Lambdas

Contents

Lambdas are a way of defining small anonymous functions in Python. The name stems from functional programming. Anonymous means that these functions do not have a name. These functions can be treated as variables in a sense.

```
# f will now refer to a function
f = lambda q: q * 10
print(f(10))
# applied multiple times
print(f(f(f(10))))
```

```
100
10000
```

Lambdas can have multiple arguments which also can be lambdas:

```
g = lambda a, d: d(a) + 10

g(10,f)
```

```
110
```

We can use lambdas in the `sort` method for arrays. This is especially useful if we want to sort by some different property of the elements. Imagine you have an array of tuples and you want to sort by their sum. If some elements are equal in this property, the order in which they appeared in the initial array will be preserved:

```
tuples = [(2,1), (1,2), (3,9), (2,5), (12,7)]
tuples.sort(key = lambda a: a[0]+a[1])
print(tuples)
```

```
[(2, 1), (1, 2), (2, 5), (3, 9), (12, 7)]
```

Lambdas are usually small functions. Longer lambdas can make code unreadable.

Exercises

- **Define** the following lambdas:

1. $f(x) = x + 7$
2. $f(x, y) = x * y + 8$
3. $f(x, g) = 4 + g(x)$

Answer

```
# 1)
f = lambda x : x + 7
# 2)
f = lambda x,y: x * y + 8
# 3)
f = lambda x, y: 4 + g(x)
```

- **Sort** a list of tuples by the second element in non-increasing order.

Answer

```
tuples = [(1,2),(3,4),(5,3), (0,4), (2,7)]
tuples.sort(key = lambda a : -a[1])
```