

1 | Strings & Index

- String ist eine Zeichenfolge.
- Index beginnt bei 0; negative Indizes zählen rückwärts.

```
text = "Python"
print(text[0])  # P
print(text[-1]) # n
```

2 | Slicing

- Teiltext mit [start:ende:schritt].
- Strings selbst sind unveränderlich.

```
text = "OpenSCAD"
print(text[0:4])
print(text[:: -1])
```

3 | Länge & Mitgliedschaft

- len zählt Zeichen.
- in prüft Teiltext; Vergleich ist groß/klein-sensitiv.

```
print(len(text))
if "CAD" in text:
    print("gefunden")
```

4 | Rand bereinigen

- strip entfernt Leerraum außen.
- lstrip/rstrip nur links/rechts.
- Entfernt nicht automatisch innen.

```
eingabe = "  PLA  "
sauber = eingabe.strip()
```

5 | Groß-/Kleinschreibung

- lower und upper ändern Darstellung.
- casefold ist robust für Vergleiche.

```
antwort = input("Ja/Nein: ").strip().casefold()
if antwort == "ja":
    print("OK")
```

6 | split() & join()

- split zerlegt in Liste.
- join verbindet Strings mit Trennzeichen.

```
zeile = "PLA;205;60"
felder = zeile.split(";")
neu = " | ".join(felder)
```

7 | replace()

- Ersetzt alle Treffer und liefert neuen String.
- Original bleibt unverändert.

```
dateiname = "Mein Teil.stl"
sauber = dateiname.replace(" ", "_")
```

8 | Suchen

- find liefert Index oder -1.
- startswith/endswith prüfen Anfang/Ende.

```
if datei.endswith(".csv"):
    print("CSV-Datei")
pos = text.find("CAD")
```

9 | f-Strings

- Werte und Ausdrücke direkt einsetzen.
- Formatangaben folgen nach Doppelpunkt.

```
name = "PLA"
temp = 205.456
print(f"{name}: {temp:.1f} °C")
```

10 | Zeichen ersetzen/entfernen

- Mehrstufige Bereinigung bewusst dokumentieren.
- Für komplexe Muster später reguläre Ausdrücke.

```
sauber = text.replace("/", "-")
sauber = sauber.replace(":", "-")
```

11 | Typische Fehler

Problem	Ursache
IndexError	Position außerhalb
ändert sich nicht	Ergebnis nicht zugewiesen
split falsch	Trennzeichen
Zahl + Text	str/f-String
Vergleich scheitert	strip/casefold

12 | Mini-Übung: Dateiname

Aufgabe

Bereinige einen Namen: Rand entfernen, klein schreiben, Leerzeichen ersetzen.

Musterlösung

```
name = " Mein Modell V2 "
sauber = name.strip().lower()
sauber = sauber.replace(" ", "_")
print(sauber + ".stl")
```