

Numerische Datentypen in Python

Einführung

Python unterstützt verschiedene numerische Datentypen, die für mathematische Berechnungen in Programmen verwendet werden können. Zu den grundlegenden numerischen Typen gehören:

- **Ganzzahlen (`int`)**: Dieser Datentyp repräsentiert ganze Zahlen, sowohl positive als auch negative.
- **Gleitkommazahlen (`float`)**: Dieser Typ wird für reelle Zahlen mit Dezimalstellen verwendet.
- **Komplexe Zahlen (`complex`)**: Komplexe Zahlen bestehen aus einem realen und einem imaginären Teil, z.B. `3+5j`.

Verwendung

Ganzzahlen (`int`)

Ganzzahlen werden häufig für Zählvorgänge, Indizierung und andere Operationen verwendet, bei denen Dezimalstellen nicht benötigt werden.

```
anzahl_der_artikel = 5
temperatur = -20
```

Gleitkommazahlen (`float`)

Gleitkommazahlen sind nützlich für präzisere Berechnungen, bei denen Dezimalstellen wichtig sind.

```
preis = 19.99
durchschnittstemperatur = 36.5
```

Komplexe Zahlen (`complex`)

Komplexe Zahlen werden seltener verwendet und sind hauptsächlich in speziellen mathematischen Anwendungen relevant.

```
komplexe_zahl = 3 + 4j
```

Best Practices

- **Typkonvertierungen**: Python erlaubt die Konvertierung zwischen den numerischen Typen. Zum Beispiel kann eine Gleitkommazahl in eine Ganzzahl konvertiert werden (und umgekehrt), aber beachten Sie, dass dabei Informationen verloren gehen können.
- **Mathematische Operationen**: Python unterstützt alle grundlegenden mathematischen Operationen. Achten Sie auf die Typen der Operanden, da das Ergebnis einer Operation zwischen einem `int` und

einem **float** immer ein **float** sein wird.

Beispiele

```
# Addition
ergebnis = 5 + 3.2 # Ergebnis ist ein float: 8.2

# Typkonvertierung
int_zu_float = float(5) # wird 5.0
float_zu_int = int(3.9) # wird 3, Verlust der Dezimalstelle
```