Modul 11 · Wahr oder falsch?

Übersicht: alle Merk- und Infokästen mit ihren Beispielen

Aufgabe 1 – Kristalle im Gang

Erst nachsehen, dann nehmen

Der Avatar soll nur dann einen Kristall nehmen, wenn auch einer da liegt. Umgangssprachlich heißt das:

WENN hier ein Kristall liegt: **nimm ihn**

Der Avatar prüft also **einmal** auf jedem Feld, ob etwas stimmt, und entscheidet dann. Das ist keine Wiederholung, sondern eine **Verzweigung**: Das Programm kann zwei verschiedene Wege gehen. Eine Verzweigung beginnt mit **wenn** – in Python mit dem Schlüsselwort `if` (englisch für *wenn*):

```
if crystal_here():
    take()
```

Hinter dem **wenn** steht die **Bedingung** – hier die Frage `crystal_here()`, „Liegt hier ein Kristall?“. Um solche Bedingungen geht es in diesem Heft.

Aufgabe 2 – Aussagen

Aussagen

Die Frage „Liegt hier ein Kristall?“ hat immer genau eine von zwei Antworten: ja oder nein. Deshalb steckt in jeder Bedingung eine **Aussage**:

- Eine **Aussage** ist ein Satz, dem man eindeutig einen **Wahrheitswert** zuordnen kann. Sie ist entweder wahr (`True`) oder falsch (`False`), aber nie beides gleichzeitig.
- Den Datentyp für die Wahrheitswerte `True` und `False` nennt man **Boolean**, in Python `bool`.
- **Fragen** und **Befehle** sind keine Aussagen. Der Wahrheitswert muss jederzeit eindeutig feststehen. Er darf also nicht von einer **subjektiven** Meinung abhängen.

Beispiele:

- „1 + 2 = 3.“ – Aussage, **wahr**
- „2 + 3 = 6.“ – Aussage, **falsch**
- „Saarbrücken liegt am Rhein.“ – Aussage, **falsch**
- „Es ist gerade 10:25 Uhr.“ – Aussage: um 10:25 Uhr **wahr**, sonst **falsch**
- „Wie viel Uhr ist es?“ – keine Aussage (Frage)
- „Setz dich hin!“ – keine Aussage (Befehl)
- „Das Wetter ist schön.“ – keine Aussage (subjektiv)

Aufgabe 3 – Aussagen verknüpfen

Beispiele

- WENN heiß UND Wochenende: Baden im Pool! – in Python: `heiss and wochenende`
- WENN Feiertag ODER Wochenende: ausschlafen! – in Python: `feiertag or wochenende`
- WENN NICHT warm: Mütze an! – in Python: `not warm`

und, oder, nicht

Noch ein Beispiel: **WENN** (Tag in den Sommerferien) **UND** (wärmer als 25 Grad) **DANN** schwenken!

- `A and B` ist genau dann wahr, wenn A **und** B wahr sind, sonst falsch.
- `A or B` ist genau dann wahr, wenn A **oder** B **oder beide** wahr sind, sonst falsch.
- `not A` **kehrt** eine Aussage **um** (Negation): Es ist wahr, wenn A falsch ist, und falsch, wenn A wahr ist.

ALLGEMEIN

```
aussage1 and aussage2
aussage1 or aussage2
not aussage
```

BEISPIEL

```
sommerferien and warm
feiertag or wochenende
not warm
```

Beispiel: „7 ist eine Primzahl **und** 8 ist kleiner als 13“ ist **wahr**, weil beide Teile wahr sind.

In der Höhle: `not wall_ahead() and not trap_ahead()` ist genau dann wahr, wenn vor dem Avatar weder Fels noch eine Falle ist.

Aufgabe 4 – Bedingungen in Python

Vergleichsoperatoren

Mit **Vergleichsoperatoren** vergleichst du zwei Werte. Das Ergebnis ist eine Aussage, also `True` oder `False`:

- `==` gleich, `!=` ungleich
- `<` kleiner, `>` größer
- `<=` kleiner oder gleich, `>=` größer oder gleich

ALLGEMEIN

```
wert1 vergleichsoperator wert2
```

BEISPIEL

```
alter >= 18
name == "Max"
alter > 18 or erlaubnis == "ja"
```

Achtung: Ein `=` ist eine **Zuweisung** (`x = 5` legt 5 in x). Zwei `==` **vergleichen** (`x == 5` fragt, ob x gleich 5 ist).

Auch **Texte** kann man vergleichen: `name == "Max"` prüft, ob der Text genau gleich ist – Groß- und Kleinschreibung zählt.

Verknüpfen kannst du die Vergleiche mit `and`, `or` und `not`: `alter > 18 or name == "Max"`.

Zusammenfassung

Das Wichtigste

Hinter `if` steht eine Bedingung. Darin steckt eine _____: Sie ist entweder wahr (_____) oder falsch (_____). Ihr Datentyp heißt _____. Ob zwei Werte gleich sind, prüft man mit _____ – ein einzelnes `=` ist eine Zuweisung. Aussagen verknüpft man mit _____ (beide wahr), _____ (mindestens eine wahr) und _____ (umkehren).