

LU03e - Higher-Order Functions

Higher-Order Functions sind ein Schlüsselkonzept in der funktionalen Programmierung. Eine Higher-Order Function ist eine Funktion, die eine oder mehrere Funktionen als Argumente akzeptiert und/oder eine Funktion zurückgibt. Higher-Order Functions ermöglichen einen flexiblen, modularen und ausdrucksstarken Programmierstil.

Definition von Higher-Order Functions

Eine Higher-Order Function erfüllt mindestens eine der folgenden Bedingungen:

1. **Nimmt eine oder mehrere Funktionen als Argumente an.**
2. **Gibt eine Funktion als Ergebnis zurück.**

Beispiele für Higher-Order Functions

Im Folgenden sind einige Beispiele für Higher-Order Functions in Python dargestellt.

map: Eine eingebaute Higher-Order Function

Die `map`-Funktion in Python nimmt eine Funktion und eine Liste (oder ein anderes iterierbares Objekt) als Argumente und wendet die Funktion auf jedes Element der Liste an.

```
def square(x):
    return x * x

numbers = [1, 2, 3, 4]
squared_numbers = map(square, numbers)
print(list(squared_numbers)) # Ausgabe: [1, 4, 9, 16]
```

Funktionen, die Funktionen Zurückgeben

Man kann auch Higher-Order Functions erstellen, die Funktionen zurückgeben.

```
def multiplier(factor):
    def multiply_by_factor(number):
        return number * factor
    return multiply_by_factor

multiply_by_two = multiplier(2)
print(multiply_by_two(4)) # Ausgabe: 8
```

Bedeutung in der Funktionalen Programmierung

Higher-Order Functions sind grundlegend für die Funktionsweise der funktionalen Programmierung. Sie erlauben es, Funktionen als Bausteine zu verwenden, die in vielfältiger Weise zusammengesetzt werden können.

1. **Modularität:** Higher-Order Functions fördern die Trennung von Bedenken, da jede Funktion einen spezifischen Zweck erfüllen kann.
2. **Wiederverwendbarkeit:** Sie ermöglichen die Erstellung wiederverwendbarer und anpassbarer Codeblöcke.
3. **Ausdruckskraft:** Sie bieten einen eleganten Weg, komplexe Logik aufzubauen, indem sie einfachere Funktionen kombinieren.



Higher-Order Functions sind ein mächtiges Werkzeug in der funktionalen Programmierung. Sie nehmen Funktionen als Argumente, geben Funktionen zurück, oder beides, und ermöglichen damit einen hohen Grad an Flexibilität und Ausdrucksstärke im Code. Durch das Verständnis und die Verwendung von Higher-Order Functions kann ein Entwickler effizienteren und saubereren Code schreiben, der leichter zu verstehen und zu warten ist.

M323-LU03



© Kevin Maurizi

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

Permanent link:
<https://wiki.bzz.ch/modul/m323/learningunits/lu03/higherorderfunctions>

Last update: **2024/03/28 14:07**

