

In PHP unterscheiden wir zwischen prozeduraler und objektorientierter Programmierung. Bisher haben wir ausschließlich den prozeduralen Weg kennen gelernt. Jetzt werden wir uns mit der Objektorientierten Programmierung (OOP) auseinandersetzen.

Dafür benötigen wir zuerst drei Begriffe: Klasse, Eigenschaft und Methode. In der OOP wird immer eine Klasse mit dem Schlüsselwort `class` gebildet. Diese kann dann als Objekt mit `new` instanziiert werden. Mit dem Objekt hat man dann Zugriff auf die Eigenschaften (vgl. Variablen) und auf die Methoden (vgl. Funktionen).

OOP strukturiert den Code besser und macht komplexere Projekte damit übersichtlicher. Darüber hinaus ist es das ideale Mittel, wenn mehrere Personen an einem Code arbeiten. Objekte lassen sich auch einfach serialisieren (z. B. auch mit JSON).

PHP



OOP Beispiel: Bankkonto

Mit `class` wird die Klasse `Konto` erstellt. Diese hat zwei Eigenschaften (`$art` und `$kontostand`) – `public` bedeutet, dass man auch von außen darauf zugreifen kann. Die Methode `einzahlen` übernimmt einen Betrag.

Mit `$this` wird auf die Eigenschaften der Klasse zugegriffen und der Betrag hinzuaddiert. Über `return` bekommt die Methode eine Ausgabe.

```
<?php
class Konto {
    public $art = "Konto";
    public $kontostand = 0;

    function einzahlen($betrag) {
        $this->kontostand = $this->kontostand + $betrag;
        return "<br>Am " . $this->art .
            " sind " . $this->kontostand;
    }
}
?>
```

PHP



OOP Beispiel: Klasse instanziiieren

Mit dem Schlüsselwort `new` und dem Namen der Klasse werden neue Objekte vereinbart. Die Zeichen `->` sollen einen Pfeil darstellen.

Mit `->` greift man auf Eigenschaften und Methoden des Objektes zu.

```
<?php
$gruber = new Konto();
$gruber->art = "Girokonto";
echo $gruber->einzahlen(1500);

$maier = new Konto();
$maier->art = "Sparbuch";
$maier->einzahlen(200);
echo '<br>Kontostand: ' . $maier->kontostand;
?>
```