

Full stack web development using python

List

Part-3



Saurabh Shukla (MySirG)

Agenda

- ① list of lists
- ② list object methods
- ③ list comprehension
- ④ user input
- ⑤ Mutability and hashability

list of lists

$l1 = [[1, 3, 5], [2, 1, 8], [5, 4, 4]]$

$l1[0]$ [1, 3, 5]

$l1[1]$ [2, 1, 8]

$l1[2]$ [5, 4, 4]

$l1[0][0]$ 1

$l1[0][1]$ 3

$l1[0][2]$ 5

list object methods

append()

insert()

remove()

→ Remove first occurrence of specified element. It gives error if the element is not present in the list

pop()

→ Remove and return last element
optionally you can pass index number
to pop specific element.

clear() → Remove all the elements

reverse() → Reverse the list

sort() → Sort list elements

index() → Returns index of first occurrence
of specified element

count() → Count number of occurrence of
specified element

list Comprehension

[expression for variable in iterable]

[$x^2 + 1$ for x in range(1, 6)]

[2, 5, 10, 17, 26]

$$x = 1, 2, 3, 4, 5$$

$$x^2 + 1 = 2 \ 5 \ 10 \ 17 \ 26$$

User input

$l = \text{list}(\text{input}())$

Mutability and hashability

- Mutable objects are changeable
- immutable objects are not changeable
- All immutable objects are hashable, but not all hashable objects are immutable.
- Hashable is a feature of Python objects that tells if the object has a hash value or not. If it has a hash value that does not change during its entire lifetime.

	Hashable	Mutable
int	Yes	NO
float	Yes	NO
complex	Yes	NO
bool	Yes	NO
str	Yes	NO
range	Yes	NO
list	No	Yes
tuple	Yes	NO
set	No	Yes
dict	No	Yes