Rückblick auf Heft 0 bis 2

Dieses Catch-up ist freiwillig. Es fasst zusammen, was Sie bisher gelernt haben, und lässt Sie das Wichtigste noch einmal üben – gut nach einer Pause, bevor es mit Heft 3 weitergeht.

Was bisher geschah

Heft 0 und 1 in Kürze. Ohne Objekte hing die Figur an losen Variablen `x` und `y`. Seit Heft 1 sind Held und Knappe **Objekte**: `held.geh()` schickt den Befehl nur an den Helden, `knappe.geh()` nur an den Knappen. Tore öffnet man mit `bediene_tor()`, Türen mit dem Spruch vom Zettel (`lese_spruch()`, `sage_spruch()`).

Heft 2 – Das Prinzip der Objektorientierung. Programme lassen sich auf zwei Arten aufbauen:

- **Funktionale Abstraktion:** Das Programm ist ein großer Algorithmus, der in Prozeduren zerlegt wird. Daten und Algorithmen sind getrennt; die Daten müssen immer wieder durchgereicht werden – so wie `x` und `y` in Heft 0.
- **Datenabstraktion (objektorientierte Programmierung):** Die Objekte stehen im Mittelpunkt. Jedes Objekt hat ein „Gedächtnis“: Es kennt den Zustand seiner Daten und hat eigene Methoden, um ihn zu ändern. Objekte schicken sich **Nachrichten** – bedient der Held ein Tor, ändert das Tor daraufhin seinen Zustand.

Merke:

Bei der funktionalen Abstraktion werden Daten und _____ getrennt behandelt. Die Daten müssen immer wieder _____ werden.

Bei der objektorientierten Programmierung haben Objekte ein _____: Sie kennen immer den _____ ihrer Daten. Untereinander schicken sie sich _____.

Was unterscheidet die objektorientierte Programmierung von der funktionalen Abstraktion?

(noch nicht beantwortet)

Üben

a) Lösen Sie **Level 203A**. Der Held muss durch ein Tor, der Knappe durch eine Tür.

(noch nicht bearbeitet)

b) Warum muss in Level 203A der Knappe den Spruch selbst lesen? Der Held könnte doch auch zum Zettel gehen.

(noch nicht beantwortet)