Lecture Assignment 11

Viraj Vijaywargiya

2022-05-04

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Part 11.3.5

Question 1

locale is used to set date and time formats, time zone, numbers, and encoding. Therefore, the most important arguments to locale() are date_names, date_format, time_format, tz, decimal_mark, grouping_mark, and encoding.

Question 2

Setting decimal_mark and grouping_mark to the same character throws an error stating that they both must be different.

```
locale(decimal_mark = ",")
```

```
## <locale>
## Numbers:  123.456,78
## Formats:  %AD / %AT
## Timezone: UTC
## Encoding: UTF-8
## <date_names>
## Days:   Sunday (Sun), Monday (Mon), Tuesday (Tue), Wednesday (Wed), Thursday
##         (Thu), Friday (Fri), Saturday (Sat)
## Months: January (Jan), February (Feb), March (Mar), April (Apr), May (May),
##         June (Jun), July (Jul), August (Aug), September (Sep), October
##         (Oct), November (Nov), December (Dec)
## AM/PM:  AM/PM
```

```
locale(grouping_mark = ".")
```

```
## <locale>
## Numbers:  123.456,78
## Formats:  %AD / %AT
## Timezone: UTC
## Encoding: UTF-8
## <date_names>
## Days:   Sunday (Sun), Monday (Mon), Tuesday (Tue), Wednesday (Wed), Thursday
##         (Thu), Friday (Fri), Saturday (Sat)
## Months: January (Jan), February (Feb), March (Mar), April (Apr), May (May),
##         June (Jun), July (Jul), August (Aug), September (Sep), October
##         (Oct), November (Nov), December (Dec)
## AM/PM:  AM/PM
```

When the decimal_mark is set to ",", the default value of grouping_mark is ".". When the grouping_mark is set to ".", the default value of decimal_mark is ",".

Question 5

read_csv() reads comma delimited files, read_csv2() reads semicolon separated files (common in countries where , is used as the decimal place).

Question 7

```
d1 <- "January 1, 2010"
d2 <- "2015-Mar-07"
d3 <- "06-Jun-2017"
d4 <- c("August 19 (2015)", "July 1 (2015)")
d5 <- "12/30/14" # Dec 30, 2014
t1 <- "1705"
t2 <- "11:15:10.12 PM"

parse_date(d1, "%B %d, %Y")
```

```
## [1] "2010-01-01"
```

```
parse_date(d2, "%Y-%b-%d")
```

```
## [1] "2015-03-07"
```

```
parse_date(d3, "%d-%b-%Y")
```

```
## [1] "2017-06-06"
```

```
parse_date(d4, "%B %d (%Y)")
```

```
## [1] "2015-08-19" "2015-07-01"
```

```
parse_date(d5, "%m/%d/%y")
```

```
## [1] "2014-12-30"
```

```
parse_time(t1, "%H%M")
```

```
## 17:05:00
```

```
parse_time(t2, "%H:%M:%OS %p")
```

```
## 23:15:10.12
```