Anonymous / Lambda Function

In Python, anonymous function is a function that is defined without a name.

While normal functions are defined using the def keyword, in Python anonymous functions are defined using the lambda keyword.

Lambda functions are used extensively along with built-in functions like filter(), map()

syntax:

```
lambda arguments: expression
```

Example:

In [1]:

```
double = lambda x: x*2


print(double(5))
```

```
10
```

In [2]:

```
def double(x):
    return x * 2

print(double(5))
```

```
10
```

In [7]:

```
#Example use with filter()
lst = [1, 2, 3, 4, 5]
even_lst = list(filter(lambda x: (x%2 == 0), lst))
print(even_lst)
```

```
[2, 4]
```

In [3]:

```
#Example use with map()
lst = [1, 2, 3, 4, 5]
new_lst = list(map(lambda x: (x ** 2), lst))
print(new_lst)
```

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

In [2]:

```
#Example use with reduce()
from functools import reduce

lst = [1, 2, 3, 4, 5]
product_lst = reduce(lambda x, y: x*y, lst)
print(product_lst)
```

```
120
```

In [ ]:

```
[10,12,14]

10*12 = 120
120
120 * 14
```