3.5.2. Anonymous

Fig. 3.5.2.1 Photo by Alejandro Piñero Amerio on Unsplash

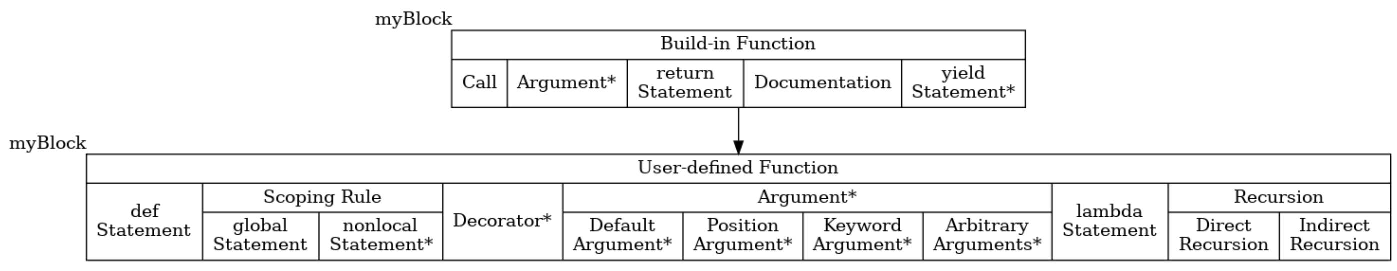
Note

Outline (v20220501)

Note

Roadmap

1. This topic: Function



2. Course: Python 1
3. Subject: Programming
4. Field
    a. Software Engineering (SE)
    b. Computer Science and Information Engineering (CSIE)
    c. Electrical/Electronics Engineering (EE)

3.5.2.1. Anonymous Function

1. Anonymous Function = Lambda Function
2. Syntax

```
lambda arguments: expression
```

3. Some key benefits of lambda functions:
    a. Concise, readable for small, simple functions.
    b. Avoid unnecessary [def] statements for small functions.
    c. Can be anonymous and passed directly to other functions.
    d. Leverage closure for data access from containing scope.
4. So in summary, lambda functions are a very useful feature in Python for creating small and in-line functions.

3.5.2.2. Examples

3.5.2.2.1. Ex1a: Doubled

1. Lambda functions can be assigned to variables.
2. This creates a function that takes x, multiplies it by 2, and returns the result.
3. Code

Listing 3.5.2.2.1.1 /src/Function/lambda/Ex1a.py

```python
'''
author: cph
since: 20230727
'''

if __name__ == '__main__':
    double = lambda x: x * 2
    print(double(5))
```

4. Output

```
10
```

3.5.2.2.2. Ex1b: Doubled: List

1. Lambdas are very useful for passing small functions as arguments.
2. Code

Listing 3.5.2.2.2.1 /src/Function/lambda/Ex1b.py

```python
'''
author: cph
since: 20230727
'''

if __name__ == '__main__':
    liNum = [2, 4, 6, 8]
    loDoubled = map(lambda x: x*2, liNum)
    print(loDoubled)
    print(list(loDoubled))
```

3. Output

```
<map object at 0x0000026706519A20>
[4, 8, 12, 16]
```

3.5.2.2.3. Ex1c: Doubled: List

1. Lambda functions can reference variables from the containing scope.
2. Code

Listing 3.5.2.2.3.1 /src/Function/lambda/Ex1c.py

```python
'''
author: cph
since: 20230727
'''

if __name__ == '__main__':
    x = 2
    inc1 = lambda x : x + 1      # 1 argument
    y = inc1(x)
    print(f'x={x}; y={y}; inc()={inc1(y)}')

    x = 5
    inc2 = lambda : x + 1       # 0 argument
    y = inc2()
    print(f'x={x}; y={y}; inc()={inc2()}')
```

3. Output

   a. We found lambda function can send 0+ argument(s).

```
x=2; y=3; inc()=4
x=5; y=6; inc()=6
```

3.5.2.2.4. Ex2: Sum

1. Code

Listing 3.5.2.2.4.1 /src/Function/p0813AnonymousFunction.py

```
# lambda function
iSum = lambda arg1, arg2: arg1 + arg2;

print("The sum of 10 and 20 is:", iSum(10, 20))
print("The sum of 20 and 50 is:", iSum(20, 50))

x = 10; y = 20
print(f"The sum of {x} and {y} is:", lambda x, y: x + y)
print(f"The sum of {x} and {y} is:", (lambda x, y: x + y)(x, y))
x = 20; y = 50
print(f"The sum of {x} and {y} is:", lambda x, y: x + y)
print(f"The sum of {x} and {y} is:", (lambda x, y: x + y)(x, y))
```

2. Output

```
The sum of 10 and 20 is: 30
The sum of 20 and 50 is: 70
The sum of 10 and 20 is: <function <lambda> at 0x0000018E460A99E0>
The sum of 10 and 20 is: 30
The sum of 20 and 50 is: <function <lambda> at 0x0000018E460A99E0>
The sum of 20 and 50 is: 70
```

3.5.2.2.5. Ex3: Judgment

1. Code

Listing 3.5.2.2.5.1 /src/Function/p0813AnonymousFunctionJudgment.py

```
# lambda function
bRet = lambda arg1, arg2: arg1 and arg2;

print("The boolean operation of true and true is:", bRet(True, True))
print("The boolean operation of true and false is:", bRet(True, False))
print("The boolean operation of false and false is:", bRet(False, True))
print("The boolean operation of false and false is:", bRet(False, False))
```

2. Output

```
The boolean operation of true and true is: True
The boolean operation of true and false is: False
The boolean operation of false and false is: False
The boolean operation of false and false is: False
```

Note

1. Start: 20170719
2. System Environment

Listing 3.5.2.2.5.2 requirements.txt

```
sphinx==7.1.2                             # Sphinx
graphviz>=0.20.1                          # Graphviz
sphinxbootstrap4theme>=0.6.0              # Theme: Bootstrap
sphinx-material>=0.0.35                   # Theme: Material
sphinxcontrib-plantuml>=0.25              # PlantUML
sphinxcontrib.bibtex>=2.5.0               # Bibliography
sphinx-autorun>=1.1.1                     # ExecCode: pycon
sphinx-execute-code-python3>=0.3          # ExecCode
btd.sphinx.inheritance-diagram>=2.3.1     # Diagram
sphinx-copybutton>=0.5.1                  # Copy button
sphinx_code_tabs>=0.5.3                   # Tabs
sphinx-immaterial>=0.11.3                 # Tabs

#------------------------------------------------------
#-- Library Upgrade Error by Library Itself
#   >> It needs to fix by library owner
#   >> After fixed, we need to try it later
#------------------------------------------------------
pydantic==1.10.10                         # 2.0: sphinx compiler error, 20230701

#------------------------------------------------------
#-- Minor Extension
#------------------------------------------------------
sphinxcontrib.httpdomain>=1.8.1           # HTTP API

#sphinxcontrib-blockdiag>=3.0.0           # Diagram: block
#sphinxcontrib-actdiag>=3.0.0             # Diagram: activity
#sphinxcontrib-nwdiag>=2.0.0              # Diagram: network
#sphinxcontrib-seqdiag>=3.0.0             # Diagram: sequence

#------------------------------------------------------
#-- Still Wait For Upgrading Version
#------------------------------------------------------

#------------------------------------------------------
#-- Still Under Testing
#------------------------------------------------------
#numpy>=1.24.2                            # Figure: numpy

#------------------------------------------------------
#-- NOT Workable
#------------------------------------------------------
#sphinxcontrib.jsdemo==0.1.4    # ExecCode: Need replace add_js_file()
#jupyter-sphinx==0.4.0          # ExecCode: Need gcc compiler
#sphinxcontrib.slide==1.0.0     # Slide: Slideshare
#hieroglyph==2.1.0              # Slide: make slides
#matplotlib>=3.7.1              # Plot: Need Python >= v3.8
#manim==0.17.2                  # Diagram: scipy, numpy need gcc
#sphinx_diagrams==0.4.0         # Diagram: Need GKE access
#sphinx-tabs>=3.4.1                       # Tabs: Conflict w/ sphinx-material
```