Python List & Tuple Cheat Sheet

☑ Built-in Functions (work on both lists and tuples)

- len()

  Description: Returns the number of elements

  Example: len([1, 2, 3]) → 3

- max()

  Description: Returns the largest item

  Example: max((1, 5, 2)) → 5

- min()

  Description: Returns the smallest item

  Example: min([4, 9, 2]) → 2

- sum()

  Description: Returns the total of elements (numeric)

  Example: sum((1, 2, 3)) → 6

- sorted()

  Description: Returns a sorted copy

  Example: sorted([3, 1, 2]) → [1, 2, 3]

- reversed()

  Description: Returns an iterator that accesses values in reverse

  Example: list(reversed([1, 2])) → [2, 1]

- enumerate()

  Description: Returns (index, value) tuples

  Example: for i, v in enumerate(lst):

- zip()

  Description: Aggregates elements from multiple iterables

Example: zip([1,2], [3,4]) → [(1,3), (2,4)]

- any()

Description: True if any element is true

Example: any([False, True]) → True

- all()

Description: True if all elements are true

Example: all([1, 2, 0]) → False

List Methods (Mutable)

- append(x)

Description: Adds an item to the end

Example: lst.append(4)

- extend(iter)

Description: Adds all items from another iterable

Example: lst.extend([4, 5])

- insert(i, x)

Description: Inserts item at a given position

Example: lst.insert(1, 'a')

- remove(x)

Description: Removes first occurrence of x

Example: lst.remove(2)

- pop([i])

Description: Removes and returns item at index (end by default)

Example: lst.pop()

- clear()

Description: Removes all items

Example: lst.clear()

- index(x)

Description: Returns first index of x

Example: lst.index(5)

- count(x)

Description: Returns number of occurrences of x

Example: lst.count(2)

- sort()

Description: Sorts the list in place

Example: lst.sort()

- reverse()

Description: Reverses the list in place

Example: lst.reverse()

- copy()

Description: Returns a shallow copy of the list

Example: new_lst = lst.copy()

Tuple Characteristics

- Immutable: You can't modify, add, or remove elements once created.

- Supports all built-in functions listed above.

- No methods like .append() or .remove().

Converting Between Types

```
tuple([1, 2, 3])    # list → tuple

list((1, 2, 3))     # tuple → list
```