

Zwei Orte vergleichen

Workshoptag 5 • Arbeitsblatt 5, Aufgabe 3

Worum es geht

Zwei Kurven in ein Bild zu legen ist technisch leicht: zweimal `plt.plot()`. Das Schwierige steht davor. Ein Vergleich ist nur dann einer, wenn sich die beiden Messreihen wirklich vergleichen lassen.

Die eine Regel

Vergleichbar ist, was **gleichzeitig** und **gleich** gemessen wurde. Verschiedene Zeiträume, verschiedene Sensoren oder verschiedene Messgrößen ergeben zwei Kurven in einem Bild, aber keinen Vergleich.

Erst prüfen, dann zeichnen

Was	Warum
derselbe Zeitraum	Sonst vergleichst du eine kalte Nacht mit einem warmen Abend. <code>from</code> und <code>to</code> für beide Anfragen gleich setzen.
dieselbe Messgröße	<code>measurand</code> muss übereinstimmen, und zwar buchstabengenau.
dieselbe Einheit	<code>unit</code> vergleichen. Ein Druck in Pa und einer in hPa unterscheiden sich um den Faktor 100.
derselbe Sensortyp	Zwei BME280 sind vergleichbar. Ein BME280 und ein anderer Sensor messen dasselbe mit unterschiedlicher Genauigkeit.

Zwei Anfragen

Zweimal derselbe Block, und nur zwei Dinge unterscheiden sich: der Ort und der Name der Ergebnisliste.

```
# Erster Ort
antwort = requests.get(
    ADRESSE,
    headers={"Authorization": "Bearer " + TOKEN},
    params={"measurand": "Temperatur", "location": "Raum 101",
           "per_page": 100,
           "from": "2026-09-10T20:00:00+02:00",
           "to": "2026-09-11T07:00:00+02:00"},
)
messwerte = antwort.json()["data"]
messwerte.reverse()

raum = []
for messwert in messwerte:
    raum.append(float(messwert["value"]))
```

```
# Zweiter Ort
antwort = requests.get(
    ADRESSE,
    headers={"Authorization": "Bearer " + TOKEN},
    params={"measurand": "Temperatur", "location": "Flur",
            "per_page": 100,
            "from": "2026-09-10T20:00:00+02:00",
            "to": "2026-09-11T07:00:00+02:00"},
)
messwerte = antwort.json()["data"]
messwerte.reverse()

flur = []
for messwert in messwerte:
    flur.append(float(messwert["value"]))
```

Hier entstehen die Fehler

Dieselben Zeilen zweimal hinzuschreiben und an einer Stelle etwas zu ändern, ist die häufigste Art, sich einen Fehler einzubauen. Am besten kopierst du den ersten Block und änderst im zweiten genau zwei Dinge: location und den Namen der Liste.

Danach beide Blöcke nebeneinanderlegen und Zeile für Zeile vergleichen. from, to und measurand **müssen** übereinstimmen, sonst ist es kein Vergleich.

antwort **und** messwerte **kommen zweimal vor**

Das ist kein Versehen. Der zweite Block überschreibt beide Variablen, und das ist in Ordnung: Was du behalten willst, steht danach schon in raum beziehungsweise flur.

Zeichnen

```
plt.plot(raum, label="Raum 101")
plt.plot(flur, label="Flur")

plt.legend()
plt.title("Zwei Orte in derselben Nacht")
plt.xlabel("Messung")
plt.ylabel("Grad Celsius")
plt.show()
```

label hängt jeder Kurve einen Namen an, legend() sammelt alle Namen ein und zeichnet die Legende. Ohne legend() bleiben die Namen unsichtbar.

Wenn die Kurven unterschiedlich lang sind

Das ist der Normalfall: Zwei Nodes senden nicht im Gleichtakt, und eine fällt vielleicht früher aus. Über der Nummer der Messung aufgetragen laufen die Kurven dann auseinander, obwohl sie zur selben Zeit gehören.

Hier hilft nur die echte Zeitachse

Trag beide über zeiten auf statt über die Nummer, dann liegen die Zeitpunkte übereinander, egal wie viele Messungen dazwischen liegen. Wie das geht, steht auf Ausbaustufe b.

Und dann die eigentliche Frage

Wenn beide Kurven da sind: Ist der Unterschied größer als das, was die Sensoren ohnehin streuen? Ein BME280 gibt die Temperatur auf etwa ein halbes Grad genau an. Zwei Kurven, die ein halbes Grad auseinanderliegen, zeigen vielleicht gar keinen Unterschied zwischen den Orten.