```
import sys
import keyword
import operator
from datetime import datetime
import os
```

Keywords : are the reserved words in pyrh and can't be used as an identifier....

```
print(keyword.kwlist)
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'co
```

```
len(keyword.kwlist)
```

```
35
```

Identifiers:An identifiers is a name given to entities like class,function,variables,etc.It helps to defferentiate one entity from another....

```
1var = 10 #identifier can't start with digit
```

```
val2@ = 35 #identifiers can't use special symbol
```

```
import = 125 #keywords can't be used as identifiers
```

```
val2 = 10
val2
```

```
10
```

```
val_=99
val_
```

```
99
```

```
#Comments
val1=10
val1
```

```
10
```

```
#Multiple
#Line
#comment
'''Multi
line
comment'''


"""Multiple
Line
Comment"""
```

Statements:Instructions that a python interpreter can execute ....

```
#Statements
p=20 #create an integer object with value 20 and assings the variable p to p
q=20 #create new refrence q which will point to value 20.p & q will be point
r=q #variable r will also point to the same location where p & q are pointin
print(p , type(p) , hex(id(p))) # variable p is pointing to memory location '0x7fff6d71a'
print(q , type(q),hex(id(q)))
print(r , type(r),hex(id(r)))
```

```
20 <class 'int'> 0xa428c8
20 <class 'int'> 0xa428c8
20 <class 'int'> 0xa428c8
```

```
p = 20
p = p + 10
print(p)
```

```
30
```

```
intvar = 10 #integer variable
floatvar=2.57
strvar = "Python Language"
print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.57
Python Language
```

```
#Multi assingment
intvar , floatvar,strvar,=10,2.34,'Python Language' #Use commas to saperate
print(intvar)
print(floatvar)
print(strvar)
```

```
10
2.34
Python Language
```

```
#all variable pointing same variable
p1=p2=p3=p4=44
print(hex(id(p1)))
print(hex(id(p2)))
print(hex(id(p3)))
print(hex(id(p4)))
```

```
0xa42bc8
0xa42bc8
0xa42bc8
0xa42bc8
```

```
#Data types
val1 = 10
print(type(val1))
print(sys.getsizeof(val1)) #size of integer object in bytes
print(val1,'is integer',isinstance(val1,int)) #Val1 is an instance of int
```

```
<class 'int'>
28
10 is integer True
```

```
val2 = 22.3
print(val2)
print(type(val2))
print(sys.getsizeof(val2))
print(val2,'is integer',isinstance(val1,float))
```

```
22.3
<class 'float'>
24
22.3 is integer False
```

```
val3 = 25+10j
print(val3)
print(sys.getsizeof(val3))
print(type(val3))
print(val3,'is complex',isinstance(val3,complex))
```

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

```
sys.getsizeof(int()) #size of integer object in bytes
```

```
28
```

```
sys.getsizeof(float())
```

```
24
```

```
sys.getsizeof(complex())
```

```
32
```

```
sys.getsizeof(str())
```

```
49
```

Boolean:data type can have only teo possible values true or false...

```
bool1 = True
bool1
```

```
True
```

```
bool2 = False
bool2
```

```
False
```

```
print(type(bool1))
print(type(bool2))
```

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

```
isinstance(bool1,bool)
```

```
True
```

```
bool(0)
```

```
False
```

```
bool(23.2)
```

```
True
```

```
bool('ten')
```

```
True
```

```
bool(0+0j)
```

```
False
```

```
bool(None)
```

```
False
```

```
bool(1)
```

```
True
```

```
str1 = "Hello Python" #String using double quotes
print(str1)
```

```
Hello Python
```

```
mystr='Hello World' #String using single quotes
```

```
mystr = ''' Hello
            Python'''

print(mystr)
```

```
 Hello
                Python
```

```
mystr1 = ('Happy\t'
'Monday\t'
'Everyone')
print(mystr1)
```

```
Happy   Monday  Everyone
```

```
mystr2 = 'Wohoo '
mystr2 = mystr2 * 5
mystr2
```

```
'Wohoo Wohoo Wohoo Wohoo Wohoo '
```

```
len(mystr2)
```

```
30
```

String indexing

```
str1 = 'HELLO PYTHON'
str1[0]
```

```
'H'
```

```
str1[len(str1)-1]
```

```
'N'
```

```
str1[-1]
```

```
'N'
```

```
str1[6]
```

```
'P'
```

```
str1[5]
```

```
' '
```

```
#String Slicing
str2 = 'Hello Python'
str2[0:5] #string slicing-fetch all charachters from 0 to 5 index location execution
```

```
'Hello'
```

```
str1[6:12] #retreive all charachters between 6 to 12 index location execution
```

```
'Python'
```

```
str1 = 'Hello python'
str1[-4:] #retreive last four charachters of string
```

```
'thon'
```

```
str2[-6:]
```

```
'Python'
```

```
str1[:4]
```

```
'Hell'
```

```
str1[:6]
```

```
'Hello '
```

```
#Update and Delete String
str1
```

```
'Hello python'
```

```
str1[0:5] = 'HOLAA'  #String is immutable which means string cannot change
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-2794791579.py in <cell line: 0>()
----> 1 str1[0:5] = 'HOLAA'

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

Next steps: Explain error

```
del str1
print(str1)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-4278673725.py in <cell line: 0>()
      1 del str1
----> 2 print(str1)

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

Next steps: Explain error

```
s1 =  "Hello"
s2 = "Python"
s3 = s1 + s2
print(s3)
```

```
HelloPython
```