In [2]:
```
import sys
import keyword
import operator
from datetime import datetime
import os
```

keyword

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

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

In [4]:
```
len(keyword.kwlist)
```

Out[4]: 35

Identifiers

In [5]:
```
1var = 10  #identifier can't start with a digit
```

```
  Cell In[5], line 1
    1var = 10
    ^
SyntaxError: invalid decimal literal
```

In [7]:
```
1var12@ = 35 #identifier can't use special symbols
```

```
  Cell In[7], line 1
    1var12@ = 35 #identifier can't use special symbols
    ^
SyntaxError: invalid decimal literal
```

In [8]:
```
import = 125 #keyword can't be used as identifiers
```

```
  Cell In[8], line 1
    import = 125 #keyword can't be used as identifiers
           ^
SyntaxError: invalid syntax
```

In [ ]:
```
"""
correct way of defining an identifir
(Identifir can be a combination of letters in lowercase(a to z) or uppercase)
"""
val2 = 10
```

In [9]:
```
val_ = 99
```

comments in python

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

In [10]:
```
#multiple
#line
#Comment
val1 = 10
```

In [11]:
```
"""
multiple
line
Comment
"""
val1 = 10
```

statements

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

Out[12]: (20, int, '0x7ffc3b0b2c18')

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

Out[13]: (20, int, '0x7ffc3b0b2c18')

In [14]:
```
r, type(r), hex(id(r))
```

Out[14]: (20, int, '0x7ffc3b0b2c18')

In [16]:
```
p = 20
p = p + 10
p
```

Out[16]: 30

variable Assignment

In [18]:
```
intvar = 10
floatvar = 2.57
strvar = "python"

print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.57
python
```

Multiple Assignments

In [20]:
```
intvar, floatvar, strvar = 10, 2.57, "python"
print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.57
python
```

In [21]:
```
p1 = p2 = p3 = p4 = 44
print(p1,p2,p3,p4)
```

```
44 44 44 44
```

datatypes

Numeric

In [22]:
```
val1 = 10
print(val1)
print(type(val1))
print(sys.getsizeof(val1))
print(val1, "is Integer?", isinstance(val1, int))
```

```
10
<class 'int'>
28
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[22], line 5
      3 print(type(val1))
      4 print(sys.getsizeof(val1))
----> 5 print(val1, "is Integer?", isinstance(val1, int))

TypeError: isinstance() arg 2 must be a type, a tuple of types, or a union
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: