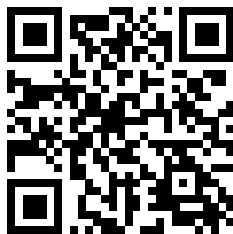


Python Grundlagen

Variablen und Datentypen



Google Colab

Lernziele

Am Ende dieser Lektion koennen Sie:

- ✓ Variablen erstellen und verwenden
- ✓ Datentypen unterscheiden
- ✓ Mit Zahlen rechnen
- ✓ Datentypen umwandeln
- ✓ Namenskonventionen anwenden

Was ist eine Variable?

Variable = Name fuer einen Wert im Speicher

```
1 name = "Anna"  
2 alter = 25  
3 groesse = 1.75  
4 ist_student = True
```

Dynamische Typisierung:

Python erkennt den Typ automatisch!

Achtung: Text braucht Anfuehrungszeichen!

Falsch:

```
1 name = Anna
```

Fehlermeldung:

```
NameError: name 'Anna' is not defined
```

Richtig:

```
1 name = "Anna"
```

Namenskonventionen

Regeln fuer Variablennamen:

✓ Erlaubt:

```
1 name = "Alice"  
2 user_age = 25  
3 _temp_value = 10  
4 _lname = "Bob"
```

✗ Nicht erlaubt:

```
1 lname = "Bob"  
2 user-age = 25  
3 my name = "Max"
```

Best Practice: snake_case verwenden

Mini-Uebung 1: Erste Variablen

► Jetzt sind Sie dran!

Erstellen Sie Variablen:

1. Ihr Name (String)
2. Ihr Alter (Integer)
3. Ihre Groesse in Metern (Float)
4. Sind Sie Student? (Boolean)

Geben Sie alle Variablen mit `print()` aus!

Loesung: Mini-Uebung 1

✓ Loesung

```
1  # Variablen erstellen
2  name = "Max"
3  alter = 25
4  groesse = 1.80
5  ist_student = True
6
7  # Ausgeben
8  print("Name:", name)
9  print("Alter:", alter)
10 print("Groesse:", groesse)
11 print("Student:", ist_student)
```


Mehrfachzuweisung

Mehrere Variablen gleichzeitig zuweisen:

```
1 x, y, z = 10, 20, 30
```

Werte tauschen:

```
1 a = 5  
2 b = 15  
3 print("Vor dem Tausch:", a, b)  
4  
5 a, b = b, a  
6 print("Nach dem Tausch:", a, b)
```

Datentyp: String (str)

String = Zeichenkette = Text

```
1 text1 = 'Hallo Welt'
2 text2 = "Python ist grossartig"
3 text3 = '''Dies ist
4 ein mehrzeiliger
5 Text'''
```

! Achtung

Strings sind **unveraenderlich** (immutable)!

Datentyp: Integer (int)

Integer = Ganzzahl

```
1 alter = 25
2 temperatur = -5
3 anzahl = 0
```

Typ pruefen:

```
1 print(type(alter))
```

```
<class 'int'>
```

Rechenoperatoren

```
1 a = 10
2 b = 3
3
4 print(a + b)      # Addition: 13
5 print(a - b)      # Subtraktion: 7
6 print(a * b)      # Multiplikation: 30
7 print(a / b)      # Division: 3.3333...
8 print(a // b)     # Ganzzahldivision: 3
9 print(a % b)      # Modulo (Rest): 1
10 print(a ** b)    # Potenz: 1000
```

Mini-Uebung 2: Rechnen

▷ Uebung

Berechnen Sie:

1. $15 + 27$

2. $100 - 47$

3. 8×7

4. $20 \div 3$ (Ganzzahldivision)

5. $20 \bmod 3$ (Rest)

6. 2^{10}

Loesung: Mini-Uebung 2

✓ Loesung

```
1 print(15 + 27)      # 42
2 print(100 - 47)     # 53
3 print(8 * 7)        # 56
4 print(20 // 3)      # 6
5 print(20 % 3)       # 2
6 print(2 ** 10)      # 1024
```

Datentyp: Float (float)

Float = Flieskommazahl = Dezimalzahl

```
1 pi = 3.14159
2 temperatur = -5.5
3 zero_float = 0.0
```

! Achtung

Python verwendet Punkt, nicht Komma!

3.14 ✓ 3,14 ✗

Die round() Funktion

Zahlen runden:

```
1 print(round(3.7))           # 4
2 print(round(3.3))           # 3
3 print(round(-3.7))          # -4
4
5 # Auf Nachkommastellen runden
6 print(round(3.14159, 2))     # 3.14
7 print(round(3.14159, 3))     # 3.142
```


Rechnen mit Pi

Das math-Modul:

```
1 import math
2
3 pi = math.pi
4 print(pi)
```

```
3.141592653589793
```

Kreisumfang berechnen:

```
1 radius = 5
2 umfang = 2 * math.pi * radius
3 print("Umfang:", round(umfang, 2))
```

Mini-Uebung 3: Kreisberechnung

▷ Uebung

Berechnen Sie fuer einen Kreis mit Radius 7:

1. Den Umfang: $U = 2\pi r$
2. Die Flaeche: $A = \pi r^2$

Runden Sie beide Ergebnisse auf 2 Nachkommastellen!

Loesung: Mini-Uebung 3

✓ Loesung

```
1 import math
2
3 radius = 7
4
5 # Umfang berechnen
6 umfang = 2 * math.pi * radius
7 print("Umfang:", round(umfang, 2))
8
9 # Flaeche berechnen
10 flaeche = math.pi * radius ** 2
11 print("Flaeche:", round(flaeche, 2))
```

Datentyp: Boolean (bool)

Boolean = Wahrheitswert

Nur zwei Werte: True oder False

```
1 x = 10
2 y = 20
3
4 print(x == y)    # False (gleich?)
5 print(x != y)    # True (ungleich?)
6 print(x > y)     # False (groesser?)
7 print(x < y)     # True (kleiner?)
```

Vergleichsoperatoren

Operator	Bedeutung
==	gleich
!=	ungleich
>	groesser
<	kleiner
>=	groesser oder gleich
<=	kleiner oder gleich

Die type() Funktion

Datentyp herausfinden:

```
1 name = "Anna"
2 alter = 25
3 groesse = 1.75
4 ist_student = True
5
6 print(type(name))          # <class 'str'>
7 print(type(alter))         # <class 'int'>
8 print(type(groesse))      # <class 'float'>
9 print(type(ist_student))  # <class 'bool'>
```

Typkonvertierung

Datentypen umwandeln:

```
1  # String zu Integer  
2  alter = int("25")  
3  
4  # String zu Float  
5  groesse = float("1.75")  
6  
7  # Integer zu String  
8  text = str(25)  
9  
10 # Float zu Integer (rundet ab!)  
11 zahl = int(3.9)  # Ergebnis: 3
```

input() mit Typkonvertierung

! Achtung

input() gibt **immer** einen String zurueck!

```
1  # Falsch - funktioniert nicht!  
2  alter = input("Alter: ")  
3  print("In 2 Jahren:", alter + 2)  # Fehler!
```

```
1  # Richtig - mit Konvertierung  
2  alter = int(input("Alter: "))  
3  print("In 2 Jahren:", alter + 2)
```


Mini-Uebung 4: Temperaturumrechnung

▷ Uebung

Programm zur Temperaturumrechnung:

1. Temperatur in Celsius einlesen
2. In Fahrenheit umrechnen:
$$F = C \times 1.8 + 32$$
3. Ergebnis ausgeben

Beispiel: $20C = 68F$

Loesung: Mini-Uebung 4

✓ Loesung

```
1  # Celsius einlesen
2  celsius = float(input("Temperatur in Celsius: "))
3
4  # Umrechnen
5  fahrenheit = celsius * 1.8 + 32
6
7  # Ausgeben
8  print("Temperatur in Fahrenheit:", fahrenheit)
9  print("Gerundet:", round(fahrenheit, 1))
```

Haeufiger Fehler 1

Komma statt Punkt

```
1 groesse = 1,75  # Falsch!
```

```
1 groesse = 1.75  # Richtig!
```

Merke: Python nutzt den Punkt als Dezimaltrennzeichen!

Haeufiger Fehler 2

Falsche Zuweisung

```
1 radius == 5    # Falsch! (Vergleich)
```

```
1 radius = 5     # Richtig! (Zuweisung)
```

Merke: = zuweisen, == vergleichen!

Häufiger Fehler 3

Fehlende Typkonvertierung

```
1 zahl = input("Zahl: ")
2 ergebnis = zahl + 10    # Fehler!
```

```
1 zahl = int(input("Zahl: "))
2 ergebnis = zahl + 10    # Richtig!
```

Grosse Uebung: BMI-Rechner

► Praxisaufgabe

Erstellen Sie einen BMI-Rechner:

1. Gewicht in kg einlesen (int)
2. Groesse in m einlesen (float)
3. BMI berechnen: $\frac{\text{Gewicht}}{\text{Groesse}^2}$
4. Ergebnis gerundet ausgeben

Beispiel: 80 kg, 1.80 m → BMI = 24.69

Loesung: Grosse Uebung

✓ Loesung

```
1  # Eingaben
2  gewicht = int(input("Gewicht in kg: "))
3  groesse = float(input("Groesse in m: "))
4
5  # BMI berechnen
6  bmi = gewicht / (groesse ** 2)
7
8  # Ausgeben
9  print("Ihr BMI:", bmi)
10 print("Gerundet:", round(bmi, 2))
```

Fehlersuche 1: Finde die Fehler!

▷ Debugging-Uebung

```
1 name = "Peter"  
2 alter = 25  
3 gr e_in_meter = 1.75  
4 ist_student = True
```

Was ist hier falsch?

Loesung: Fehlersuche 1

✓ Loesung

Fehler: ö im Variablennamen

```
1  # Richtig:
2  name = "Peter"
3  alter = 25
4  groesse_in_meter = 1.75  # Keine Umlaute!
5  ist_student = True
```

Fehlersuche 2: Kreisberechnung

▷ Debugging-Uebung

```
1 import math
2
3 radius == 5
4 umfang = 2 + math.pi * radius
5 flaeche = math.pi * (radius)2
6
7 print("Umfang:" umfang)
8 print("Flaeche" flaeche)
```

Finden Sie alle Fehler!

Loesung: Fehlersuche 2

✓ Loesung

```
1 import math
2
3 radius = 5                # = statt ==
4 umfang = 2 * math.pi * radius # * statt +
5 flaeche = math.pi * (radius ** 2) # ** 2
6
7 print("Umfang:", umfang)    # Komma fehlt
8 print("Flaeche:", flaeche) # Komma fehlt
```

Cheat Sheet: Datentypen

Typ	Beispiel	Verwendung
str	"Hallo"	Text
int	25	Ganzzahlen
float	3.14	Dezimalzahlen
bool	True	Wahrheitswerte

Wichtige Funktionen:

- ▶ `type(x)` - Typ anzeigen
- ▶ `int(x)`, `float(x)`, `str(x)` - Konvertieren
- ▶ `round(x, n)` - Runden

Heute gelernt:

- ▶ Variablen erstellen und benennen
- ▶ Vier Datentypen: str, int, float, bool
- ▶ Rechenoperationen
- ▶ Typkonvertierung
- ▶ Häufige Fehler vermeiden

Hausaufgaben

Ueben Sie zu Hause:

1. Rechner fuer Rechteckflaeche
2. Waehrungsumrechner (Euro \leftrightarrow Dollar)
3. Altersrechner in Tagen
4. Durchschnittsrechner fuer 3 Zahlen