Statistical Techniques for Monitoring Industrial Processes

Lecture : Python Language Basics

Module : Python Installation and Basics

Course TOC

Basic Data Types

● Integers

```
i = 2 # type(i) = int
```

● Floating-point numbers

```
f = 1.2 # type(f) = float
```

● Strings

```
s = 'two' # type(s) = str
```

● Boolean

```
b = True # type(b) = bool
```

Data Sequence: Lists

● A sequence of data (of same or different data types) can be put together in a list

```
listVar = ['air’, 3, 1, 5]
```

Ways of creating a list

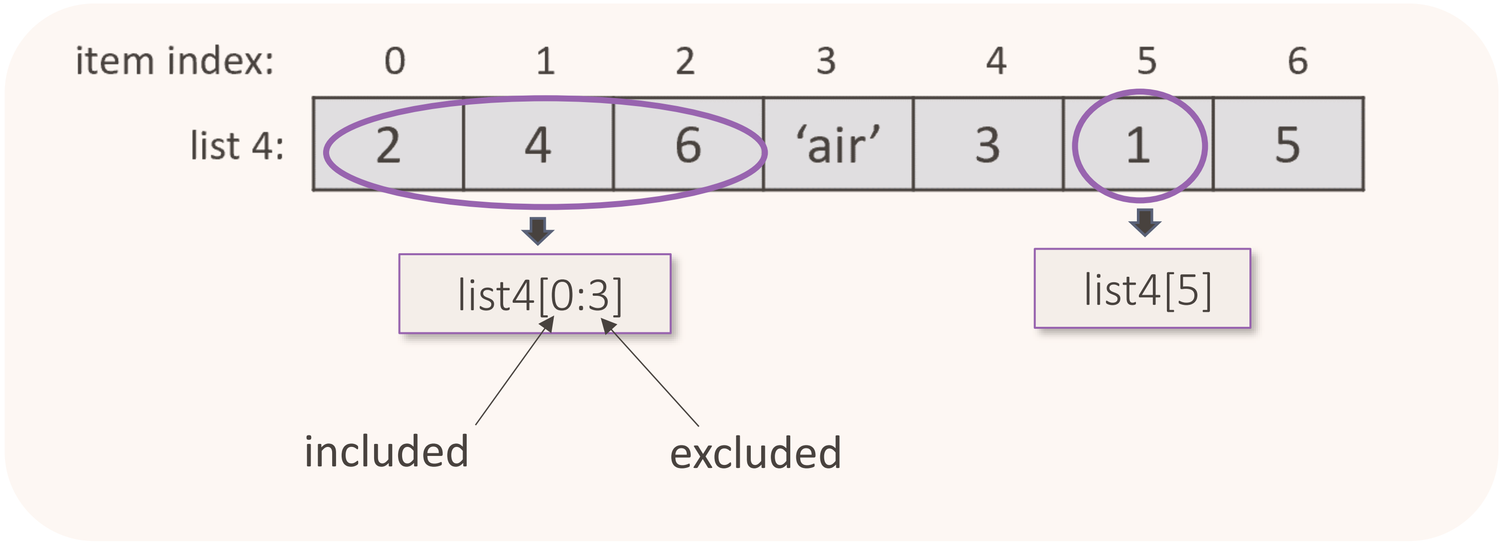
```
list1 = [2,4,6]

list2 = ['air',3,1,5]

list3 = list(range(4)) # equals [0,1,2,3]

        ⋮
```

List Comprehension

```
# generate powers of individual items in list3

newList1 = [item**2 for item in list3]
# equals [0,1,4,9]
```

Indexing and Slicing Sequences

- In Python, indexing starts from 0 ⇒ the first element in a sequence has an index of 0



Slicing Syntax: givenList[start, stop, step]

Execution Control

● Execute code conditionally or multiple times

Conditional execution

```
# selectively execute code based on condition
if list1[0] > 0:
    list1[0] = 'positive'
else:
    list1[0] = 'negative'
# list1 becomes ['positive', 4, 6]
```

Loop execution

```
# compute sum of squares of numbers in list3
sum_of_squares = 0
for i in range(len(list3)):
    sum_of_squares += list3[i]**2

print(sum_of_squares) # displays 78
```

Custom Functions

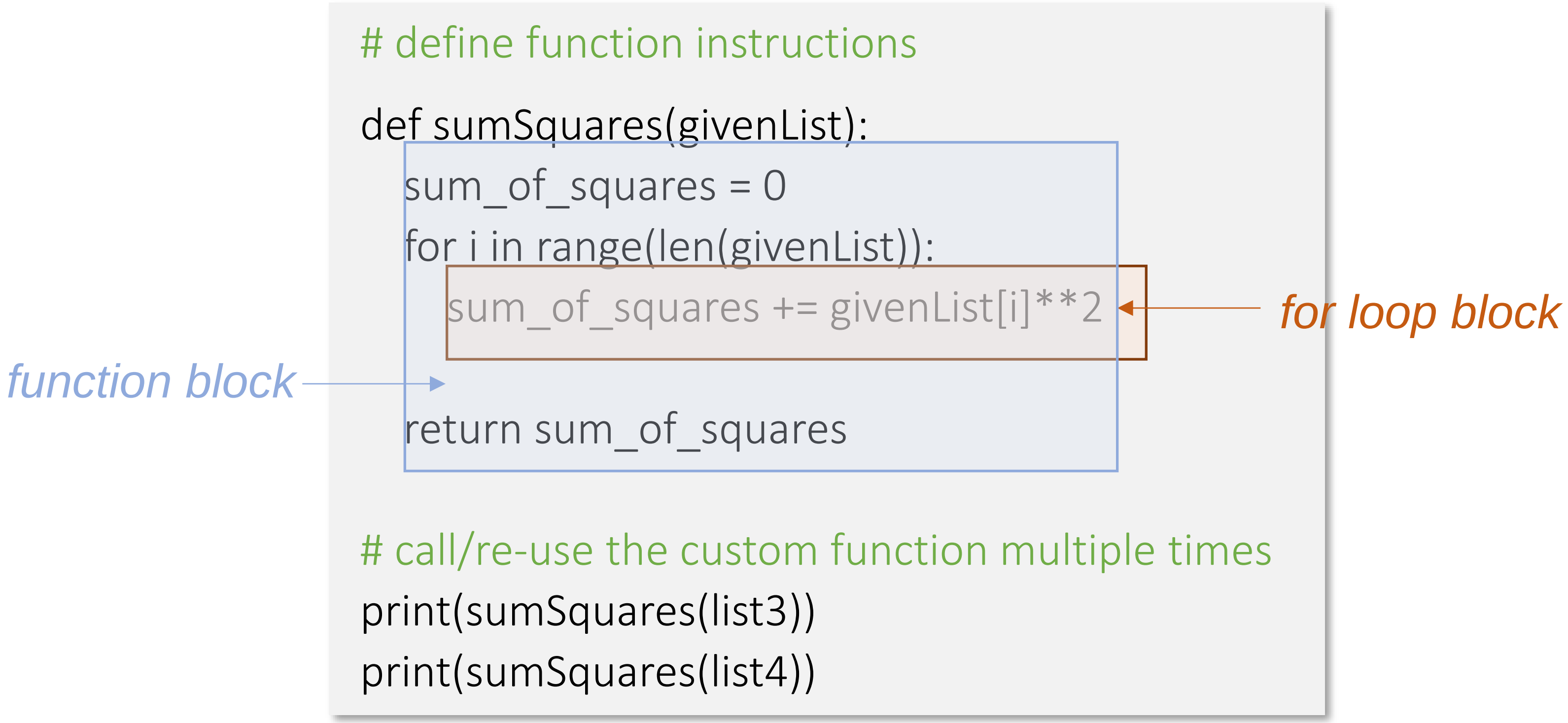
● Define your own set of instructions once and then reuse multiple times within a script and across scripts

```
# define function instructions

def sumSquares(givenList):
    sum_of_squares = 0
    for i in range(len(givenList)):
        sum_of_squares += givenList[i]**2

    return sum_of_squares

# call/re-use the custom function multiple times
print(sumSquares(list3))
print(sumSquares(list4))
```

Indentation in Python

● In Python, indentation is used to define code blocks

- *amount of whitespace (spaces or tabs) at the beginning of a line determines which block of code it belongs to*
- *all statements within the same block must have the same level of indentation*



```
# define function instructions

def sumSquares(givenList):
    sum_of_squares = 0
    for i in range(len(givenList)):
        sum_of_squares += givenList[i]**2

    return sum_of_squares

# call/re-use the custom function multiple times
print(sumSquares(list3))
print(sumSquares(list4))
```

Statistical Techniques for Monitoring Industrial Processes

Next Lecture : Scientific Computing Package: NumPy

Module : Python Installation and Basics