

Ziel:

Du lernst den Unterschied zwischen **Listen** und **Tupeln** kennen. Beide speichern mehrere Werte, aber sie verhalten sich unterschiedlich.

Beispiel 1: Liste

```
1 # Eine Liste ist veränderbar
2 farben = ["rot", "blau", "grün"]
3
4 # Wir geben die Liste aus
5 print(farben)
6
7 # Wir ändern das erste Element(0)
8 farben[0] = "gelb"
9 print(farben)
```

Erklärung:

- Listen verwendet man, wenn man etwas ändern will.
- `farben[0]` ist das erste Element. Mit `=` können wir es ändern.
- `print(farben)` zeigt die aktuelle Liste an

Aufgabe 1:

1. Erstelle eine Liste mit deinen drei Lieblingsobstsorten.
2. Ändere das zweite Element der Liste.
3. Gib die Liste vorher und nachher mit `print()` aus.
4. Speichere das Programm mit dem Namen **Liste**.

Beispiel 2: Tupel

```
1 # Ein Tupel ist unveränderbar
2 person = ("Anna", 25)
3
4 # Wir geben das Tupel aus
5 print(person)
```

Erklärung:

- Tupel verwendet man, wenn sich die Werte nicht ändern sollen.
- Sie sind sicherer, weil niemand sie aus Versehen verändert.
- Typische Beispiele sind Name und Geburtsjahr, Koordinaten oder andere feste Daten

Aufgabe 2:

1. Erstelle ein Tupel mit deinem Namen und deinem Alter.
2. Gib das Tupel mit `print()` aus.
3. Überlege: Was passiert, wenn du versuchst, das Alter im Tupel zu ändern?

Zusammenfassung:

Merkmal	Liste	Tupel	Beispiel
Klammern	[]	()	["rot", "blau"] / ("Anna", 25)
Veränderbar	Ja	Nein	
Verwendung	Wenn Daten sich ändern	Wenn Daten gleich bleiben	

Weitere Aufgaben: Ü 17, Ü18, Ü19a,b**Ü19b**

Hier ist das Tupel mit den Koordinaten ist in der Variablen `punkt` gespeichert.

`punkt` ist ein **Tupel** mit zwei Werten:

- `punkt[0]` → X-Koordinate
- `punkt[1]` → Y-Koordinate

Um nur die X-Koordinate auszugeben, rufst du `print` mit `punkt[0]` auf. Dadurch erscheint hinter dem Text "X-Koordinate:" ausschließlich der X-Wert.