



Welcome

Advanced Python

Python certification
course



Course Contents

1. Introduction
2. The Programming Cycle for Python
3. Getting Started
4. Variables and simple data types
5. Elements of python
6. Type conversion
7. Expressions
8. Assignment statement
9. Arithmetic operators
10. Operator precedence
11. Boolean expression
- 12. Introducing lists**
- 13. Working with lists**
14. For loop
15. Nested loops
- 16. Tuples**
17. Unpacking sequences
- 18. Lists**
- 19. Mutable sequences**
20. List comprehension
- 21. Sets**
22. If statements
23. Conditions
24. Expression evaluation
25. Float representation
- 26. Dictionaries**
27. User input and loops
28. Break and continue
29. Function
30. Parts of a function
31. Execution of a function
32. Keyword and default arguments
33. Scope rules
- 34. Strings**
- 35. Indexing and slicing of strings**
- 36. More slicing**
37. Higher order functions
38. Sieve of Eratosthenes
39. Abstract data types
40. Classes
41. Modules
42. Importing modules
43. Classes
44. Class example
45. Special methods
46. Inheritance
47. Inheritance and OOPS
48. Files and exceptions
49. File I/O
50. Exceptions
51. Testing your code
52. Assertions
53. Iterators
54. Recursion
55. Simple search
56. Estimating search time
57. Binary search
58. Estimating Binary search time
59. Recursive Fibonacci
60. Tower of Hanoi
61. Sorting
62. Selection sort
63. Merge list
64. Merge sort
65. Higher order sort



Sequence types in Python

What is sequence type?

▶ A sequence type is type that holds a sequence of elements.

Sequence types in Python

- List - Sequence of elements enclosed in square brackets : []
- Tuple - Sequence of elements enclosed in parenthesis : ()
- Set - Sequence of elements enclosed in curly braces : {}
- Dictionary - Sequence of elements enclosed in curly braces : {} : each element is a pair
- String - Sequence of characters, digits, or any symbols enclosed in quotes : "" or ''

Elements of a sequence are separated by using comma and elements may of different data type



Sequence types in Python

What is sequence type?

▶ A sequence type is type that holds a sequence of elements.

Sequence types in Python

Mutab

List
Dictionary
Set

Immutable

Tuple
String



Introduction to list in Python

What is
list?

A list is a collection of indexed elements of different data types.

Properties of list in
Python

List in python has implemented using a built-in
class list

List is represented using square
brackets []

Every element in a list separated with comma (,) symbol

All list elements are indexed and the index starts with '0'

The list elements are also indexed with negative numbers from the last element to the first element in the list, and it begins with -1 for each element decreased by 1.

List and its
Elements

Mutable



Introduction to list in Python

What is
list?

A list is a collection of indexed elements of different data types.

How to create a
list?

```
listName = [value1, value2,  
            value3, ...]
```



Introduction to list in Python

What is
list?

A list is a collection of indexed elements of different data types.

How to access individual elements of
a list?

```
listName[ind  
ex]
```



Introduction to list in Python

What is
list?

A list is a collection of indexed elements of different data types.

How to access sub list from
a list?

Slicing

```
listName[startIndex :  
            endIndex]  
listName[startIndex : endIndex :  
            indexAddition]
```




Introduction to list in Python

What is
list?

A list is a collection of indexed elements of different data types.

Built-in methods
of list

append(value)
insert(index,
value)
extend(list)

remove(value)
pop()
pop(index)
clear()
del

len(list)
count(value)
sort()
reverse()
max(list)
min(list)
index(value)

| Tuple in Python

What is
tuple?

A tuple is a collection of indexed elements of different data types.

Properties of tuple in Python

Tuple in python has implemented using a built-in class tuple

Tuple is represented using
Parenthesis ()

Every element in a tuple separated with comma (,) symbol

All tuple elements are **indexed** and the index starts with '0'

The Tuple elements are also indexed with negative numbers from the last element to the first element in the tuple, and it begins with -1 for each element decreased by 1.

Tuple elements
are
Immutable



Tuple in Python

What is
tuple?

A tuple is a collection of indexed elements of different data types.

How to create a
tuple?

```
tupleName = (value1, value2,  
             value3, ...)
```

| Tuple in Python

What is
tuple?

A tuple is a collection of indexed elements of different data types.

How to access individual elements of a
tuple?

```
tupleName[index]
```

| Tuple in Python

What is
tuple?

A tuple is a collection of indexed elements of different data types.

How to access sub tuple from a
tuple?

Slicing

```
tupleName[startIndex :  
            endIndex]  
tupleName[startIndex : endIndex :  
            indexAddition]
```

| Tuple in Python

What is
tuple?

A tuple is a collection of indexed elements of different data types.

Built-in methods of
tuple

```
len(tuple)  
count(value)  
sorted(tuple,  
reverse=)  
max(list)  
min(list)  
index(value  
)
```

| Set in Python

What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

Properties of set in
Python

Set in python has implemented using a built-in class set

Set is represented using curly braces { }

Every element in a set separated with comma (,) symbol

Set elements
are

Immutable



Set in Python

What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

How to create a
set?

```
setName = {value1, value2,  
           value3, ...}
```


What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

How to access individual elements of
a set?

NOT
ALLOWED

Set in Python

What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

How to access sub set from a
set?

Slicing
ng
**NOT
ALLOWED**

Set in Python

What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

Built-in methods of
set

```
add(value)  
append(list_of_values)
```

```
discard(value)  
remove(value)  
pop()  
clear()  
del
```

```
len(set)  
sorted(set, reverse=)  
max(set)  
min(set)
```

| Set in Python

What is
set?

A set is a collection of un-ordered and un-indexed elements of different data types.

Special operations
on set

Union
union(set)
|

Intersection
intersection(set)
&

Difference
difference(set)
-

Symmetric
Difference
symmetric_difference(set)
^



Dictionary in Python

What is
dictionary?

A dictionary is a collection of un-ordered elements where each element is a pair of key and value.

Element in dict

key:
value

Dictionary and it's elements
are

Mutable



| Dictionary in Python

Properties of dictionary in Python

Dictionary in python has implemented using a built-in class dict

Dictionary is represented using curly braces { }

Every element in a dictionary separated with comma (,) symbol

Every element in a dictionary is a pair of key and value

The key and the corresponding value are separated with : symbol →

key: value

Dictionary doesn't allow duplicate keys, but value does not have any restriction

Key must be either a string or a number or any immutable object and value can be any object



Dictionary in Python

How to create a dictionary?

```
dictName = {key1:value1, key2: value2,  
            key3:value3, ...}
```

How to access individual elements of a dictionary?

```
dictName[k  
ey]
```

How to access sub set from a
dictionary?

Slicing
ng
**NOT
ALLOWED**



Dictionary in Python

Built-in methods of dictionary

get(key
)

items(
)

keys()

values(
)

Pop(key
)

popitem(
)

clear(
)

de
l

len(dictio
nary)

update({key:
value})



String in Python

What is
string?

▶ A string is a sequence of characters enclosed in quotes

"Hell
o"

"""Python"""

'Rank
1'

""It's
simple""



String in Python

Properties of string in Python

String in python has implemented using a built-in class str

String is represented using quotes

“ ”

String may contain any character like alphabets, digits, special symbols

String elements are immutable

All string elements are **indexed** and the index starts with '0'

The string elements are also indexed with negative numbers from the last element to

the first element in the string, and it begins with -1 for each element decreased by 1.

How to create a
string?

```
strName =  
    'value'
```

How to access individual elements of a
string?

```
strName[ind  
ex]
```



String in Python

How to access sub string from a string?

Slicing

strName[start :
end]

strName[start : end :
modifier]



Understanding Slicing in Python

str = "PYTHON IS EASY"

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
str	P	Y	T	H	O	N		I	S		E	A	S	Y
r	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

str[2 :
8]
↓
THO



Understanding Slicing in Python

str = "PYTHON IS EASY"

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
str	P	Y	T	H	O	N		I	S		E	A	S	Y
r	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

● ●

str[-11 :
-5]

↓

HON



Understanding Slicing in Python

str = "PYTHON IS EASY"



str[7 :
-3]

↓

IS



Understanding Slicing in Python

str = "PYTHON IS
EASY"

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
str	P	Y	T	H	O	N		I	S		E	A	S	Y
r	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

str[2 : 9 :
1]
↓
THON



Understanding Slicing in Python

str = "PYTHON IS
EASY"

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
str	P	Y	T	H	O	N		I	S		E	A	S	Y
r	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

str[8 : 1 :
-1]

SI



Understanding Slicing in Python

str = "PYTHON IS
EASY"



str[2 : 9 :
2]

↓
TO



Understanding Slicing in Python

str = "PYTHON IS
EASY"

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
str	P	Y	T	H	O	N		I	S		E	A	S	Y
r	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

str[8 : 1 :
-2]

↓

S



String in Python

Built-in methods of String

capitalize
()

center(width,
fill_char)

startswith(str, beginindex,
endindex)

endswith(str, beginindex,
endindex)

find(str, beginindex,
endindex)

isdigit(
)

isnumeric(
)

isspace(
)

isalpha(
)

isdecimal(
)

isalnum(
)

lower(
)

upper(
)

split('separator',
maxsplit)

max(str
)

min(str
)