe (%>%) to show only the rows in the showstopper variable coded as "UNKNOWN".

# Load dplyr  
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(readr)  
  
# Filter rows where showstopper is UNKNOWN  
#bakeoff %>% filter(showstopper == "UNKNOWN")

Import bakers data

library(readr)

Exercise Arrange and glimpse

From here, if we don’t ask you to load a package, you can assume it’s already loaded.

You can combine glimpse() with other functions in a sequence using the pipe (%>%) operator. For example, you can use other dplyr functions like arrange first, then use glimpse by adding a line after the final pipe (%>%):

bakers\_mini %>% arrange(age) %>% glimpse() # no argument needed here

Take a glimpse of the bakeoff data we imported in the first set of exercises. On which date did the first episode of the show air in the US?

(Pressing Enter in the console will not add a new line, but will run the code on the current line. If you’d like to add a new line before running your code, you can use Shift+Enter)

#=========================== 02 Tame your data