

LU01.L08 - Paradigmen-Detektiv

Lösungstabelle

Nr.	Stil	Merkmal im Code	Verändert bestehende Daten?
1	imperativ	Schleife über einen Bereich, Akkumulator summe wird mehrfach überschrieben (+=)	ja – die Variable summe
2	deklarativ	sum(...) beschreibt das Ergebnis, kein Zwischenzustand sichtbar	nein – neuer Wert
3	imperativ	sort() als Methode auf der Liste, gibt None zurück	ja – die Liste namen selbst
4	deklarativ	sorted() liefert eine neue Liste zurück, Original bleibt unangetastet	nein – neuer Wert
5	deklarativ	Es wird beschrieben, was dargestellt wird; Schriftgrösse, Position und Aufzählungszeichen entscheidet der Browser	nein
6	imperativ	leeres Gefäss [], for-Schleife, append – Ergebnis wird Schritt für Schritt aufgebaut	ja – die Liste gefiltert wächst
7	deklarativ	List Comprehension, liest sich als Satz über das Ergebnis	nein – neuer Wert
8	deklarativ	SQL beschreibt was geändert werden soll, nicht wie die Datenbank die Zeilen findet	ja – der Zustand der Datenbank
9	imperativ	global plus += – der Zustand ausserhalb der Funktion wird verändert	ja – die globale Variable zaehler
10	deklarativ	map beschreibt die Transformation, die Iteration steckt im Sprachkonstrukt	nein – neue Liste

Antworten auf die Zusatzfragen

1. Deklarativ und trotzdem zustandsverändernd

Ausschnitt 8 (UPDATE). Die Behauptung, deklarativ sei automatisch seiteneffektfrei, ist **falsch**. Seiteneffektfreiheit ist eine Eigenschaft der *puren funktionalen* Programmierung, nicht der deklarativen Programmierung insgesamt.

- **Deklarativ** heisst nur: das *Was* wird beschrieben, nicht das *Wie*.
- **Pur funktional** verlangt zusätzlich, dass nichts ausserhalb der Funktion verändert wird.

Ein UPDATE beschreibt zwar deklarativ, welche Zeilen betroffen sind – die Wirkung auf die Datenbank bleibt aber ein handfester Seiteneffekt. Mehr dazu in LU02 (Pure Functions).

2. sort() vs. sorted()

Ausschnitt 3 (namen.sort()) ist gefährlicher. Die Methode sortiert die Liste **an Ort und Stelle**. Jede andere Stelle im Programm, die dieselbe Liste referenziert, sieht ab sofort die veränderte Reihenfolge – auch dann, wenn sie auf die ursprüngliche Eingabereihenfolge angewiesen war. Solche Fehler treten oft erst weit entfernt von der Ursache auf und sind entsprechend mühsam zu finden.

`sorted()` erzeugt dagegen eine neue Liste und lässt das Original in Ruhe. Wer die alte Reihenfolge noch braucht, hat sie noch.

3. Deklarativ ohne Funktionen

HTML ist deklarativ, kennt aber überhaupt keine Funktionen. Daraus folgt: **deklarativ ist der Oberbegriff, funktional eine Teilmenge davon**. Funktionale Programmierung ist ein deklarativer Stil, der zusätzlich Funktionen als Bausteine und Unveränderlichkeit verlangt. Andere deklarative Familien sind zum Beispiel die logische Programmierung (Prolog), Abfragesprachen (SQL) und Auszeichnungssprachen (HTML, CSS).

Häufige Fehler in dieser Aufgabe

- Ausschnitt 10 wird oft als imperativ eingestuft, weil `lambda` nach Code aussieht. Entscheidend ist aber: Es wird keine Schleife geschrieben und kein Zwischenzustand aufgebaut.
- Ausschnitt 1 und 2 werden korrekt getrennt, Ausschnitt 6 und 7 dann aber vergessen – dabei ist es dasselbe Muster mit einer Bedingung mehr.
- Ausschnitt 8 wird als imperativ eingestuft, weil er etwas verändert. Verändern und *Wie*-Beschreiben sind aber zwei verschiedene Dinge.

 © Kevin Maurizi

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
<https://wiki.bzz.ch/modul/m323/learningunits/lu01/loesungen/paradigmendetektiv>

Last update: **2026/08/18 10:18**

