intro_to_lambdas

January 7, 2025

1 Introduction to Lambdas

What is a Lambda Function? In Python, a lambda function is a small anonymous function defined with the lambda keyword. Unlike a regular function defined using def, a lambda function is a single expression which returns a value without needing a return statement. Lambda functions are syntactically restricted to a single expression.

When to Use Lambda Functions Lambda functions are particularly useful for simple operations that can be summarised in a single expression. They are often used when operating with methods that expect a function as an argument, such as map(), filter(), and sorted().

Basic Syntax of Lambda Functions The basic syntax of a lambda function is: lambda arguments: expression

1.1 Using Lambda with map()

The map() function applies a given function to each item of an iterable (like a list) and returns a list of the results.

[1]:
```
# First declare our list of numbers
numbers = [1, 2, 3, 4, 5]
```

[2]:
```
# The following is lambda function takes x as an argument and returns x ** 2 (x␣
↪squared)
# lambda x: x ** 2

# Here we map the lambda function to all the values in the numbers list
squared = map(lambda x: x ** 2, numbers)

# We can display the result as a list
list(squared)
```

[2]:
```
[1, 4, 9, 16, 25]
```

1.2 Using Lambda with filter()

The filter() function creates a list of elements for which a function returns true.

[3]:
```
# This lambda function returns True if dividing x by 2 has a remainder of zero
even_numbers = filter(lambda x: x % 2 == 0, numbers)

list(even_numbers)
```

[3]:
```
[2, 4]
```

1.3 Using Lambda with sorted()

You can use lambda functions to specify the sorting criteria for sorted().

[4]:
```
# First we'll create a list of words to sort
words = ['banana', 'avocado', 'date', 'cherry']

# If we just use the sorted function, the words will be sorted in alphabetical␣
 ↪order
sorted(words)
```

[4]:
```
['avocado', 'banana', 'cherry', 'date']
```

[5]:
```
# But we can use a lambda to specify a custom sort funtion
# Here we sort by the length of the word
sorted(words, key=lambda word: len(word))
```

[5]:
```
['date', 'banana', 'cherry', 'avocado']
```

[6]:
```
# Here we sort using reverse=True to instead sort the list from the longest to␣
 ↪the shortest.
sorted(words, key=lambda word: len(word), reverse=True)
```

[6]:
```
['avocado', 'banana', 'cherry', 'date']
```