LU04.L01 - Einfache Lambda Ausdrücke

```
add = lambda a, b: a + b  # Funktion zum Addieren zweier Zahlen

subtract = lambda a, b: a - b  # Funktion zum Subtrahieren zweier Zahlen

multiply = lambda a, b: a * b  # Funktion zum Multiplizieren zweier Zahlen

divide = lambda a, b: a / b if b != 0 else 'Division durch Null ist nicht
erlaubt!'  # Funktion zum Teilen zweier Zahlen


if __name__ == '__main__':

    # Testen Sie Ihre Funktionen hier
    print (add(5, 10))
    print(subtract(10, 5))
    print(multiply(3, 4))
    print(divide(15, 3))
```

Erklärung des if in der Lambda-Funktion

Der Lambda-Ausdruck für die Division sieht wie folgt aus:

```
divide = lambda x, y: x / y if y != 0 else "Division durch
Null nicht möglich"
```

In diesem Lambda-Ausdruck wird der ternäre Bedingungsoperator von Python verwendet, der folgendermaßen strukturiert ist:

```
x if condition else y
```

Dieser Operator gibt x zurück, wenn die `condition` wahr ist, und y sonst.

In der Lambda-Funktion für die Division wird der Wert von `x / y` zurückgegeben, wenn y nicht Null ist. Ist y jedoch Null, wird der String „Division durch Null nicht möglich“ zurückgegeben.

Dieser Ausdruck ist eine kompakte Darstellung der folgenden if-else-Struktur:

```
if y != 0:
    return x / y
else:
    return "Division durch Null nicht möglich"
```

Der Einsatz von Einzeiler-Bedingungen kann den Code kompakter gestalten, sollte jedoch sparsam eingesetzt werden, um die Lesbarkeit nicht zu beeinträchtigen.

© Kevin Maurizi

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m323/learningunits/lu04/loesungen/lambda1

Last update: **2024/03/28 14:07**