Lambda Function

A lambda function is a small anonymous function.

A lambda function can take any number of arguments, but can only have one expression.

Syntax

lambda arguments : expression

The expression is executed and the result is returned:

Example

A lambda function that adds 10 to the number passed in as an argument, and print the result:

x = lambda a : a + 10

print(x(5))

Lambda functions can take any number of arguments:

Example

A lambda function that multiplies argument a with argument b and print the result:

x = lambda a, b : a * b

print(x(5, 6))

Example

A lambda function that sums argument a, b, and c and print the result:

x = lambda a, b, c : a + b + c

print(x(5, 6, 2))

Why Use Lambda Functions?

The power of lambda is better shown when you use them as an anonymous function inside another function.

Say you have a function definition that takes one argument, and that argument will be multiplied with an unknown number:

```python
def myfunc(n):
  return lambda a : a * n
```

Use that function definition to make a function that always doubles the number you send in:

Example

```python
def myfunc(n):
  return lambda a : a * n

mydoubler = myfunc(2)
print(mydoubler(11))
```