

Objektorientierte Programmierung

Ludwig Ettner

18. April 2024

Übersicht

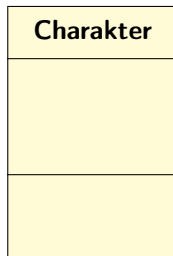
- 1 Einführung
- 2 UML
- 3 Objekte und Klassen
- 4 Attribute
- 5 Methoden/Funktionen
- 6 Vererbung
- 7 Beispiel: Implementierung in Python

- In dieser Präsentation werden wir die Grundlagen der objektorientierten Programmierung (OOP) untersuchen.
- OOP ist ein Paradigma zur Strukturierung von Programmcode, das auf dem Konzept von Objekten und Klassen basiert.
- Wir werden zunächst die Konzepte von Objekten und Klassen kennenlernen und dann anhand eines Beispiels Schritt für Schritt durchgehen.

Beispiel: Klasse Charakter

- Beispiel an einem Computerspiel
- Spielcharakter als Objekt
- Besitzt folgende Eigenschaften
 - Lebenspunkte
 - Position in der Welt
 - name des Charakters
- Kann folgende Aktionen ausführen
 - Bewegen
 - Angreifen

Beispiel: Klasse Charakter



Beispiel: Klasse Charakter

Charakter
- lebenspunkte: int

Beispiel: Klasse Charakter

Charakter
<ul style="list-style-type: none">- lebenspunkte: int- position: (x, y)

Beispiel: Klasse Charakter

Charakter
<ul style="list-style-type: none">- lebenspunkte: int- position: (x, y)- name: String

Beispiel: Klasse Charakter

Charakter
<ul style="list-style-type: none">- lebenspunkte: int- position: (x, y)- name: String
<ul style="list-style-type: none">+ bewegen(x: int, y: int)

Beispiel: Klasse Charakter

Charakter
<ul style="list-style-type: none">- lebenspunkte: int- position: (x, y)- name: String
<ul style="list-style-type: none">+ bewegen(x: int, y: int)+ angreifen(ziel: Charakter)

Definitionen

- **Klasse:** Ein Bauplan für die Erzeugung von Objekten. Definiert Attribute und Methoden.
- **Objekt:** Eine Instanz einer Klasse. Besitzt bestimmte Attribute und kann Methoden ausführen.

Python-Implementierung:

```
1 class Charakter:  
2     # Attribute und Funktionen  
3
```

Definitionen

- **Attribute:** Eigenschaften eines Objekts, die in der Klasse definiert werden.
- Diese Werte die das Objekt besitzt, können verändert werden.

Python-Implementierung:

```
1 class Charakter:
2     def __init__(self, name, lebenspunkte=100, position=(0,
3         # muss bei der Instanziierung uebergeben werden
4         self.name = name
5         # standartwert 100
6         self.lebenspunkte = lebenspunkte
7         # standartwert (0, 0)
8         self.position = position
9
```

Definitionen

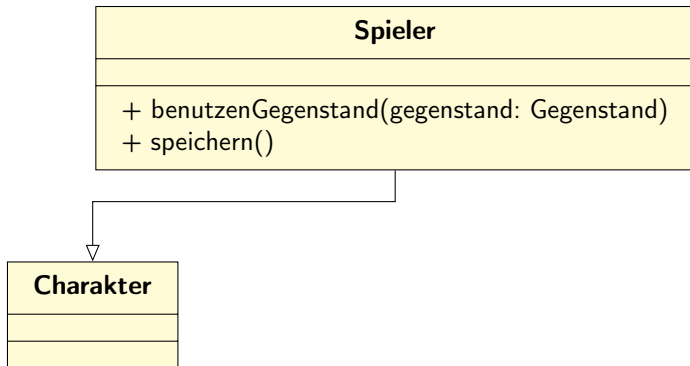
- **Mathoden:** Methoden sind Funktionen, die in einer Klasse definiert werden und auf Objekten dieser Klasse ausgeführt werden können.
- Sie können auf die Attribute des Objekts zugreifen und diese verändern.
- Methoden können auch Parameter entgegennehmen.

Python-Implementierung:

```
1 class Charakter:
2     def __init__(self, name, lebenspunkte=100, position=(0,
3         0)):
4         self.name = name
5         self.lebenspunkte = lebenspunkte
6         self.position = position
7
8     def bewegen(self, x, y):
9         self.position = (self.position[0] + x, self.position
10            [1] + y)
11
12     def angreifen(self, ziel):
13         # Implementierung des Angriffs
14         pass
```

Definition

- **Vererbung:** Ein Konzept, das es erlaubt, Eigenschaften und Verhalten einer bestehenden Klasse zu erben und zu erweitern.
- Die abgeleitete Klasse erbt die Attribute und Methoden der Basisklasse und kann eigene hinzufügen.
- Dies ermöglicht die Wiederverwendung von Code und die Strukturierung von Klassen.



Python-Implementierung:

- Definiere eine Klasse, die von einer anderen Klasse erbt.
- Füge spezifische Attribute und Methoden hinzu.

Vererbung (Fortsetzung)

Beispiel: Klasse Spieler

- Erbt von Klasse Charakter
- Spezifische Methoden: BenutzenGegenstand, Speichern

Python-Implementierung:

```
1 class Spieler(Charakter):  
2     def benutzen_gegenstand(self, gegenstand):  
3         # Implementierung der Benutzung eines Gegenstands  
4         pass  
5  
6     def speichern(self):  
7         # Implementierung des Speicherns des Spielstands  
8         pass  
9
```

Python-Klassen

- Python bietet eine einfache Syntax für die Definition von Klassen und Vererbung.
- Attribute und Methoden werden ähnlich wie in der UML-Syntax definiert.
- Zeigen Sie die Python-Implementierung für die Charakter- und Spielerklassen.
- Erläutern Sie die Syntax und die Verwendung von Konstruktoren, Attributen und Methoden.