In [1]:
```
import sys
import keyword
import operator
from datetime import datetime
import os
```

In [2]:
```
print(keyword.kwlist)
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'brea
k', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finall
y', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonloc
al', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yiel
d']
```

In [3]:
```
len(keyword.kwlist)
```

Out[3]:
```
35
```

In [4]:
```
1var = 10
```

```
  Cell In[4], line 1
    1var = 10
    ^
SyntaxError: invalid decimal literal
```

In [5]:
```
val2@ = 35
```

```
  Cell In[5], line 1
    val2@ = 35
         ^
SyntaxError: invalid syntax
```

In [6]:
```
import = 125
```

```
  Cell In[6], line 1
    import = 125
           ^
SyntaxError: invalid syntax
```

In [7]:
```
"""
Correct way of defining an identifier
(Identifiers can be a combination of letters in lowercase (a to z) or upper
"""
val2 = 10
```

In [8]:
```
val_ = 99
```

In [9]:
```
# Single line comment
val1 = 10
```

In [10]:
```
# Multiple
# line
# comment
val1 = 10
```

In [11]:
```
'''
Multiple
line
comment
'''
val1 = 10
```

In [12]:
```
p = 20
q = 20
r = q
p , type(p), hex(id(p))
```

Out[12]:
```
(20, int, '0x7ff9e00a9588')
```

In [13]:
```
q , type(q), hex(id(q))
```

Out[13]:
```
(20, int, '0x7ff9e00a9588')
```

In [14]:
```
p = 20
p = p + 10
p
```

Out[14]:
```
30
```

In [15]:
```
intvar = 10
floatvar = 2.57
strvar = "Python Language"
print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.57
Python Language
```

In [16]:
```
intvar , floatvar , strvar = 10,2.57,"Python Language" # Using commas to se
print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.57
Python Language
```

In [17]:
```
p1 = p2 = p3 = p4 = 44 # All variables pointing to same value
print(p1,p2,p3,p4)
```

```
44 44 44 44
```

In [18]:
```
val1 = 10 # Integer data type
print(val1)
print(type(val1)) # type of object
print(sys.getsizeof(val1)) # size of integer object in bytes
print(val1, " is Integer?", isinstance(val1, int))
```

```
10
<class 'int'>
28
10  is Integer? True
```

In [19]:
```
val2 = 92.78 # Float data type
print(val2)
print(type(val2)) # type of object
print(sys.getsizeof(val2)) # size of float object in bytes
print(val2, " is float?", isinstance(val2, float))
```

```
92.78
<class 'float'>
24
92.78  is float? True
```

In [20]:
```
val3 = 25 + 10j # Complex data type
print(val3)
print(type(val3)) # type of object
print(sys.getsizeof(val3)) # size of float object in bytes
print(val3, " is complex?", isinstance(val3, complex))
```

```
(25+10j)
<class 'complex'>
32
(25+10j)  is complex? True
```

In [21]:
```
sys.getsizeof(int())
```

Out[21]:
```
28
```

In [22]:
```
sys.getsizeof(float())
```

Out[22]:
```
24
```

In [23]:
```
sys.getsizeof(complex())
```

Out[23]:
```
32
```

In [24]:
```
bool1= True
bool2 = False
```

In [26]:
```
print(type(bool1))
print(type(bool2))
```

```
<class 'bool'>
<class 'bool'>
```

In [27]:
```
isinstance(bool1,bool)
```

Out[27]:
```
True
```

In [28]:
```
bool(0)
```

Out[28]:
```
False
```

In [30]:
```
bool(1)
```

Out[30]:
```
True
```

In [31]:
```
bool(None)
```

Out[31]:
```
False
```

In [32]:
```
bool(False)
```

Out[32]:
```
False
```

In [33]:
```
#String
str1="Hello"
print(str1)
```

```
Hello
```

In [34]:
```
mystr = "HelloWorld"
print(mystr)
```

```
HelloWorld
```

In [35]:
```
mystr = '''Hello
World '''
print(mystr)
```

```
Hello
World
```

In [37]:
```
mystr = """Hello
                World"""
print(mystr)
```

```
Hello
                World
```

In [38]:
```
 mystr = ('Happy '
'Monday '
'Everyone')
print(mystr)
```

```
Happy Monday Everyone
```

In [39]:
```
mystr2 = 'Woohoo '
mystr2 = mystr2*5
mystr2
```

Out[39]: 'Woohoo Woohoo Woohoo Woohoo Woohoo '

In [40]:
```
len(mystr2)
```

Out[40]: 35

In [41]:
```
str1
```

Out[41]: 'Hello'

In [42]:
```
str1[1]
```

Out[42]: 'e'

In [43]:
```
str1[len(str1)-1]
```

Out[43]: 'o'

In [44]:
```
str1[-1]
```

Out[44]: 'o'

In [45]:
```
str1[0:5]
```

Out[45]: 'Hello'

In [46]:
```
str1[6:12]
```

Out[46]: ''

In [47]:
```
str1[-4:]
```

Out[47]: 'ello'

In [48]:
```
str1[:6]
```

Out[48]: 'Hello'

In [49]:
```
str1
```

Out[49]:
```
'Hello'
```

In [50]:
```
str1[0:5]="hello"
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[50], line 1
----> 1 str1[0:5]="hello"

TypeError: 'str' object does not support item assignment
```

In [51]:
```
del str1
print(str1)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[51], line 2
      1 del str1
----> 2 print(str1)

NameError: name 'str1' is not defined
```

In [52]:
```
s1="abcd"
s2="efgh"
s3=s1+s2
print(s3)
```

```
abcdefgh
```

In [ ]:
```

```