

Userinput via `input()` in Python

Einführung

Die `input()`-Funktion in Python ermöglicht es, zur Laufzeit Eingaben vom Benutzer über die Kommandozeile zu erhalten. Die Funktion hält die Ausführung des Programms an und wartet auf eine Eingabe. Nachdem der Benutzer die Eingabe bestätigt (in der Regel durch Drücken der Enter-Taste), gibt `input()` den eingegebenen Text als String zurück.

Verwendung von `input()`

Grundlegende Verwendung

```
benutzer_eingabe = input("Bitte geben Sie etwas ein: ")
print(f"Sie haben eingegeben: {benutzer_eingabe}")
```

Eingabeaufforderung mit Typkonvertierung

Da `input()` immer einen String zurückgibt, kann es notwendig sein, den Rückgabewert in einen anderen Typ zu konvertieren, z.B. in eine Ganzzahl (`int`) oder eine Gleitkommazahl (`float`), wenn mathematische Operationen durchgeführt werden sollen.

```
alter = int(input("Bitte geben Sie Ihr Alter ein: "))
print(f"Nächstes Jahr sind Sie {alter + 1} Jahre alt.")
```

Best Practices

- **Fehlerbehandlung:** Verwenden Sie Fehlerbehandlung (try-except Blöcke), um die Anwendung vor Fehleingaben zu schützen. Dies ist besonders wichtig, wenn Sie den eingegebenen String in einen spezifischen Datentyp konvertieren.
- **Klare Anweisungen:** Geben Sie klare und präzise Anweisungen in der Eingabeaufforderung, damit der Benutzer weiß, was er eingeben soll.

Beispiele

Einfache Eingabe

```
name = input("Wie heißt du? ")
print(f"Willkommen, {name}!")
```

Fehlerbehandlung bei der Eingabe

```
try:  
    zahl = int(input("Geben Sie eine Zahl ein: "))  
    print(f"Das Doppelte Ihrer Zahl ist {zahl * 2}.")  
except ValueError:  
    print("Das war keine gültige Zahl. Bitte versuchen Sie es erneut.")
```