

# SCHOOL-SCOUT.DE



Unterrichtsmaterialien in digitaler und in gedruckter Form

## Auszug aus:

*Lehrkonzept: PRIMM im Programmierunterricht*

Das komplette Material finden Sie hier:

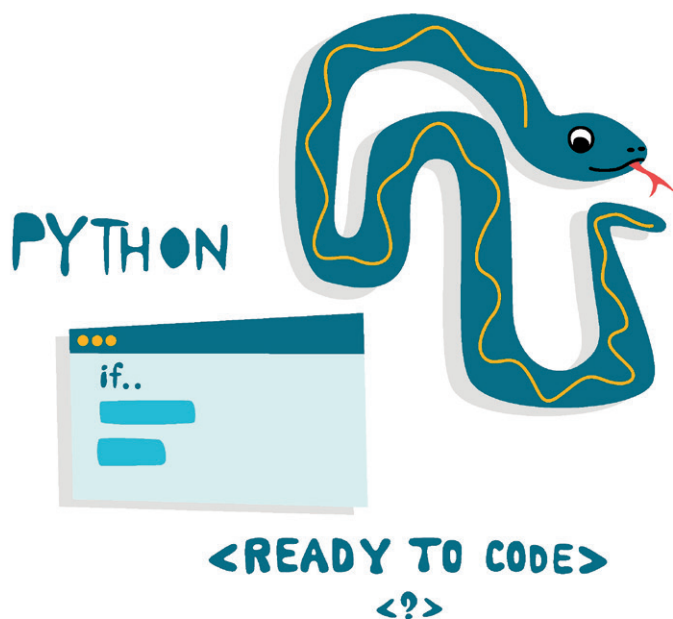
[School-Scout.de](https://www.school-scout.de)



## Inhaltsfeld Algorithmen

# Programmierunterricht mit dem PRIMM-Lehrkonzept strukturieren – Umsetzungsbeispiel mit *Python*

Ein Beitrag von Jonathan Pfeiffer



© RAABE 2021

© Elena Karaseva /iStock/Getty Images Plus

Gerade für Programmieranfängerinnen und -anfänger ist es oftmals nicht einfach motiviert und erfolgreich am Ball zu bleiben. Ein in der praktischen Unterrichtsumgebung bewährter Ansatz ist das sogenannte PRIMM-Lehrkonzept zur Strukturierung Ihres Programmierunterrichts. Dieser Ansatz setzt das Quellcode lesen vor das Quellcode schreiben. Kollaboratives Lernen wird in den Fokus gesetzt, damit die Lernenden über die Programme und Konzepte sprechen. Dabei wird auf eine Reduzierung der kognitiven Belastung der Programmieranfängerinnen und -anfänger geachtet, indem sich die Lernenden schrittweise und allmählich ein Programm zu eigen machen. Dieser Beitrag stellt Ihnen das PRIMM-Konzept mit kleinen Praxisbeispielen innerhalb der Programmierumgebung *Python* für die Umsetzung im Unterricht vor.

---

### METHODIK & DIDAKTIK

<b>Klassenstufe:</b>	7–10 + Sek. II
<b>Thematische Bereiche:</b>	Algorithmen, Programmieren, Programmierunterricht, Umsetzung an Beispielen in <i>Python</i>
<b>Kompetenzbereiche:</b>	Implementieren, Kooperieren und Kommunizieren
<b>Methodischer Ansatz:</b>	<i>Use-Modify-Create</i> -Konzept, PRIMM-Konzept
<b>Sozialform:</b>	Partnerarbeit

---

## Symbolerklärungen

	Diese Symbole markieren eine Einzel-, Partner- bzw. Gruppenarbeit.
	Diese Symbole markieren alternative Durchführungsmöglichkeiten bzw. alternative Durchführungsmöglichkeiten nach Niveaustufen.
	Tauchen diese Symbole auf, handelt es sich um binnendifferenzierte Materialien.
	Dieses Symbol markiert Materialien auf einfacherem G-Niveau bzw. Materialien eher für niedrigere Klassenstufen.
	Dieses Symbol markiert Materialien auf Normalniveau (M-Niveau).
	Dieses Symbol markiert Materialien auf höherem E-Niveau bzw. Materialien eher für höhere Klassenstufen oder Exkursmaterialien.
	Dieses Symbol markiert Hilfestellungen bzw. Tipps.
	Dieses Symbol markiert Zusatzaufgaben für schnelle Lernende.
	Dieses Symbol markiert Merkkästen und wichtige Inhalte.
	Dieses Symbol markiert am Laptop/PC zu bearbeitende Aufgaben.
	Dieses Symbol taucht auf, wenn ein Dateidownload notwendig ist.
	Dieses Symbol markiert interaktive Aufgaben zur Bearbeitung mit einem digitalen Endgerät.
	Dieses Symbol markiert das Einbinden eines Videos/Films.
	Dieses Symbol markiert eine Internetrecherche.
	Dieses Symbol taucht auf, wenn näher recherchiert werden soll oder tiefgreifende Informationen hinterlegt sind.
	Diese Symbole markieren Pro- und Kontraargumente bzw. eine Diskussion.
	Dieses Symbol markiert Aufgaben zum Ankreuzen.
	Dieses Symbol markiert Aufgaben, bei denen gerechnet werden muss.

---

## Auf einen Blick

---

### Definition PRIMM

Seite 4

---

### *Use-Modify-Create*-Lehrkonzept als Grundlage

Seite 4–5

---

### Empfohlene Sozialform

Seite 6

---

### Umsetzungsbeispiel des PRIMM-Ansatzes im Unterricht

Seite 7–15

Seite 7

Vorwissen für dieses Thema

Seite 7

Methodisch-didaktische Überlegungen zu den Materialien

Seite 8–9

M 1: Alternative (if-else-Verzweigung) in Python –  
Programm verwenden und untersuchen

Seite 10–11

M 2: Alternative (if-else-Verzweigung) in Python –  
Programm modifizieren I

Seite 12

M 3: Alternative (if-else-Verzweigung) in Python –  
Programm modifizieren II

Seite 13

M 4: Alternative (if-else-Verzweigung) in Python –  
Eigenes Programm erstellen

Seite 14–15

Lösungen

---

### Weiterführende Medien

Seite 16

## Definition PRIMM

In der nationalen und internationalen methodisch-didaktischen Fachliteratur finden sich verschiedene Frameworks und didaktische Ansätze, die sich mit der Entwicklung der Schülerinnen und Schüler beim Programmieren Lernen auseinandersetzen. In diesem Beitrag soll PRIMM als ein sinnvoller Ansatz zur guten Strukturierung des Informatikunterrichts, insbesondere des Programmierunterrichts gerade für Programmieranfängerinnen und -anfänger, vorgestellt werden.



Die Abkürzung **PRIMM** steht für:

- P = *predict* (vorhersagen)
- R = *run* (ausführen)
- I = *investigate* (untersuchen)
- M = *modify* (anpassen)
- M = *make* (machen)

## Use-Modify-Create-Lehrkonzept als Grundlage

Der PRIMM-Ansatz bezieht sich auf das **Use-Modify-Create-Lehrkonzept** (Lee et al. 2011, 35). Dieses gliedert sich in die drei Phasen *Use* (Nutzen), *Modify* (Modifizieren) und *Create* (Gestalten).



### Use-Phase

In der *Use*-Phase, die im PRIMM-Ansatz *predict*, *run* und *investigate* entspricht, sind die Schülerinnen und Schüler Konsumenten der von jemand anders erstellten „Kreationen“. D.h. die Programmieranfängerinnen und -anfänger arbeiten in dieser ersten Phase zunächst mit bereits bestehenden und fertigen Programmen, nutzen also Programme, an deren Erstellungsprozess sie unbeteiligt sind („nicht meins“). Es kann sich dabei auch um die Anwendung bestehender Computermodelle oder ein fertig erstelltes Computerspiel handeln. Das entlastet die Lernenden in vielerlei Hinsicht. Da es „nicht meins“ ist, besteht keine emotionale Beziehung zu diesen Programmen. Die kognitive Belastung beim Programmieren (Problem verstehen, Algorithmus entwickeln, Fehlermeldungen interpretieren, Fehler suchen usw.) wird reduziert, weil es zunächst einmal nur um das Nutzen, Ausprobieren, Analysieren und Verstehen geht. Zudem kann der Fokus auf ausgewählte Aspekte gelegt werden, z. B. Syntax, Reihenfolge, häufige Fehlerquellen usw. Haben die Lernenden das Programm ausreichend z. B. mithilfe einer Syntax-Checkliste, eines Fehlerprotokolls oder geeigneten Analyse-Fragen (Was ist neu? Was bewirkt der neue Bestandteil?) untersucht und das zugrundeliegende Konzept (zumindest teilweise) verstanden, dann können (und wollen) sie das Programm verändern und modifizieren. Meist entwickeln die Lernenden selbst bei der Anwendung des bereits bestehenden Programms die Motivation Veränderungen vorzunehmen. Dies können zunächst kleine Änderungen, wie z. B. bei einem Videospiel die eingestellte Hintergrundfarbe sein, was sich aber mit zunehmenden Programmierkenntnissen mehr und mehr ausbauen kann. Wichtig ist, dass die Herausforderungen nur allmählich gesteigert werden sollten. So kann verhindert werden, dass die Schülerinnen und Schüler vor einer Aufgabe Angst haben müssen und sie fühlen sich den nach und nach kommenden Herausforderungen stets gewachsen.

# SCHOOL-SCOUT.DE



Unterrichtsmaterialien in digitaler und in gedruckter Form

## Auszug aus:

*Lehrkonzept: PRIMM im Programmierunterricht*

Das komplette Material finden Sie hier:

[School-Scout.de](https://www.school-scout.de)

