Python Data Structures Cheat Sheet

**Primitive Data Types**

String: Ram, "Shyam" -sequences of characters

Boolean: True, False -logical values

Integer: 42, -7 -whole numbers

Float: 3.14, 2.7 -decimals

Built-In Data Structures

**List**

Ordered, mutable, allows duplicates

Syntax: [1, 2, 3]

**Operations:**

.append(item) -Add to end

.insert(i, item) -Insert at position

.remove(item) -Remove first occurrence

.pop([i]) -Remove & return item at index (last by default)

.sort() -In-place sort

.reverse() -Reverse items

Slicing: list[start:stop:step]

**Tuple**

Ordered, immutable, allows duplicates

Syntax: (1, 2, 3)

Operations:

tuple.count(item) -Count occurrences

tuple.index(item) -Find index

**Set**

Unordered, mutable, no duplicates

Syntax: 1, 2, 3} or set ([1, 2, 3])

**Operations:**

.add(item) Add item

. update([items]) Add multiple

. remove(item) /. discard(item) -Remove item

**Dictionary**

Key-value pairs, ordered (Python 3.7+), mutable

Syntax: {'name': 'Alice', 'age': 30}

Operations:

.get(key, default) Safe access

.set default(key, default) - Set if key missing

.keys() - All keys

. values() -All values

. items() -Key-value pairs

. update({key: value}) -Batch update

.pop(key) -Remove and return value

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type | Ordered | Changeable | Duplicates | Syntax | Example |
| List | Yes | Yes | Yes | [ ] | [1, 2, 'a'] |
| Tuple | Yes | No | Yes | ( ) | (1, 'b', 3) |
| Set | No | Yes | No | { } or set() | {1, 2, 3} |
| Dictionary | Yes | Yes | No | {key: value} | {'a': 1} |