date_map homework

Owen Auston-Babcock and Bradley Collins

2025-02-25

Bradley's section

Question 1

Generate sequence of dates using `lubridate`

```r
dates <- ymd("2015-01-01") %m+% months(seq(0, (2025 - 2015) * 6, by = 2))
```

Extract year, quarter, and ISO week number. Print the data.

```r
date_info <- tibble(
  DATE = dates,
  YEAR = year(dates),
  QUARTER = quarter(dates),
  ISO_WEEK = isoweek(dates)
)

date_info
```

```
## # A tibble: 31 x 4
##    DATE        YEAR QUARTER ISO_WEEK
##    <date>     <dbl>   <int>    <dbl>
##  1 2015-01-01  2015       1        1
##  2 2015-03-01  2015       1        9
##  3 2015-05-01  2015       2       18
##  4 2015-07-01  2015       3       27
##  5 2015-09-01  2015       3       36
##  6 2015-11-01  2015       4       44
##  7 2016-01-01  2016       1       53
##  8 2016-03-01  2016       1        9
##  9 2016-05-01  2016       2       17
## 10 2016-07-01  2016       3       26
## # i 21 more rows
```

Question 2

Import sample

```r
sample_dates <- c("2018-03-15", "2020-07-20", "2023-01-10", "2025-09-05")
```

Convert sample to `lubridate` format

```r
sample_dates <- ymd(sample_dates)
```

Put sample into `tibble` including the differentials. Print the `tibble`.

```
date_diffs <- tibble(
  DATE1 = head(sample_dates, -1),
  DATE2 = tail(sample_dates, -1),
  MONTH_DIFF = interval(DATE1, DATE2) %/% months(1),
  WEEK_DIFF = interval(DATE1, DATE2) %/% weeks(1)
)

date_diffs
```

```
## # A tibble: 3 x 4
##   DATE1      DATE2      MONTH_DIFF WEEK_DIFF
##   <date>     <date>          <dbl>     <dbl>
## 1 2018-03-15 2020-07-20         28       122
## 2 2020-07-20 2023-01-10         29       129
## 3 2023-01-10 2025-09-05         31       138
```

Owen's section

Question 3

Generate list of numeric vectors.

```
num_lists <- list(c(4,16,25,36,49), c(2.3,5.7,8.1,11.4), c(10,20,30,40,50))
```

Extract mean, median and standard deviation using `map`.

```
map(num_lists, mean)
```

```
## [[1]]
## [1] 26
##
## [[2]]
## [1] 6.875
##
## [[3]]
## [1] 30
```

```
map(num_lists, median)
```

```
## [[1]]
## [1] 25
##
## [[2]]
## [1] 6.9
##
## [[3]]
## [1] 30
```

```
map(num_lists, sd)
```

```
## [[1]]
## [1] 17.42125
##
## [[2]]
## [1] 3.8422
##
## [[3]]
## [1] 15.81139
```

Extract mean, median and standard deviation using `map_dbl`.

```r
map_dbl(num_lists, mean)
```

```
## [1] 26.000  6.875 30.000
```

```r
map_dbl(num_lists, median)
```

```
## [1] 25.0  6.9 30.0
```

```r
map_dbl(num_lists, sd)
```

```
## [1] 17.42125  3.84220 15.81139
```

Question 4

Generate list of mixed date formats.

```r
date_strings <- list("2023-06-10", "2022/12/2", "15-Aug-2021", "InvalidDate")
```

Create a function that safely converts characters into `Date` format.

```r
safe_convert <- possibly(~ parse_date_time(.x, orders=c("ymd","d-b-Y")),
           otherwise=NA_Date_, quiet=TRUE)
```

Apply the converter function to the character list.

```r
date_strings <- date_strings %>%
  map(safe_convert)
```

Extract the full month name for the non-`NA` dates.

```r
date_strings %>%
  map(month, label=TRUE, abbr=FALSE)
```

```
## [[1]]
## [1] June
## 12 Levels: January < February < March < April < May < June < ... < December
##
## [[2]]
## [1] December
## 12 Levels: January < February < March < April < May < June < ... < December
##
## [[3]]
## [1] August
## 12 Levels: January < February < March < April < May < June < ... < December
##
## [[4]]
## [1] <NA>
## 12 Levels: January < February < March < April < May < June < ... < December
```