

# Lernnachweis zu Kompetenz B3E

## Verwendung von Lambda-Ausdrücken für den Programmfluss

### Sortieren von Listen mit benutzerdefinierten Kriterien

Die Kompetenz B3E beinhaltet die Verwendung von Lambda-Ausdrücken, um den Programmfluss zu steuern, z.B. durch Sortieren von Listen basierend auf benutzerdefinierten Kriterien. Dieser Lernnachweis verwendet Python, um die Anwendung von Lambda-Ausdrücken für die Sortierung zu zeigen.

#### Codebeispiel:

```
# Liste von Benutzern mit Namen und Alter
users = [
    {"name": "Alice", "age": 25},
    {"name": "Bob", "age": 30},
    {"name": "Charlie", "age": 22}
]

# Sortieren der Liste nach dem Alter mit Lambda-Ausdruck
sorted_users_age = sorted(users, key=lambda user: user["age"])

# Sortieren der Liste nach dem Namen mit Lambda-Ausdruck
sorted_users_name = sorted(users, key=lambda user: user["name"])

# Ausgabe der sortierten Listen
print("Sortiert nach Alter:", sorted_users_age)
print("Sortiert nach Namen:", sorted_users_name)
```

#### Erklärung:

In diesem Beispiel wird eine Liste von Benutzerobjekten erstellt, jedes mit einem Namen und einem Alter. Die «sorted»-Funktion wird verwendet, um die Liste nach benutzerdefinierten Kriterien zu sortieren, die durch Lambda-Ausdrücke definiert werden.

- Der Lambda-Ausdruck «lambda user: user["age"]» wird verwendet, um die Liste nach dem Alter der Benutzer zu sortieren.
- Der Lambda-Ausdruck «lambda user: user["name"]» wird verwendet, um die Liste nach den Namen der Benutzer zu sortieren.

Die Verwendung von Lambda-Ausdrücken in Verbindung mit der «sorted»-Funktion ermöglicht eine dynamische Steuerung des Sortierprozesses basierend auf benutzerdefinierten Kriterien.

Dieses Beispiel illustriert die praktische Anwendung der Kompetenz B3E durch die Verwendung von Lambda-Ausdrücken, um den Programmfluss, insbesondere die Sortierung von Listen, zu steuern.