PROFESSOR ERKLÄRBÄR

Serie Informatik · Folge 8 · Grundlagen

INFORMATIK

LEHRERINFO ZUM VIDEO

Entscheidungen und Schleifen

FACH	STUFE	DAUER	VIDEO
Informatik	Klasse 7–9	11 min	youtube.com/@Erklaerbaer

01 Lernziele

Nach dem Video können die Schülerinnen und Schüler …

- Bedingungen mit den Vergleichsoperatoren `==` `!=` `<` `>` `<=` `>=` formulieren,
- Entscheidungen mit `if`, `elif` und `else` programmieren und richtig einrücken,
- Bedingungen mit `and`, `or` und `not` verknüpfen (Rückbezug Logikgatter, Folge 4),
- Wiederholungen mit `for` und `range()` programmieren,
- `while`-Schleifen schreiben und Endlosschleifen vermeiden,
- Zähl- und Summenvariablen einsetzen.

02 Voraussetzungen

Variablen, Datentypen, `input()` und `print()` (Folge 7); Sequenz, Verzweigung, Wiederholung und Halbieren beim Zahlenraten (Folge 6); UND, ODER, NICHT (Folge 4).

03 Ablauf des Videos

ZEIT	KAPITEL	INHALT UND BEISPIEL
0:00	Einstieg	ohne Entscheidung immer dasselbe · Spiel ohne Schleife · Ident · Fahrplan
0:34	Vergleiche	age >= 12 → True/False · sechs Operatoren · = gegen == · Passwort · lives == 0
1:23	if und else	Altersfreigabe ab 12 · Einrückung = Block · falsch eingerückt
2:07	elif: mehrere Fälle	Akku-Anzeige 80 / 35 / 10 · Reihenfolge zählt · Merksatz
2:55	and, or, not	wie die Gatter · if lives > 0 and not paused · Thermostat 18–24 °C
3:30	for-Schleifen	Siebnerreihe mit range(1, 11) · range(4) PAUSE-FRAGE range(3, 8)
4:21	while-Schleifen	Countdown · Passwortsperre · Endlosschleife · Wecker PAUSE-FRAGE x − 3 bis −2 · Merksatz

6:00	Zählen und Summieren	total = total + n · Punkte im Spiel · Gauss: 5050
6:47	Alles zusammen: Zahlenraten	randint, while, if/elif · Testlauf 50 → 25 → 37
7:32	Projekt-Baustein	Taschenrechner 0.2: Rechenart wählen, Division durch null abfangen, Nochmal?
7:51	Typische Fehler	vier Fehlvorstellungen (siehe 04)
8:29	Zusammenfassung	Kernaussagen · Programm reagiert
9:04	Quiz	4 Fragen · Endkarte Folge 9

04 Typische Fehlvorstellungen

- `if x = 5:` – verglichen wird mit `==`; ein Gleichheitszeichen weist zu.
- `range(1, 10)` zählt bis 10 – nur bis 9; die Endzahl ist nicht dabei.
- Einrückung ist nur Optik – in Python bestimmt sie, welche Zeilen zum Block gehören.
- Die Zählvariable der while-Schleife muss sich nicht ändern – sonst entsteht eine Endlosschleife. Die Bedingung wird vor jedem Durchlauf geprüft, nicht ständig überwacht.

05 Einsatz im Unterricht

Einstieg: Die Klasse spielt Zahlenraten zu zweit und notiert das Vorgehen als Pseudocode – danach wird daraus Python (Kapitel 8).

Schreibtischtest: Schleifen von Hand mit einer Variablentabelle durchspielen (Aufgaben 3 und 4), dann in Thonny prüfen.

Fehlerjagd: Endlosschleife bewusst bauen und mit dem Stoppknopf beenden – danach die Ursache finden.

Mini-Projekt: Aufgabe 6 (Notenrechner) eignet sich als Hausaufgabe.

Hinweis: Das Passwort im Beispiel steht nur zur Veranschaulichung im Code; echte Passwörter werden nie im Klartext gespeichert (Folge 13). Die Gauss-Anekdote ist überliefert; ihre Details sind historisch nicht gesichert. `randint` stammt aus dem Modul `random` der Standardbibliothek.

06 Quellen

- Python Software Foundation: The Python Tutorial, Kapitel 4 «More Control Flow Tools» (docs.python.org/3/tutorial)
- Python-Dokumentation: Built-in Types – Comparisons und Boolean Operations · random.randint
- Thonny – Python IDE for beginners, Universität Tartu (thonny.org)
- Lehrplan 21, Medien und Informatik: Bereich «Informatik – Programmieren» (MI.2.2)

NAME _____ KLASSE _____ DATUM _____

Entscheidungen und Schleifen

1 True oder False? ●○○

Es gilt `a = 7`, `b = 3`. **Bestimme**: a) `a > b` b) `a == b + 4` c) `a != 7` d) `a > 5 and b > 5` e) `a > 5 or b > 5` f) `not a < b`

2 Ampel für Noten ●●○

Schreibe ein Programm mit `if/elif/else`: Note ab 5 → «gut», ab 4 → «genügend», sonst → «ungenügend». **Teste** mit 5.5, 4 und 3.5. Was passiert, wenn du die Bedingungen vertauschst?

3 for von Hand ●●○

Was geben die Schleifen aus? a) `for i in range(5): print(i)` b) `for i in range(2, 11, 2): print(i)` c) Wie oft läuft `for i in range(10, 15):`?

4 while von Hand ●●○

`n = 1 · while n < 50: n = n * 3 · print(n)` – **notiere** n nach jedem Durchlauf. Was wird ausgegeben? Wie oft läuft die Schleife?

5 Endlosschleife finden ●●○

`i = 10 · while i > 0: · print(i) · i = i - 1` (ohne Einrückung). **Erkläre**, warum das Programm nie endet, und **korrigiere** es.

6 Notenschnitt

Schreibe ein Programm, das so lange Noten einliest, bis 0 eingegeben wird, und danach den Durchschnitt ausgibt. Verwende eine Zähl- und eine Summenvariable.

Lösungen

1 True oder False?

a) True · b) True · c) False · d) False · e) True · f) True (7 < 3 ist False, not False ist True)

2 Ampel für Noten

`if note >= 5: print("gut") · elif note >= 4: print("genügend") · else: print("ungenügend")` → 5.5 gut, 4 genügend, 3.5 ungenügend. Steht `note >= 4` zuerst, wird auch 5.5 nur «genügend».

3 for von Hand

a) 0, 1, 2, 3, 4 · b) 2, 4, 6, 8, 10 (Schrittweite 2) · c) 5-mal (10 bis 14)

4 while von Hand

n = 3 → 9 → 27 → 81; dann ist 81 < 50 falsch. Ausgabe 81, die Schleife läuft viermal.

5 Endlosschleife finden

`i = i - 1` steht nicht im Block, i bleibt immer 10 – die Bedingung wird nie falsch. Korrektur: die Zeile ebenfalls einrücken (vier Leerzeichen).

6 Notenschnitt

```
summe = 0 · anzahl = 0 · note = float(input("Note: ")) · while note != 0: · summe = summe + note · anzahl = anzahl + 1 · note = float(input("Note: ")) · print(summe / anzahl)
```

(Vorsicht: Wird sofort 0 eingegeben, ist anzahl = 0 → vorher prüfen.)

Q Quiz im Video

1 B (==) · 2 C (0, 1, 2, 3) · 3 B (wenn die Bedingung falsch ist) · 4 B (False)