In [3]:
```python
b=True
print(b,type(b),id(b))
```

True <class 'bool'> 140730229128064

In [25]:
```python
True+False
```

Out[25]: 1

In [9]:
```python
c1=10+20j
print(c1,type(c1),id(c1))
```

(10+20j) <class 'complex'> 2017149585200

In [13]:
```python
c1=10+20j
c2=20+30j
c1.real
```

Out[13]: 10.0

In [15]:
```python
c1.imag
```

Out[15]: 20.0

In [21]:
```python
c1=10+20j
c2=20+30j
print(c1)
print(c2)
```

(10+20j)
(20+30j)

In [23]:
```python
c1+c2
```

Out[23]: (30+50j)

In [10]:
```python
c1=10+20j
c2=20+30j
c1-c2
```

Out[10]: (-10-10j)

In [ ]:
```python
c1=10+20j
```

In [ ]:
```python
c2=20+30j
```

In [18]:
```python
c1 * c2
```

Out[18]: (-500+1200j)

In [45]:
```python
l=[]
l
```

Out[45]: []

In [47]:
```python
l.append(10)
l.append(20)
l.append(30)
l
```

Out[47]: [10, 20, 30]

In [49]:
```python
print(l,type(l),id(l))
```

[10, 20, 30] <class 'list'> 1572898954944

In [51]:
```python
import keyword
keyword.kwlist
```

Out[51]: ['False',
         'None',
         'True',
         'and',
         'as',
         'assert',
         'async',
         'await',
         'break',
         'class',
         'continue',
         'def',
         'del',
         'elif',
         'else',
         'except',
         'finally',
         'for',
         'from',
         'global',
         'if',
         'import',
         'in',
         'is',
         'lambda',
         'nonlocal',
         'not',
         'or',
         'pass',
         'raise',
         'return',
         'try',
         'while',
         'with',
         'yield']

In [53]:
```python
len(keyword.kwlist)
```

Out[53]: 35

In [55]:
```python
l
```

Out[55]: `[10, 20, 30]`

In [61]:
```python
l.append(10)
l.append(20)
l.append(30)
l
```

Out[61]: `[10, 20, 30, 10, 20, 30]`

Slicing in List

In [63]:
```python
l[0]
```

Out[63]: `10`

In [65]:
```python
l[1]
```

Out[65]: `20`

In [67]:
```python
l[-3]
```

Out[67]: `10`

In [69]:
```python
l1 = l.copy()
l1
```

Out[69]: `[10, 20, 30, 10, 20, 30]`

In [71]:
```python
l==l1
```

Out[71]: `True`

In [73]:
```python
print(len(l))
print(len(l1))
```

```
6
6
```

In [77]:
```python
l1.append(2.3)
l1.append(True)
l1.append(1+3j)
l1
```

Out[77]: `[10, 20, 30, 10, 20, 30, 2.3, True, (1+3j)]`

In [79]:
```python
l1.append(60)
l1
```

Out[79]: `[10, 20, 30, 10, 20, 30, 2.3, True, (1+3j), 60]`

In [81]: `l1.count(10)`

Out[81]: 2

In [83]: `l1.count(100)`

Out[83]: 0

In [85]: `l1`

Out[85]: `[10, 20, 30, 10, 20, 30, 2.3, True, (1+3j), 60]`

In [121… `l1.remove(True)`

In [123… `l1`

Out[123… `[10, 20, 30, 10, 20, 30, 2.3, (1+3j), 60]`

In [125… `l1.clear()`

In [127… `l1`

Out[127… `[]`

In [129… `del l1`

In [131… `l1`

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[131], line 1
----> 1 l1

NameError: name 'l1' is not defined
```

In [ ]: