

```
1 class Fahrrad():
2     def __init__(self, farbe, anzahlGaenge):
3         self.farbe = farbe
4         self.anzahlGaenge = anzahlGaenge
5     def fahren(self):
6         print("Das Fahrrad faehrt.")
7     def lackieren(self, neueFarbe):
8         self.farbe = neueFarbe
9 meinFahrrad = Fahrrad("rot", 5)
10 meinFahrrad.fahren()
11 meinFahrrad.lackieren("blau")
```



Definition von Klassen

In Python werden Klassen mit dem Keyword `class` gefolgt vom Namen der Klasse definiert. Alles, was zur Klasse gehört, wird darunter eingerückt.

Methoden der Klasse werden wie Funktionen mit dem Keyword `def` definiert, haben jedoch als ersten Parameter das Keyword `self`.

Auf Attribute der Klasse kann mit `self.variablenname` zugegriffen werden.

Jede Klasse hat eine besondere Methode – den sogenannten Konstruktor. Der Konstruktor ist eine Methode, die beim Erstellen eines Objekts aufgerufen wird. Hier werden beispielsweise die Werte der Attribute eines neuen Objekts festgelegt. In Python erkennt man den Konstruktor am Methodennamen `__init__` – im obigen Beispiel in Zeile 2.



Verwenden von Objekten

Nachdem eine Klasse definiert wurde, können Objekte dieser Klasse erstellt werden. Dazu wird der Name der Klasse mit Klammern geschrieben, in welche als Parameter die Parameter des Konstruktors übergeben werden. Im obigen Code sieht man dies in Zeile 9.

Die Methoden des neu erstellten Objekts können nun durch `variablenname.methodenname()` aufgerufen werden. Mögliche Parameter der Methoden müssen dabei mit übergeben werden, wie beispielsweise in Zeile 11.