

Lösungen zu den Aufgaben

1. Aufgabe

Korrigieren Sie den Fehler in der Syntax:

```
mean(x = c(1, 5, 10, 52)
```

Ändern Sie *nur* diejenigen Teile der Syntax, die zwingend geändert werden müssen, damit der Fehler korrigiert wird.

Lösung

```
mean(x = c(1, 5, 10, 52))
```

```
## [1] 17
```

Die Antwort lautet: `mean(x = c(1, 5, 10, 52))`.

2. Aufgabe

R gibt folgende Fehlermeldung aus:

```
(Fehler in library(XXX): es gibt kein Paket namens 'XXX'),
```

wobei für `xxx` ein Paketname wie `tidyverse` angeführt wird.

Wählen Sie die plausibelste Ursache aus!

- a. Das Paket `xxx` ist nicht installiert auf dem aktuellen Rechner.
- b. Das Paket `xxx` ist nicht verfügbar genau für dieses Betriebssystem.
- c. Es existiert kein Paket mit Namen `xxx`.
- d. Das Paket `xxx` ist nicht geladen.
- e. Das Paket `xxx` ist defekt.

Lösung

Das Paket `xxx` ist nicht installiert auf dem aktuellen Rechner.

- a. Richtig.
- b. Falsch.
- c. Falsch.
- d. Falsch.
- e. Falsch.

3. Aufgabe

Gegeben sei diese Syntax:

```
x <- 42
Y <- 1
```

Lässt man folgende Syntax laufen, so kommt eine Fehlermeldung:

```
X + Y

## Error in eval(expr, envir, enclos): object 'X' not found
```

Geben Sie die korrekte Syntax ein (zur Berechnung der Summe), die *nicht* zu einer Fehlermeldung führt!

Bitte verwenden Sie *keine* Leerzeichen bei Ihrer Eingabe.

Lösung

```
x+Y

## [1] 43
```

Die Antwort lautet: `x+Y`.

4. Aufgabe

Gegeben sei diese Syntax, die einen Fehlermeldung ausgibt:

```
mean(c(1,2,3,4). na.rm = TRUE)

## Error: <text>:1:16: unexpected symbol
## 1: mean(c(1,2,3,4).
##                      ^
```

Geben Sie die korrekte Syntax ein, die *nicht* zu einer Fehlermeldung führt!

Bitte verwenden Sie *keine* Leerzeichen bei Ihrer Eingabe.

Lösung

```
mean(c(1,2,3,4), na.rm = TRUE)

## [1] 2.5

sol <- "mean(c(1,2,3,4),na.rm=TRUE)"
```

Die Antwort lautet: `mean(c(1,2,3,4),na.rm=TRUE)`.

5. Aufgabe

Gegeben sei diese Syntax, die einen Fehlermeldung ausgibt:

```
notendaten_url <- "https://vincentarelbundock.github.io/Rdatasets/csv/MASS/nlschools.csv"

noten <- read_csv(notendaten_url)

filter(noten, IQ = 10)
```

```
## Error in `filter()`:
## ! We detected a named
##   input.
## i This usually means that
##   you've used `=` instead of
##   `==`.
## i Did you mean `IQ == 10`?
```

Der Datensatz stammt aus [dieser Quelle](#).

Geben Sie die korrekte Syntax ein in der Zeile mit `filter()`, die *nicht* zu einer Fehlermeldung führt!

Es sollen nur die Schüler im Datensatz verbleiben, die im IQ-Test einen Wert von 10 aufweisen.

Bitte verwenden Sie *keine* Leerzeichen bei Ihrer Eingabe.

Lösung

```
filter(noten, IQ == 10)
```

...1 lang IQ class GS SES COMB

15	33	10	180	29	10	0
24	21	10	180	29	10	0
25	42	10	180	29	10	0
46	37	10	1280	31	20	1
59	41	10	1580	35	30	1
89	18	10	1880	28	35	0

Die Antwort lautet: `filter(noten, IQ==10)`.

6. Aufgabe

Betrachten Sie folgende R-Syntax, für die R eine Fehlermeldung ausgibt:

```
sum(square(mean(x = c(1,2,3)) - x))

## Error: <text>:1:36: unexpected ')'
## 1: sum(square(mean(x = c(1,2,3)) - x))
##                                     ^
```

Geben Sie die korrekte Syntax an! Ändern Sie *nur* die *notwendigen* Zeichen an der Syntax oben. Gehen Sie davon aus, dass die aufgerufenen Funktionen existieren.

Lösung

Hinten ist eine (schließende) Klammer zu viel, die muss weg:

```
square <- function(x) ``^(x, 2) # Funktion `square` definieren
sum(square(mean(x = c(1,2,3)) - x)) # so geht's

## Error in square(mean(x = c(1, 2, 3)) - x): object 'x' not found
```

Die Antwort lautet: `sum(square(mean(x = c(1,2,3)) - x))`.

7. Aufgabe

Sie möchten in R eine Textvariable (String-, Charaktervariable) definieren und zwar soll die Variable `gruss` mit dem Wert "Hallo" "befüllt" werden.

Sie geben in der Konsole folgende Syntax ein:

```
gruess <- "hall
```

Das quittiert R folgendermaßen:

```
> gruss <- "hallo  
+
```

Wie ist diese Reaktion von R zu interpretieren bzw. was ist zu tun?

- a. R ist abgestürzt; am besten neu starten.
- b. R verträgt im Standard nur Grüße in englischer Sprache. Sprachpakete updaten.
- c. R wartet auf das Ende der Text-Auszeichnung, also auf das schließende Anführungszeichen. Das muss noch eingegeben werden. Alternativ kann man "Escape" drücken.
- d. Es gibt kein Problem; man kann einfach den nächsten Befehl eingeben.
- e. R hat gewartet auf das Ende der Text-Auszeichnung, also auf das schließende Anführungszeichen. Jetzt ist R abgestürzt und muss neu gestartet werden.

Lösung

- a. Falsch
- b. Falsch
- c. Richtig
- d. Falsch
- e. Falsch

8. Aufgabe

Eine Dozentin überprüft Klausurnoten. Diese seien im Vektor `noten` gespeichert:

Sie möchte nun alle Noten auslesen, die schlechter als 3.3 sind. Geben Sie die passende R-Syntax dazu sein. Die Syntax soll nur aus einem Befehl bestehen und muss ohne Zusatzpakete lauffähig sein.

Lösung

Die Antwort lautet: `noten[noten>3.3]`.

9. Aufgabe

Welchen Wert bzw. welches Ergebnis liefert folgende R-Syntax für `ergebnis` zurück?

`x` hat zu Beginn den Wert 24.

Hinweis: `sqrt(x)` liefert die Quadratwurzel von `x` zurück.

Lösung

Es wird 5 zurückgeliefert.

10. Aufgabe

Welchen Wert bzw. welches Ergebnis liefert folgende R-Syntax für `ergebnis` zurück?

`x` hat zu Beginn den Wert 24.

Hinweise:

- `sqrt(x)` liefert die (positive) Quadratwurzel von `x` zurück.
- `x^2` liefert die zweite Potenz von `x` zurück.

Lösung

Es wird 25 zurückgeliefert.

11. Aufgabe

Geben Sie die R-Syntax ein, um zu prüfen, dass die Variable `loesung` den Wert 42 hat.

Hinweis: Geben Sie Ihre Lösung *ohne* Leerzeichen an, da sonst eine richtige Lösung nicht erkannt werden kann.

Lösung

```
loesung==42
```

12. Aufgabe

Wählen Sie die syntaktisch korrekten Varianten aus, um einer Variablen den Wert (Zahl) 42 zuzuweisen.

- a. `loesung <-42`
- b. `loesung < - 42`
- c. `loesung->42`
- d. `loesung==42`
- e. `loesung<-"42"`

Lösung

Die Antwort lautet: `loesung <-42`.

- a. Richtig
- b. Falsch
- c. Falsch
- d. Falsch
- e. Falsch

13. Aufgabe

Weisen Sie dem Objekt `loesung` den Wert `42` zu. Geben Sie den korrekten R-Code dafür ein.

Hinweis: Verzichten Sie *jegliche Leerzeichen* in Ihrer Eingabe, da sonst die Eingabe nicht als korrekt erkannt werden kann.

Lösung

```
loesung<-42
```

14. Aufgabe

Recherchieren Sie den Datensatz "Palmer Penguins" als CSV-Datei im Internet.

1. Importieren Sie die Datendatei in R von der Online-Quelle.
2. Laden Sie die Datendatei herunter, speichern Sie Sie in den Ordner Ihres aktuellen RStudio-Projekts. Dann importieren Sie die Datendatei in R von diesem Ort.

Lösung

Ad 1)

```
penguins_url <- "https://vincentarelbundock.github.io/Rdatasets/csv/palmerpenguins/penguins.csv"
d <- read_csv(penguins_url)
## Error in gt(.): could not find function "gt"
```

Ad 2)

Wenn Sie die Datei heruntergeladen haben und in Ihrem (aktuellen) RStudio-Projektordner abgespeichert haben, können Sie sie ohne Angabe eines Pfades in R importieren:

```
d <- read_csv("penguins.csv") # die Datei muss im aktuellen Verzeichnis liegen
```