## **LU04b - Die lambda-Funktion in Python**

Die lambda-Funktion, auch bekannt als anonyme Funktion, ist eine kleine, eingeschränkte Funktion, die durch das Schlüsselwort lambda definiert wird. Sie kann beliebige Anzahl von Argumenten haben, aber nur einen Ausdruck. Der Ausdruck wird ausgewertet und zurückgegeben.

### **Syntax**

```
lambda arguments: expression
```

- **arguments**: Die Argumente, die der Lambda-Funktion übergeben werden (kann auch leer sein).
- expression: Der Ausdruck, der ausgewertet und zurückgegeben wird.

## **Beispiele**

Hier sind einige Beispiele für die Verwendung der `lambda`-Funktion:

1. Ohne Argumente:

```
greet = lambda: "Hello, World!"
print(greet()) # Output: "Hello, World!"
```

2. Mit einem Argument:

```
square = lambda x: x**2
print(square(4)) # Output: 16
```

3. Mit mehreren Argumenten:

```
add = lambda x, y: x + y
print(add(2, 3)) # Output: 5
```

4. In Kombination mit anderen Funktionen, wie map:

```
numbers = [1, 2, 3, 4]
squares = map(lambda x: x**2, numbers)
print(list(squares)) # Output: [1, 4, 9, 16]
```

#### Last update: 2025/11/17 16:20

# Verwendungszwecke

Die lambda-Funktion wird oft verwendet, wenn eine kurze, temporäre Funktion benötigt wird, ohne sie formal mit def zu definieren. Dies ist besonders nützlich in Fällen, wo eine einfache, kurzlebige Funktion als Argument für Higher-Order Functions wie map, filter oder reduce benötigt wird.

M323-LU04, M323-CG3, M323-CF3, M323-CE3



From:

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

Permanent link:

https://wiki.bzz.ch/modul/m323/learningunits/lu04/lambda

Last update: 2025/11/17 16:20



https://wiki.bzz.ch/ Printed on 2025/11/26 13:37