

# Lernnachweis zu Kompetenz A1E

## Problemlösungsansätze in Objektorientierter, Prozeduraler und Funktionaler Programmierung in Python

**Ziel:** Vermittlung des Verständnisses, wie Probleme in den verschiedenen Programmierparadigmen in Python gelöst werden und deren Vergleich.

### Inhalt:

Die Lösung von Problemen in der Programmierung kann durch verschiedene Paradigmen angegangen werden, die in Python durch objektorientierte, prozedurale und funktionale Konzepte repräsentiert werden.

### Objektorientierte Programmierung (OO):

OO nutzt Klassen und Objekte, um Daten und Methoden zu kapseln. Hierbei werden Probleme durch Interaktion von Objekten gelöst.

```
class Car:
    def __init__(self, brand):
        self.brand = brand

    def drive(self):
        print(f"{self.brand} is driving.")

my_car = Car("Tesla")
my_car.drive()
```

### Prozedurale Programmierung:

Die prozedurale Programmierung verwendet Funktionen, die eine Reihe von Schritten ausführen. Sie strukturiert Code in ausführbare Prozeduren.

```
def drive_car(brand):
    print(f"{brand} is driving.")

drive_car("Tesla")
```

**Funktionale Programmierung:**

Funktionale Programmierung fokussiert auf unveränderliche Daten und reine Funktionen ohne Seiteneffekte. Sie zielt darauf ab, Probleme durch die Komposition von Funktionen zu lösen.

```
def drive():  
    return lambda brand: f"{brand} is driving."  
  
drive_car = drive()  
print(drive_car("Tesla"))
```

**Relevanz:**

Das Verständnis dieser Paradigmen ermöglicht es Entwicklern, die Stärken und Schwächen jedes Ansatzes zu erkennen und den geeignetsten für das jeweilige Problem auszuwählen. In Python ermöglicht die Vielseitigkeit, Elemente aus allen drei Paradigmen zu nutzen und so eine flexible und effiziente Problemlösung zu gestalten.