Alright — here's your **color-coded one-page cheat sheet** for `list`, `tuple`, `set`, and `dict` methods so you can remember them at a glance.

📝 Python Collection Methods Cheat Sheet

📍 LIST *(ordered, mutable)*

💡 *Think: add/remove anywhere, sort, reverse*

```
append(x)       # add to end
insert(i, x)    # add at position
extend(iter)    # add multiple
remove(x)       # remove first match
pop(i)          # remove & return (default last)
sort()          # sort in place
reverse()       # reverse in place
index(x)        # find position
count(x)        # count occurrences
```

📍 TUPLE *(ordered, immutable)*

💡 *Think: can't change, only check*

```
count(x)        # count occurrences
index(x)        # find position
```

(That's it — all tuple can do!)

📍 SET *(unordered, unique, mutable)*

💡 *Think: math operations + add/remove*

```
add(x)
remove(x)       # error if missing
discard(x)      # no error if missing
pop()           # remove random element
clear()         # empty the set
union(s)        # combine
intersection(s) # common elements
```

```
difference(s)   # remove s from current
symmetric_difference(s) # elements not in both
```

📍 DICT *(key-value pairs, mutable, ordered [3.7])*

💡 *Think: keys, values, pairs*

```
get(key, default)
keys()          # all keys
values()        # all values
items()         # all (key, value) pairs
update({})      # add/overwrite
pop(key)        # remove & return value
popitem()       # remove & return last pair
clear()         # empty dict
```

🕐 Memory trick:

- **List** = L for "Long" → many methods.
- **Tuple** = T for "Tiny" → only 2 methods.
- **Set** = S for "Subtract" → has math-like methods (union, intersection).
- **Dict** = D for "Double" → always (key, value) pairs.