

# Übung: Campusbier-Bestellungen transformieren

## Vorbereitung

Ladet den Datensatz der Campusbier-Bestellungen mit den folgenden Zeilen:

```
library(tidyverse)
orders <- read_csv("data/orders.csv")
```

Achtung: Der Code oben geht davon aus, dass die Datei `orders.csv` in eurem Arbeitsverzeichnis im Unterordner `data/` liegt. Ist das nicht Fall, muss die Pfadangabe angepasst werden.

Speichert die Zeilen in einem neuen R-Skript und tragt dort auch eure Lösungen für die folgenden Aufgaben ein. Vergesst nicht, die Datei zwischendurch zu speichern.

## Aufgabe 1: Spalten auswählen mit `select`

Findet Lösungen für die folgenden Aufgaben, in denen ihr jeweils eine Teilmenge der Spalten auswählen sollt:

1. Wählt alle Spalten aus, die mit `__at` enden!
2. Wählt alle Spalten aus, die Informationen zur Lieferadresse enthalten!
3. Wählt alle Spalten aus, die einen Datums- oder Zeitwert enthalten!
4. Wählt die erste und die letzten drei Spalten aus!
5. Wählt alle Spalten aus, die keine Kundeninformation oder Informationen zur Lieferadresse enthalten!

## Aufgabe 2: Zeilen einschränken mit `filter`

Findet eine Lösung, um die Zeilen des Datensatzes wie beschrieben einzuschränken:

1. Wählt nur Bestellungen aus dem Jahr 2020 aus!
2. Entfernt alle Bestellungen, die mit PayPal bezahlt wurden!
3. Entfernt die Top 1% der Bestellungen mit dem höchsten Umsatz!
4. Lasst nur Bestellungen von außerhalb des 49 Postleitzahlengebietes im Ergebnis!
5. Wählt alle Bestellungen aus, in denen ein Discount-Code verwendet wurde!

## Aufgabe 3: Neue Spalten erzeugen mit `mutate` oder `transmute`

1. Erstellt eine neue Spalte mit dem Wochentag, an dem die Bestellung getätigt wurde! Die Spalte soll einen sprechenden Namen des Wochentags (Montag, Dienstag etc.) enthalten!
2. Ermittelt alle Kunden, die Marketing-Mails akzeptieren und die gleichzeitig PayPal verwenden! Erstellt eine neue Spalte `marketing_paypal`, die `TRUE` beinhaltet wenn beides zutrifft!
3. Berechnet eine neue Spalte `turnover_rank`, die jeder Bestellung anhand des Umsatzes eine Rangnummer zuweist. Die Bestellung mit dem höchsten Umsatz soll den Rang 1 bekommen.
4. Fügt eine Spalte hinzu, die angibt, ob eine Bestellung vollständig bezahlt und ausgeführt wurde! Schaut euch dazu die Spalten `financial_status` und `fulfillment_status` an!
5. Berechnet die Zeitspanne zwischen Bestellungseingang und dem endgültigen Abschluss der Bestellung und speichert den Wert in einer neuen Spalte!

## Aufgabe 4: Daten zusammenfassen mit `group_by` und `summarize`

1. Erstellt eine Übersicht für die Häufigkeit der verwendeten Zahlungsmethoden! Sortiert das Ergebnis in absteigender Reihenfolge!
2. Welche Top 10 Kunden haben den meisten Umsatz generiert?
3. Zeigt die Summe des Umsatzes für alle im Datensatz enthaltenen Jahre! Gebt neben dem Umsatz auch die Anzahl der Bestellungen in jedem Jahr aus!
4. Aus welchen Postleitzahlengbieten kommen wie viele Bestellungen?
5. In welchem Postleitzahlengebiet ist der durchschnittliche Einkaufswert pro Bestellung am höchsten?