

	List	Tuple	Dictionary	Set	String
① Symbol	[]	()	{ }	{ }	" "
② add	<code>l.append(el)</code> <code>l.insert(i, el)</code> <code>l[i] = el</code> (NO) <code>l.extend(iter)</code> <code>l += iter</code>	⊖	<code>d[key] += val</code> <code>d.update(d2)</code>	<code>s.add(el)</code> <code>s.update(iter)</code> <code>s += {1, 2, 3}</code> (NO)	⊖
③ Remove	<code>l.remove(el)</code> <code>l.pop()</code> <code>l.pop(i)</code> <code>del l[i]</code> <code>del l</code>	<code>del t</code> <code>(NO) del t[i]</code>	<code>del d[key]</code> <code>d.pop(key)</code> (returns val) <code>d.popitem()</code> <code>del d</code>	<code>s.discard(el)</code> (no error) <code>s.remove(el)</code> (error) <code>s.difference_update(iter)</code> <code>s -= {1, 2, 3}</code> set only (but <u>not</u> <code>s += {1, 2, 3}</code>)	⊖
④ modify	<code>l[i] = el</code> <code>l *= int</code>	⊖	<code>d[key] = val</code> <code>d.update(d2)</code>	⊖	⊖
⑤ retrieve	<code>l[i]</code> <code>l[i:]</code>	<code>t[i]</code> <code>t[i:]</code>	<code>d[key]</code> <code>d.get(key)</code> <code>d.get(key, 'shit')</code>	⊖	<code>s[index]</code> <code>s.find(el)</code> // returns <u>index</u> // -1 if no el
⑦ clear	<code>l.clear()</code>	! (NO)	<code>d.clear()</code>	<code>s.clear()</code>	(NO)
⑧ sort	<code>l.sort()</code> <code>sorted(l)</code>	<code>NO.sort()</code> <code>sorted(t)</code> ✓ returns <u>list</u>	<code>NO.sort()</code> <code>sorted(d)</code> returns list of sorted keys	<code>NO.sort()</code> <code>sorted(s)</code> returns list	<code>NO.sort()</code> <code>sorted('tomas')</code>
⑨					

⑨

mutable

(+)

(-)

(+)

(+)

(-)

⑩

index

`l.index(el)` `t.index(el)`

No

No

`str[i]`
`str[i:]`

⑪

count

`l.count(el)` `t.count(el)`

No

No

No