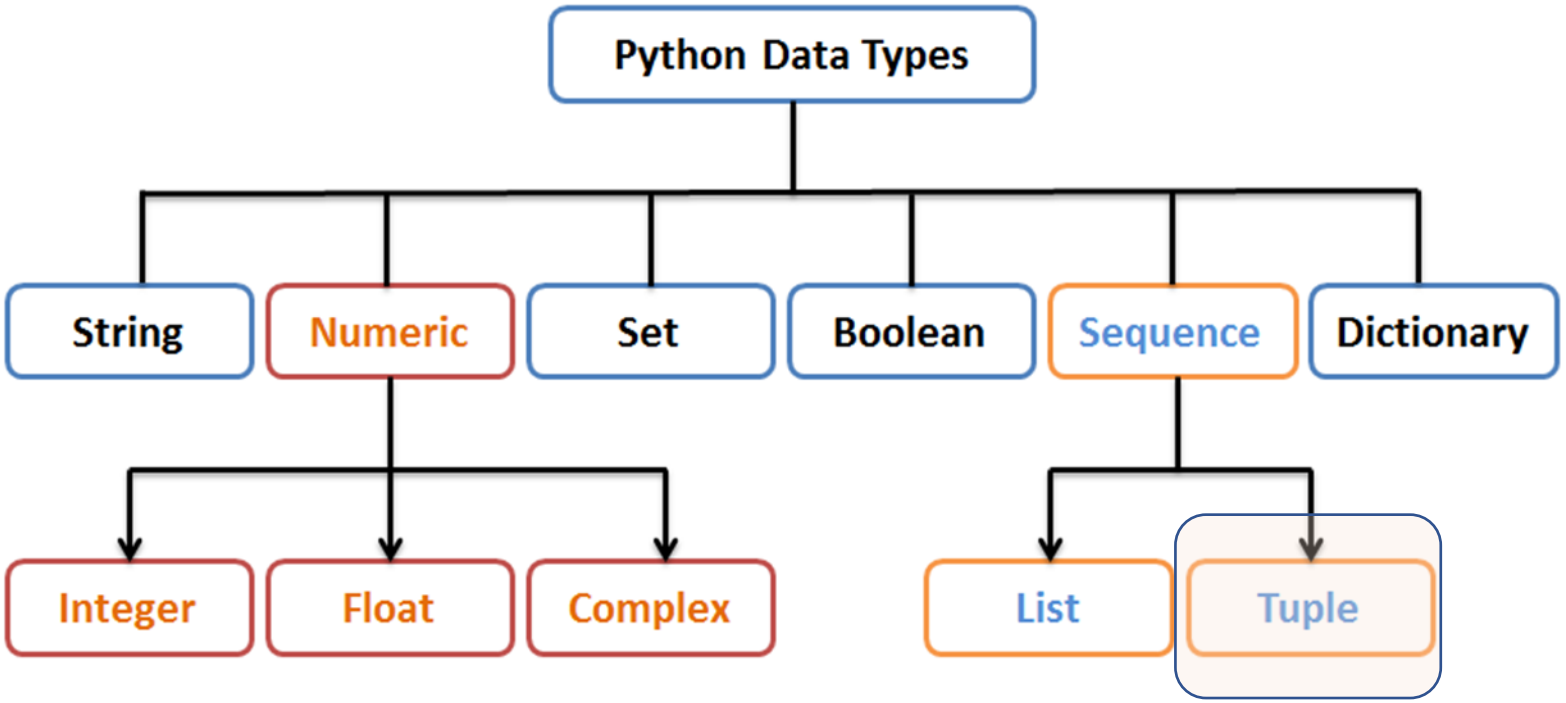
# Python Tuples

Anoop S Babu
Faculty Associate
Dept. of Computer Science & Engineering
bsanoop@am.amrita.edu

# Tuple


Python Data Types
String
Numeric
Set
Boolean
Sequence
Dictionary
Integer
Float
Complex
List
Tuple

# Tuple

- A sequence of comma separated values
- Enclosed in () (parentheses)
- Can hold multiple data types
- Allow duplicate values
- Ordered
- Use index to access contents
- Immutable

# Creating a tuple

```
# using ()
>>> studId = (101, 102,103,104)
>>> print(studId)
(101, 102, 103, 104)

# packing a tuple
>>> tp = 1, "Python", 3.8
>>> print(tp)
(1, 'Python', 3.8)
```

# Creating a tuple with single element

```
>>> studId = (101)
>>> print(type(studId))
<class 'int'>


>>> studId = (101,)
>>> print(type(studId))
<class 'tuple'>
```

Add a comma ‘,’ after the element to create a singleton tuple.

# Accessing tuple elements

Using index in square brackets

```
>>> tp = 1, "Python", 3.8
>>> print(tp[1])
Python
```

# Slicing a tuple

Using slice operator [:]

```
>>> tp = 1, "Python", 3.8
>>> print(tp[:2])
(1, 'Python’)
>>> print(tp[:-2])
(1,)
```

Slicing is same as in strings and list

# Deleting a tuple

Using del() function

```
>>> tp = 1, "Python", 3.8
>>> print(tp)
(1, 'Python', 3.8)
>>> del(tp)
>>> print(tp)
…
NameError: name 'tp' is not defined
```

# Immutable property

```
# reassign a tuple element
>>> tp = 1, "Python", 3.8
>>> tp[0] = 3
…
TypeError: 'tuple' object does not support item assignment
```

Can’t change tuple elements.

Also can’t delete individual tuple elements.

# Traversing a tuple

```
>>> numbers = (1,3,5,3,5)
>>> for n in numbers:
        print(n)
```

```
1
3
5
3
5
```

# Nested tuples

```
>>> points = ((2,5), (3,6), (5,9))
#accessing single item
>>> print(points[1])
(3, 6)

# accessing single value in the item
>>> print(points[2][0])
5
```

# Nested tuples

```
# any object can be nested in a tuple.
>>> languages = ("Python", ["C", "C++"], "Java")
>>> print(languages)
('Python', ['C', 'C++'], 'Java’)
```