P04 – stringr und dplyr

3. Mai 2021

Contents

Hinweise zur Abgabe:

Erstelle pro Aufgabe eine R-Code-Datei und benenne diese nach dem Schema `P<Woche>-<Aufgabe>.R` also hier `P04-1.R` und `P04-2.R`. Schreibe den Code zur Lösung einer Aufgabe in die jeweilige Datei.

Es ist erlaubt (aber nicht verpflichtend) zu zweit abzugeben. Abgaben in Gruppen von drei oder mehr Personen sind nicht erlaubt. Diese Gruppierung gilt nur für die Abgabe der Programmierprobleme, nicht für die Live-Übungen.

Bei Abgaben zu zweit gibt nur eine der beiden Person ab. Dabei müssen in **jeder** abgegebenen Datei in der **ersten Zeile** als Kommentar **beide** Namen stehen also zB

```
# Ada Lovelace, Charles Babbage

1+1
# ...
```

Die Abgabe der einzelnen Dateien (kein Archiv wie `.zip`) erfolgt über Moodle im Element namens `P04`. Die Abgabe muss bis spätestens Sonntag, 9. Mai 2021, 23:59 erfolgen.

1 stringr (24 Punkte)

Eine Buchhändlerin klassifiziert ihre Bücher in 26 Kategorien `A` bis `Z`. Jedes Buch kann dabei mehreren Kategorien angehören. In dem Textdokument `books.txt` ist angegeben, wie viele Bücher einer Kategoriekombination vorhanden sind. Diese Datei wird mit `lines <- read_lines("books.txt")` als `character`-Vektor geladen.

Leider gab es bisher kein System, wie die Angaben aufgeschrieben werden. Es werden Klein- und Großbuchstaben verwendet, wobei `a` gleich `A` ist. Einzelne Kategorien einer Kombination werden direkt hintereinander geschrieben oder mit Trennzeichen. Trennzeichen zwischen einzelnen Einträgen sind auch nicht immer gleich. Manchmal stehen Kommentare in runden Klammern `(`, `)`. Diese haben dann keine Bedeutung für den aktuellen Bestand.

1. Extrahiere aus der Datei ein Tibble `data` mit zwei Spalten: `category` (Typ `character`) und `count` (Typ `integer`). `category` ist jeweils ein String der ausschließlich aus Großbuchstaben besteht in alphabetischer Reihenfolge.

```
data
## # A tibble: 200 x 2
##    category count
##    <chr>    <int>
##  1 ABGHQV      21
##  2 ABILO       22
```

```
##  3 ABJY        30
##  4 ABY         12
##  5 ACDMNOTW    14
##  6 ACDMNP      24
##  7 ACDRWX      15
##  8 ACEO        21
##  9 ACOW        18
## 10 ACWX        18
## # ... with 190 more rows
```

2. Schreibe eine Funktion `books_of_category(data, cat_let)`, welche einen `character`-Vektor ausgibt. Dabei ist jeder Eintrag von der Form `We have <x> books of category <y>.`. Hier steht `<x>` für die Anzahl an Büchern und `<y>` für die zugehörige Kategoriekombination. Es werden nur Kategoriekombinationen ausgegeben, die den Buchstaben `cat_let` enthalten.

```
books_of_category(data, "R")
## We have 15 books of category ACDRWX.
## We have 14 books of category AFRY.
## We have 23 books of category AJKMRSV.
## We have 22 books of category AKRW.
## We have 26 books of category ALMR.
## We have 14 books of category BEILRTW.
## We have 21 books of category BJKRV.
## We have 20 books of category BLR.
## We have 26 books of category CDLR.
## We have 25 books of category CERSVWY.
## We have 22 books of category CGILR.
## We have 27 books of category CHRWZ.
## We have 24 books of category DEHKRUW.
## We have 28 books of category DEMPRW.
## We have 25 books of category DFHJRST.
## We have 25 books of category DGKLRS.
## We have 19 books of category EGRS.
## We have 26 books of category EORWZ.
## We have 20 books of category FIMPRTV.
## We have 30 books of category FKMRSW.
## We have 26 books of category FRW.
## We have 19 books of category GKMRY.
## We have 18 books of category GR.
## We have 20 books of category GRV.
## We have 19 books of category HJRTU.
## We have 27 books of category HLRSYZ.
## We have 36 books of category HRVWX.
## We have 18 books of category IJLRWZ.
## We have 19 books of category ILR.
## We have 25 books of category KPR.
## We have 20 books of category LMORVYZ.
## We have 20 books of category OPR.
## We have 22 books of category R.
## We have 16 books of category RZ.
```

2 dplyr (36 Punkte)

In der Datei `P04-2-exerc.R` wird zu Beginn ein Dataset durch Simulation erzeugt.

```
sports
## # A tibble: 1,000 x 7
##    name      grade letter time100 time200 time400 time1000
##    <chr>     <int> <chr>    <dbl>   <dbl>   <dbl>    <dbl>
##  1 Jess          8 a         25.0    52.3   127.      372.
##  2 Maxie         8 b         22.2    46.1   107.      285.
##  3 Iona         11 d         19.3    43.8    97.4     260.
##  4 Leslie        6 d         30.4    67.1   135.      449.
##  5 Mordechai     5 d         31.4    72.0   177.      521.
##  6 Meir         11 a         21.9    46.0   101.      266.
##  7 Nannette      9 a         24.9    48.5   117.      348.
##  8 Jevon         9 b         22.1    41.7   104.      258.
##  9 Mae          11 a         23.6    46.4    97.1     279.
## 10 Paisley       7 c         26.5    66.1   129.      413.
## # ... with 990 more rows
```

Die Tabelle `sports` enthält Zeiten des 100m-, 200m-, 400m- und 1000m-Laufs des Sportfests einer Schule (alles fiktiv). Die Spalten sind

- `name`: Vorname des Schülers / der Schülerin
- `grade`: Jahrgangsstufe (zwischen 5 und 11)
- `letter`: in jeder Jahrgangsstufe gibt es 4 Klassen: `a` bis `d`.
- `timeXXX`: die Zeit in Sekunden, die die Person zum Laufen der Distanz `XXX` benötigt hat

In der Datei `P04-2-exerc.R` stehen im unteren Teil Kommentar-Abschnitte markiert mit `a)` bis `l)`. Diese beschreiben eine erwartete Ausgabe, die mit `dplyr`-Verben erzeugt werden soll.

Schreibe einen Ausdruck, der die geforderten Werte liefert direkt unter den entsprechenden Kommentar.

Nutze ggf den Pipe-Operator `%>%` (Shortcut: `Ctrl` + `Shift` + `M` bzw `Cmd` + `Shift` + `M`).

Sofern nicht anders angegeben, müssen Spalten nicht extra ausgewählt werden.