

### 💡 Sets in Python

Ähnlich wie Listen können wir in Python ein **Set** festlegen. Das Konzept des Sets ist aus mathematischer Perspektive eine Menge.

Die Syntax ist exakt dieselbe wie bei den Listen, mit dem Unterschied, dass geschweifte Klammern statt eckigen Klammern verwendet werden: `set1 = {1, 2, 3}`

Elemente können auch nachträglich zu einem Set hinzugefügt (`set1.add(element)`) oder aus einem Set entfernt werden (`set1.remove(element)`).

### 👥 Aufgabe 1: Eigenschaften von Sets

Legen Sie Sets und Listen an und vergleichen Sie diese miteinander. Füllen Sie dazu die folgende Tabelle aus.

Eigenschaft	Liste	Set
indizierbar?  ( <code>set1[0]</code> bzw. <code>list1[0]</code> )		
Reihenfolge relevant?		
Duplikate möglich?		

## Aufgabe 2: Mengenoperationen

Legen Sie zwei Sets `set1 = {1, 2, 3, 4, "apple", "banana"}` und `set2 = {3, 4, 5, 6, "banana", "cherry"}` an.

Finden Sie heraus, wie die verschiedenen Operationen die beiden Sets miteinander verknüpfen:

Operation	Beschreibung
<code>set1 - set2</code>	
<code>set1 &amp; set2</code>	
<code>set1   set2</code>	
<code>set1 ^ set2</code>	
<code>set1 &lt; set2</code>	

## Aufgabe 3: Verschiedene Wörter in Sätzen

Schreiben Sie ein Programm, das einen Satz vom Nutzer einliest. Geben Sie dann alle eingegebenen Wörter, die *unterschiedlich* sind, nacheinander aus. Geben Sie auch aus, wie viele Wörter es waren.

Verwenden Sie zur Lösung ein Set.