

Errata-Liste (nur für den Erstdruck)

S. 49:

„So ist beispielsweise 1 eine Instanz [...]“

Es muss heißen:

„So ist beispielsweise 2 eine Instanz [...]“

S. 85:

```
fobj.write("{} {}\n".format(engl, woerter[engl]))
```

Die schließende Klammer fehlt. Korrekt ist:

```
fobj.write("{} {}\n".format(engl, woerter[engl]))
```

S. 109:

„Wenn `help` ohne Parameter aufgerufen wird, wird ein Einleitungstext ausgegeben [...]“

Es muss heißen:

„Mit `help()` wird ein Einleitungstext ausgegeben [...]“

S. 109:

„Über die Tastenkombination **Shift+Q** gelangen Sie [...]“

Es muss heißen:

„Über die Taste **Q** gelangen Sie [...]“

S. 143:

Tabelle 13.2, zweite Spalte, vorletzte Zeile:

„Gibt den Index k des ersten Vorkommens von x in der Liste s im Bereich $i \leq k < j$ zurück.“

Es muss heißen:

„Gibt den Index k des ersten Vorkommens von x in der Sequenz s im Bereich $i \leq k < j$ zurück.“

S. 145:

```
>>> s = "Musik"
>>> s += "lautsprecher"
>>> s
'Musiklautsprecher'
```

Es muss heißen:

```
>>> s = "Musik"
>>> t = "lautsprecher"
>>> s += t
>>> s
'Musiklautsprecher'
```

S. 145:

„Es existieren also nach der Operation `s += t` drei verschiedene Instanzen im Speicher, nämlich mit den Werten `s`, `t` und `s + t`.“

Es muss heißen:

„Es existieren also nach der Operation `s += t` drei verschiedene Instanzen im Speicher, nämlich `"Musik"`, `"lautsprecher"` und `"Musiklautsprecher"`, wobei es keine Referenz mehr auf `"Musik"` gibt.“

S. 168:

Die Zeile

```
>>> zahlen = [11, 17, 17, 19, 10]
```

einfügen vor der Zeile

```
>>> *irgendwas, neunzehn, zehn = zahlen
```

S. 173:

In der vierten Zeile des ersten Listings fehlt ein Backslash vor `t`. Die Zeile muss lauten:

```
>>> 'Ein \\tString mit \\ \\ vielen \\nEscape-Sequenzen\\t'
```

S. 193:

„Es können also $2^7 = 128$ verschiedene Zeichen abgebildet werden.“

Es muss heißen:

„Es können also $2^7 = 128$ verschiedene Zeichen abgebildet werden.“

S. 206:

Tabelle 14.2, zweite Spalte, sechste Zeile (`k` statt `key`):

„Gibt den zum Schlüssel `key` gehörigen Wert zurück und löscht das Schlüssel-Wert-Paar aus dem Dictionary `d`.“

Es muss heißen:

„Gibt den zum Schlüssel `k` gehörigen Wert zurück und löscht das Schlüssel-Wert-Paar aus dem Dictionary `d`.“

S. 225:

Unter Tabelle 16.1 muss dieser Absatz ergänzt werden:

Um das Modul `collections` verwenden zu können, muss es zunächst eingebunden werden:

```
>>> import collections
```

S. 235:

Vor Abschnitt 17.1 diesen Absatz ergänzen:

Um die Module `time` und `datetime` verwenden zu können, müssen sie zunächst eingebunden werden:

```
>>> import time
```

```
>>> import datetime
```

S. 255:

Vor der Zeile

```
>>> class Wochentag(enum.Enum):
```

diese Zeile einfügen:

```
>>> import enum
```

S. 261:

„Dazu bekommt sie das entsprechende Objekt als Parameter übergeben [...]“

Es muss heißen:

„Dazu bekommt sie das entsprechende Objekt als Argument übergeben [...]“

S. 261:

```
>>> len("Dieser String ist ein Parameter")
31
```

Es muss heißen:

```
>>> len("Dieser String ist ein Argument")
30
```

S. 261:

„Bei einem Funktionsaufruf wird dann der Funktionskörper unter Berücksichtigung der übergebenen Parameter ausgeführt.“

Es muss heißen:

„Bei einem Funktionsaufruf wird dann der Funktionskörper unter Berücksichtigung der übergebenen Argumente ausgeführt.“

S. 262:

Vor der Zeile

```
>>> for i in ergebnis:
```

diese Zeile einfügen:

```
>>> ergebnis = range(0, 10, 2)
```

S. 375:

„Ein Beispiel für die Verwendung binärer Operatoren finden Sie in Abschnitt 21.7.2.“

Es muss heißen:

„Ein Beispiel für die Verwendung binärer Operatoren finden Sie zu Beginn dieses Abschnitts.“

S. 403/404:

„Ein Generator ist eine Funktion, die bei jedem Aufruf das nächste Element einer virtuellen Sequenz zurückgibt.“

Es muss heißen:

„Ein Generator ist eine Funktion, die beim Aufruf ein iterierbares Objekt erzeugt, welches der Reihe nach die Elemente einer virtuellen Sequenz liefert.“

S. 547:

Tabelle 32.2, zweite Spalte, erste Zeile (e entfällt).

„Meldet eine neue Aufgabe beim Executor e an.“

Es muss heißen:

„Meldet eine neue Aufgabe beim Executor an.“

S. 547:

Tabelle 32.2, erste Spalte, zweite Zeile: `func` durch „`fn`“ ersetzen:

```
map(fn, [*iterables], {timeout})
```

S. 552:

In der Formel ganz oben fehlt vor $\frac{\pi}{2}$ ein Gleichheitszeichen:

$$\frac{2}{1} \cdot \frac{2}{3} \cdot \frac{4}{3} \cdot \frac{4}{5} \cdot \frac{6}{5} \cdot \frac{6}{7} \cdot \frac{8}{7} \cdot \frac{8}{9} \cdots = \frac{\pi}{2}$$

S. 553:

Im ersten Listing muss die dritte Zeile lauten:

```
print(f.result())
```

S. 642:

Tabelle 34.5, zweite Spalte, erste und zweite Zeile:

„[...] und gibt sie als String zurück“

Es muss heißen:

„[...] und gibt sie als `bytes`-String zurück“.

S. 642:

Tabelle 34.5, zweite Spalte, dritte Zeile:

„[...] und gibt sie in Form einer Liste von Strings zurück“.

Es muss heißen:

„[...] und gibt sie in Form einer Liste von `bytes`-Strings zurück“.

S. 945:

„In diesem Beispiel wird ein Array mit zwei Dimensionen, also eine Matrix, erzeugt, die zeilenweise die Elemente der übergebenen Listen beinhaltet.“

Es muss heißen:

„Im Beispiel hat das Array `x` demnach eine Dimension mit vier Komponenten.“