```
>>> print "Magna Electronics"
  File "<stdin>", line 1
    print "Magna Electronics"
    ^^^^^^^^^^^^^^^^^^^^^^^^^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(...)?
>>> print("Magna Electronics")
Magna Electronics
>>>
```

```
>>> Input1 = 3;
>>> Input2 = 4;
>>> Input3 = 1;
>>> Total = Input1 + Input2 + Input3;
>>> print(total)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'total' is not defined. Did you mean: 'Total'?
>>> print(Total)
8
```

```
>>> word = 'word'
>>> sentence = "This is a sentence."
>>> paragraph = """This is a paragraph. It is
... made up of multiple lines and sentences."""
>>> print(word)
word
>>> print(sentence)
This is a sentence.
>>> print(paragraph)
This is a paragraph. It is
made up of multiple lines and sentences.
```

```
>>> Total = Input1*\
... Input2+Input3
>>> print(Total)
13
```

```
>>> String1 = 'Magna'
>>> String2 = 'Magna Electronics'
>>> String3 = 'Magna Electronics, Pune' # This is a Full name of Company with City.
>>> print(String1)
Magna
>>> print(String2)
Magna Electronics
>>> print(String3)
Magna Electronics, Pune
```

```
>>> days = ['Sunday','Monday','Tuesday']
>>> days_1 = ["Wednesday","Thursday","Friday"]
>>> print(days)
['Sunday', 'Monday', 'Tuesday']
>>> print(days_1)
['Wednesday', 'Thursday', 'Friday']
>>> print('days_1')
days_1
>>> print("days_1")
days_1
```

```
>>> Input1 = input("Enter Data1:")
Enter Data1:7
>>> Input2 = input("Enter Data2:")
Enter Data2:2
>>> Arithmetic_1 = Input1 + Input2
>>> print(Arithmetic_1)
72
>>> Arithmetic_1 = Input1+Input2
>>> print(Arithmetic_1)
72
>>> Arithmetic_1 = float(Input1) + float(Input2)
>>> print(Arithmetic_1)
9.0
```

```
>>> Data1 = 4; Data2 = 3; Data3 = 2;
>>> Operation = Data1-Data2/Data3
>>> print(Operation)
2.5
>>> Data1 = 4; Data2 = 3; Data3 = 2;
>>> Data1 = 4; Data2 = 3; Data3 = 2;Operation = Data1-Data2/Data3; print(Operation)
2.5
```

```
>>> # Assigning Variables
>>> First = 100
>>> Second = 200;
>>> Third = 300:
  File "<stdin>", line 1
    Third = 300:
              ^
SyntaxError: invalid syntax
>>> Fourth = 400.5
>>> Fifth = 'Fifth'
>>> Sixth = "Sixth"
>>> print First'
  File "<stdin>", line 1
    print First'
              ^
SyntaxError: unterminated string literal (detected at line 1)
>>> print(First)
100
>>> print(Second)
200
>>> print(Third)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Third' is not defined
>>> print(Fourth)
400.5
>>> Print(Fifth)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Print' is not defined. Did you mean: 'print'?
>>> print(Fifth)
Fifth
>>> print(Sixth)
Sixth
```

```
>>> a = b = c = 3
>>> Add = float(a) + float(b)
>>> print(Add)
6.0
```

```
>>> a,b,c,d = 10,3,"Magna",'Electronics'
>>> print(a)
10
>>> print(b)
3
>>> print(c)
Magna
>>> print(d)
Electronics
```

```
>>> Var1 = 2;
>>> Var2 = 3;
>>> Var3 = 4;
>>> Add = long(Var1) + long(Var2)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'long' is not defined
>>> Add = float(Var1) + float(Var2)
>>> print(Add)
5.0
>>> print(int(Add))
5
>>> print(long(Add))
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'long' is not defined
>>> print(complex(Add))
(5+0j)
>>> print(float(Add))
5.0
>>> Add = float(Var1 + Var2)
>>> print(Add)
5.0
>>> Add = int(Var1 + Var2)
>>> print(Add)
5
>>> Add = complex(Var1 + Var2)
>>> print(Add)
(5+0j)
>>> del Var2
>>> Add = complex(Var1 + Var2)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Var2' is not defined. Did you mean: 'Var1'?
```

```
>>> Str = 'Magna Electronics'
>>> print Str
  File "<stdin>", line 1
    print Str
    ^^^^^^^^^
SyntaxError: Missing parentheses in call to 'print'.
>>> print(Str)
Magna Electronics
>>> print(Str[0])
M
>>> print(Str[2])
g
>>> print(Str[0:2])
Ma
>>> print(Str[1:2])
a
>>> print(Str[0:3])
Mag
>>> print(Str[0:])
Magna Electronics
>>> print(Str[2:])
gna Electronics
>>> print(Str[152:])

>>> print(Str[15:])
cs
>>> print(Str*2)
Magna ElectronicsMagna Electronics
>>> print(Str*2 )
Magna ElectronicsMagna Electronics
>>> print( Str*2 )
Magna ElectronicsMagna Electronics
>>> print(Str + 'Pune')
Magna ElectronicsPune
>>> print(Str + ' ' + 'Pune')
Magna Electronics Pune
```

```
>>> List1 = ['Data1','Data2','Data3','Data4']
>>> List2 = ['Data5','Data6']
>>> print(List[1])
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'List' is not defined. Did you mean: 'List1'?
>>> print(List[0])
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'List' is not defined. Did you mean: 'List1'?
>>> print(List1*2)
['Data1', 'Data2', 'Data3', 'Data4', 'Data1', 'Data2', 'Data3', 'Data4']
>>> print(List1 + List2)
['Data1', 'Data2', 'Data3', 'Data4', 'Data5', 'Data6']
>>> print(List1[0])
Data1
>>> print(List1[1:3])
['Data2', 'Data3']
>>> print(List1[1:3] + List2[0:1])
['Data2', 'Data3', 'Data5']
```

```
>>> List1 = [Data1]
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Data1' is not defined
>>> List1 = ['Data1','Data2','Data3']
>>> List1[1] = 'Data4'
>>> List1
['Data1', 'Data4', 'Data3']
>>> List2 = ['Data1','Data2','Data3']
>>> List2 = ('Data1','Data2','Data3')
>>> List2[1] = 'Data4'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
>>> List1[1] = 1000
>>> List1
['Data1', 1000, 'Data3']
```

```
>>> dictionary = {}
>>> dictionary['First'] = "This is First Data"
>>> dictionary['Second'] = "This is Second Data"
>>> String = {'Magna':'Electronics','Magna':412101,'Magna':'Pune'}
>>> print(dictionary['First'])
This is First Data
>>> print(dictionary['Second'])
This is Second Data
>>> print(string['Magna'])
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'string' is not defined. Did you mean: 'String'?
>>> print(String.keys())
dict_keys(['Magna'])
>>> print(String.values())
dict_values(['Pune'])
>>> String1 = {'Magna_1':'Electronics','Magna_2':412101,'Magna_3':'Pune'}
>>> print(String.keys())
dict_keys(['Magna'])
>>> print(String1.keys())
dict_keys(['Magna_1', 'Magna_2', 'Magna_3'])
>>> print(String.keys())
dict_keys(['Magna'])
>>> print(String.values())
dict_values(['Pune'])
>>> print(String_1.values())
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'String_1' is not defined. Did you mean: 'String1'?
>>> print(String1.values())
dict_values(['Electronics', 412101, 'Pune'])
```

```
>>> A = 3;
>>> B = 4.2;
>>> C = A+B
>>> print(C)
7.2
>>> C = A-B
>>> print(C)
-1.2000000000000002
>>> C = A*B
>>> print(C)
12.600000000000001
>>> C = A/B
>>> print(C)
0.7142857142857143
>>> C = A&B
Traceback (most rece
  File "<stdin>", li
TypeError: unsupport
>>> C = A%B
>>> print(C)
3.0
>>> C = B%A
>>> print(C)
1.2000000000000002
>>> C = A**B
>>> print(C)
100.90420610885693
>>> C =B//A
>>> print(C)
1.0
>>> C = A//B
>>> print(C)
0.0
>>>
```

```
>>> a = 21; b = 10; c = 0;
>>>
>>> c = a + b
>>> print("Line 1 : Value of c is ", c)
Line 1 : Value of c is  31
>>>
>>> c += a
>>> print("Line 2 : Value of c is ", c)
Line 2 : Value of c is  52
>>>
>>> c *= a
>>> print("Line 3 : Value of c is ", c)
Line 3 : Value of c is  1092
>>>
>>> c /= a
>>> print("Line 4 : Value of c is ", c)
Line 4 : Value of c is  52.0
>>>
>>> c  = 2
>>> c %= a
>>> print("Line 5 : Value of c is ", c)
Line 5 : Value of c is  2
>>>
>>> c **= a
>>> print("Line 6 : Value of c is ", c)
Line 6 : Value of c is  2097152
>>>
>>> c //= a
>>> print("Line 7 : Value of c is ", c)
Line 7 : Value of c is  99864
```

```
>>> a = 60            # 60 = 0011 1100
>>> b = 13            # 13 = 0000 1101
>>> c = 0
>>>
>>> c = a & b;        # 12 = 0000 1100
>>> print("Line 1 - Value of c is ", c)
Line 1 - Value of c is  12
>>>
>>> c = a | b;        # 61 = 0011 1101
>>> print("Line 2 - Value of c is ", c)
Line 2 - Value of c is  61
>>>
>>> c = a ^ b;        # 49 = 0011 0001
>>> print("Line 3 - Value of c is ", c)
Line 3 - Value of c is  49
>>>
>>> c = ~a;           # -61 = 1100 0011
>>> print("Line 4 - Value of c is ", c)
Line 4 - Value of c is  -61
>>>
>>> c = a << 2;       # 240 = 1111 0000
>>> print("Line 5 - Value of c is ", c)
Line 5 - Value of c is  240
>>>
>>> c = a >> 2;       # 15 = 0000 1111
>>> print("Line 6 - Value of c is ", c)
Line 6 - Value of c is  15
```

```
>>> a = 20; b = 10; c = 15; d = 5;
>>> e = 0
>>>
>>> e = (a + b) * c / d       #( 30 * 15 ) / 5
>>> print("Value of (a + b) * c / d is ",  e)
Value of (a + b) * c / d is  90.0
>>>
>>> e = ((a + b) * c) / d     # (30 * 15 ) / 5
>>> print("Value of ((a + b) * c) / d is ",  e)
Value of ((a + b) * c) / d is  90.0
>>>
>>> e = (a + b) * (c / d);    # (30) * (15/5)
>>> print("Value of (a + b) * (c / d) is ",  e)
Value of (a + b) * (c / d) is  90.0
>>>
>>> e = a + (b * c) / d;      #  20 + (150/5)
>>> print("Value of a + (b * c) / d is ",  e)
Value of a + (b * c) / d is  50.0
```

```
>>> var1 = 100
>>> if var1:
...    print("1 : Got a true expression value")
...    print("var1")
...
1 : Got a true expression value
var1
>>>
```

```
>>> var2 = 0
>>> if var2:
...    print("2 : Got a true expression value")
...    print ("var2")
...    print ("Good bye!")
...
>>>
>>>
```

```
>>> var1 = 100
>>> if var1:
...    print("1 : Got a true expression value")
...    print("var1")
... else:
...    print("1 : Got a false expression value")
...    print("var1")
...
1 : Got a true expression value
var1
```

```
>>> var2 = 0
>>> if var2:
...    print("2 : Got a true expression value")
...    print("var2")
... else:
...    print("2 : Got a false expression value")
...    print("var2")
...
2 : Got a false expression value
var2
>>> print("Good bye!")
Good bye!
```

```
>>> var = 100
>>> if var == 200:
...    print("1 - Got a true expression value")
...    print("var")
... elif var == 150:
...    print("2 - Got a true expression value")
...    print("var")
... elif var == 100:
...    print("3 - Got a true expression value")
...    print("var")
... else:
...    print("4 - Got a false expression value")
...    print("var")
...
3 - Got a true expression value
var
```

```
>>> count = 0
>>> while (count < 9):
...    print('The count is:', count)
...    count = count + 1
...
The count is: 0
The count is: 1
The count is: 2
The count is: 3
The count is: 4
The count is: 5
The count is: 6
The count is: 7
The count is: 8
```

```
>>> var = 1
>>> while var == 1:  # This constructs an infinite loop
...     num = input("Enter a number  :")
...     print("You entered a Value: ", num)
...
Enter a number  :3.5
You entered a Value:  3.5
Enter a number  :7
You entered a Value:  7
Enter a number  :1000
You entered a Value:  1000
Enter a number  :0
You entered a Value:  0
Enter a number  :0.1
You entered a Value:  0.1
Enter a number  :0.0000000001
You entered a Value:  0.0000000001
Enter a number  :1.00001
You entered a Value:  1.00001
Enter a number  :Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
KeyboardInterrupt
>>>
```

```
>>> count = 0
>>> while count < 5:
...     print(count, " is  less than 5")
...     count = count + 1
... else:
...     print(count, " is not less than 5")
...
0  is  less than 5
1  is  less than 5
2  is  less than 5
3  is  less than 5
4  is  less than 5
5  is not less than 5
```

```
>>> flag = 1
>>> while (flag):
...   print('Given flag is really true!')
...   print("Good bye!")
...
Given flag is really true!
Good bye!
Given flag is really true!
Good bye!
Given flag is really true!
Good bye!
Given flag is really true!
Good bye!
Given flag is really true!
Good bye!
```

```
>>> for letter in 'Magna':     # First Example
...     print('Current Letter :', letter)
...
Current Letter : M
Current Letter : a
Current Letter : g
Current Letter : n
Current Letter : a
>>> for letter in 'Magna Electronics':     # Second Example
...     print('Current Letter :', letter)
...
Current Letter : M
Current Letter : a
Current Letter : g
Current Letter : n
Current Letter : a
Current Letter :
Current Letter : E
Current Letter : l
Current Letter : e
Current Letter : c
Current Letter : t
Current Letter : r
Current Letter : o
Current Letter : n
Current Letter : i
Current Letter : c
Current Letter : s
```

```
>>> fruits = ['banana', 'apple',  'mango']
>>> for fruit in fruits:        # Second Example
...     print('Current fruit :', fruit)
...
Current fruit : banana
Current fruit : apple
Current fruit : mango
```

```
>>> for index in range(len(fruits)):
...     print('Current fruit :', fruits[index])
...
Current fruit : banana
Current fruit : apple
Current fruit : mango
```

```
>>> for num in range(10,20):     #to iterate between 10 to 20
...    for i in range(2,num):    #to iterate on the factors of the number
...       if num%i == 0:         #to determine the first factor
...          j=num/i             #to calculate the second factor
...          print('%d is a Even number' % num)
...          break #to move to the next number, the #first FOR
...       else:                  # else part of the loop
...          print('%d is a Odd number'% num)
...          break
...
10 is a Even number
11 is a Odd number
12 is a Even number
13 is a Odd number
14 is a Even number
15 is a Odd number
16 is a Even number
17 is a Odd number
18 is a Even number
19 is a Odd number
```

```
>>> var = 10                    # Second Example
>>> while var > 0:
...    print('Current variable value %d:', var)
...    var = var -1
...    if var == 5:
...       break
...
Current variable value %d: 10
Current variable value %d: 9
Current variable value %d: 8
Current variable value %d: 7
Current variable value %d: 6
```

```
>>> for letter in 'Magna':
...    if letter == 'n':
...       pass
...       print('This is pass block')
...       print('Current Letter :', letter)
...
This is pass block
Current Letter : n
```

```
>>> for letter in 'Magna':     # First Example
...    if letter == 'n':
...       continue
...    print('Current Letter :', letter)
...
Current Letter : M
Current Letter : a
Current Letter : g
Current Letter : a
```

```
>>> for letter in 'Magna':     # First Example
...    if letter == 'n':
...       break
...    print('Current Letter :', letter)
...
Current Letter : M
Current Letter : a
Current Letter : g
```

```
>>> var = 10                    # Second Example
>>> while var > 0:
...    var = var -1
...    if var == 5:
...       continue
...    print('Current variable value :', var)
...
Current variable value : 9
Current variable value : 8
Current variable value : 7
Current variable value : 6
Current variable value : 4
Current variable value : 3
Current variable value : 2
Current variable value : 1
Current variable value : 0
```

```
>>> abs(-45)
45
>>> abs(100.12)
100.12
>>> abs(119L)
  File "<stdin>", line 1
    abs(119L)
           ^
SyntaxError: invalid decimal literal
>>> abs(11900000)
11900000
>>> abs(-13.87)
13.87
```

```
>>> ceil(-45.17)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'ceil' is not defined
>>> math.ceil(-45.17)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'math' is not defined
>>> import math
>>> math.ceil(-45.17)
-45
>>> math.ceil(100.12)
101
>>> math.ceil(100.72)
101
>>> math.ceil(119L)
  File "<stdin>", line 1
    math.ceil(119L)
                ^
SyntaxError: invalid decimal literal
>>> math.ceil(pi)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'pi' is not defined
>>> math.ceil(math.pi)
4
```

```
>>> cmp(80,100)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'cmp' is not defined
>>> 80<100
True
>>> 80>100
False
>>> a = 80>100
>>> print('a')
a
>>> print(a)
False
>>> b = 80<100
>>> print(b)
True
```

```
>>> exp(-45.17)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'exp' is not defined
>>> math.exp(-45.17)
2.4150062132629406e-20
>>> math.exp(100.12)
3.0308436140742566e+43
>>> math.exp(100.72)
5.522557130248187e+43
>>> math.exp(119L)
  File "<stdin>", line 1
    math.exp(119L)
                ^
SyntaxError: invalid decimal literal
>>> math.exp(math.pi)
23.140692632779267
```

```
>>> math.abs(-45.17)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: module 'math' has no attribute 'abs'.
Did you mean: 'fabs'?
>>> abs(-45.17)
45.17
>>> math.fabs(-45.17)
45.17
>>> abs(100.12)
100.12
>>> math.fabs(100.12)
100.12
>>> abs(-57)
57
>>> math.fabs(-57)
57.0
>>> abs(math.pi)
3.141592653589793
>>> math.fabs(math.pi)
3.141592653589793
```

```
>>> math.floor(-45.17)
-46
>>> math.floor(100.12)
100
>>> math.floor(100.72)
100
>>> math.floor(math.pi)
3
```

```
>>> math.log(2)
0.6931471805599453
>>> math.log(3)
1.0986122886681098
>>> math.log(1)
0.0
>>> math.log(0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: math domain error
>>> math.log(100.12)
4.6063694665635735
>>> math.log(100.72)
4.612344389736092
>>> math.log(-45.17)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: math domain error
>>> math.log(math.pi)
1.1447298858494002
>>> math.log10(100.12)
2.0005208409361854
>>> math.log10(100.72)
2.003115717099806
>>> math.log10(math.pi)
0.49714987269413385
```

```
>>> max(80,100,120)
120
>>> max(-100,-80,-60)
-60
>>> max(-50,0,50)
50
```

```
>>> min(80,100,120)
80
>>> min(-100,-80,-60)
-100
>>> min(-50,0,50)
-50
```

```
>>> math.modf(100.12)
(0.12000000000000455, 100.0)
>>> math.modf(100.72)
(0.7199999999999989, 100.0)
>>> math.modf(math.pi)
(0.14159265358979312, 3.0)
```

```
>>> math.pow(100,2)
10000.0
>>> math.pow(100,-2)
0.0001
>>> math.pow(2,4)
16.0
>>> math.pow(3,0)
1.0
```

```
>>> round(80.23456,1)
80.2
>>> round(80.23456,2)
80.23
>>> round(80.23456,3)
80.235
>>> round(80.23456,5)
80.23456
>>> round(100.00056)
100
>>> round(100.00056,1)
100.0
>>> round(100.00056,2)
100.0
>>> round(100.00056,3)
100.001
>>> round(100.00056,4)
100.0006
>>> round(100.00056,5)
100.00056
```

```
>>> math.sqrt(100)
10.0
>>> math.sqrt(7)
2.6457513110645907
>>> math.sqrt(math.pi)
1.7724538509055159
```

```
>>> import random
>>> random.random()
0.11071511207721285
>>> random.random()
0.07301298616090024
>>> random.random()
0.5304741983793724
```

```
>>> import math
>>> acos(0.64)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'acos' is not defined
>>> math.acos(0.64)
0.8762980611683406
>>> math.acos(0)
1.5707963267948966
>>> math.acos(-1)
3.141592653589793
>>> math.acos(1)
0.0
>>> math.asin(0.64)
0.694498265626556
>>> math.asin(0)
0.0
>>> math.asin(-1)
-1.5707963267948966
>>> math.asin(1)
1.5707963267948966
>>> math.atan(0.64)
0.5693131911006619
>>> math.atan(0)
0.0
>>> math.atan(10)
1.4711276743037347
>>> math.atan(-1)
-0.7853981633974483
>>> math.atan(1)
0.7853981633974483
>>> math.atan2(-0.50,-0.50)
-2.356194490192345
>>> math.atan2(0.50,0.50)
0.7853981633974483
>>> math.atan2(5,5)
0.7853981633974483
>>> math.atan2(-10,10)
-0.7853981633974483
>>> math.atan2(-10,20)
-0.4636476090008061
```

```
>>> math.cos(3)
-0.9899924966004454
>>> math.cos(-3)
-0.9899924966004454
>>> math.cos(0)
1.0
>>> math.cos(math.pi)
-1.0
>>> math.cos(2*math.pi)
1.0
>>> math.hypot(3,2)
3.605551275463989
>>> math.hypot(-3,3)
4.242640687119285
>>> math.hypot(0,2)
2.0
>>> math.hypot(4,3)
5.0
>>> math.sin(3)
0.1411200080598672
>>> math.sin(-3)
-0.1411200080598672
>>> math.sin(0)
0.0
>>> math.sin(math.pi)
1.2246467991473532e-16
>>> math.sin(math.pi/2)
1.0
>>> math.tan(3)
-0.1425465430742778
>>> math.tan(-3)
0.1425465430742778
>>> math.tan(0)
0.0
>>> math.tan(math.pi)
-1.2246467991473532e-16
>>> math.tan(math.pi/2)
1.633123935319537e+16
>>> math.tan(math.pi/4)
0.9999999999999999
```

```
>>> math.degrees(3)
171.88733853924697
>>> math.degrees(-3)
-171.88733853924697
>>> math.degrees(0)
0.0
>>> math.degrees(math.pi)
180.0
>>> math.degrees(math.pi/2)
90.0
>>> math.degrees(math.pi/4)
45.0
>>> math.radians(3)
0.05235987755982989
>>> math.radians(-3)
-0.05235987755982989
>>> math.radians(0)
0.0
>>> math.radians(math.pi)
0.05483113556160755
>>> math.radians(math.pi/2)
0.027415567780803774
>>> math.radians(math.pi/4)
0.013707783890401887
```

```
>>> var1 = 'Magna'
>>> var2 = "Magna Electronics"
>>>
>>> print("var1[0]: ", var1[0])
var1[0]:  M
>>> print("var2[1:5]: ", var2[1:5])
var2[1:5]:  agna
>>>
>>> print("First Digit: ", var1[0])
First Digit:  M
>>> print("Four Digits: ", var2[1:5])
Four Digits:  agna
```

```
>>> print("Updated String :- ", var1[:6] + 'Electronics, Pune')
Updated String :-  Magna Electronics, Pune
```

```
>>> print("I am Employee of %s at Location of %s Since %d" % ('Magna Electronics','Pune', 2021))
I am Employee of Magna Electronics at Location of Pune Since 2021
```

```
>>> para_str = """This is a Magna Electronics Located at Pune."""
>>> print(para_str)
This is a Magna Electronics Located at Pune.
```

```
>>> print('c:\\MITE')
c:\MITE
>>> print(r'c:\\MITE')
c:\\MITE
```

```
>>> print('Magna Electronics, Pune')
Magna Electronics, Pune
>>> print(u'Magna Electronics, Pune')
Magna Electronics, Pune
```

```
>>> print("str.capitalize() : ", str.capitalize())
str.capitalize() :  This is a magna electronics, pune!!!
```

```
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(40, 'P') : ", str.center(40, 'a'))
str.center(40, 'P') :  this is a magna electronics,... pune!!!a
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(40, 'P') : ", str.center(40, 'P'))
str.center(40, 'P') :  this is a magna electronics,... pune!!!P
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(45, 'P') : ", str.center(45, 'P'))
str.center(45, 'P') :  PPPthis is a magna electronics,... pune!!!PPP
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(50, 'P') : ", str.center(45, 'P'))
str.center(50, 'P') :  PPPthis is a magna electronics,... pune!!!PPP
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(60, 'P') : ", str.center(60, 'P'))
str.center(60, 'P') :  PPPPPPPPPPthis is a magna electronics,... pune!!!PPPPPPPPPP
>>>
>>> str = "this is a magna electronics,... pune!!!"
>>> print("str.center(20, 'P') : ", str.center(20, 'P'))
str.center(20, 'P') :  this is a magna electronics,... pune!!!
```

```
>>> print(str1.find(str2))
17
>>> print(str1.find(str2, 10))
17
>>> print(str1.find(str2, 40))
-1
```

```
>>> str1 = "this is a magna electronics,... pune";
>>> str2 = "lectro";
>>>
>>> print(str1.index(str2))
17
>>> print(str1.index(str2, 10))
17
>>> print(str1.index(str2, 40))
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: substring not found
```

```
>>> str = "this is a magna electronics,... pune!!!";
>>> sub = "i";
>>> print("str.count(sub, 4, 40) : ", str.count(sub, 4, 40))
str.count(sub, 4, 40) :  2
>>> sub = "pune";
>>> print("str.count(sub) : ", str.count(sub))
str.count(sub) :  1
```

```
>>> str = "this is a magna electronics,... pune!!!";
>>> suffix = "pune!!!";
>>> print(str.endswith(suffix))
True
>>> print(str.endswith(suffix,20))
True
>>> suffix = "magna";
>>> print(str.endswith(suffix, 10, 15))
True
>>> print(str.endswith(suffix, 10, 17))
False
```

```
>>> str = "this is a mag\tna electronics,... pune!!!";
>>>
>>> print("Original string: " + str)
Original string: this is a mag  na electronics,... pune!!!
>>> print("Defualt exapanded tab: " +  str.expandtabs())
Defualt exapanded tab: this is a mag   na electronics,... pune!!!
>>> print("Double exapanded tab: " +  str.expandtabs(32))
Double exapanded tab: this is a mag                    na electronics,... pune!!!
>>>
```

```
>>> str = "pune2022";  # No space in this string
>>> print(str.isalnum())
True
>>> str = "this is a magna electronics,... pune!!!";
>>> print(str.isalnum())
False
```

```
>>> str = "india";  # No space & digit in this string
>>> print(str.isalpha())
True
>>>
>>> str = "this is string example....wow!!!";
>>> print(str.isalpha())
False
```

```
>>> str = "123456";  # Only digit in this string
>>> print(str.isdigit())
True
>>>
>>> str = "this is a magna electronics,... pune!!!";
>>> print(str.isdigit())
False
```

```
>>> str = "THIS is a magna electronics,... pune!!!";
>>> print(str.islower())
False
>>>
>>> str = "this is a magna electronics,... pune!!!";
>>> print(str.islower())
True
```

```
>>> s = "-";
>>> seq = ("P", "u", "n", "e"); # This is sequence of strings.
>>> print(s.join( seq ))
P-u-n-e
```

```
>>> str = u"pune2022";
>>> print(str.isnumeric())
False
>>>
>>> str = u"23443434";
>>> print(str.isnumeric())
True
>>>
>>> str = "       ";
>>> print(str.isspace())
True
>>>
>>> str = "This is a magna electronics,... pune!!!";
>>> print(str.isspace())
False
```

```
>>> str = "This Is A Magna Electronics,... Pune!!!";
>>> print(str.istitle())
True
>>>
>>> str = "This is a magna electronics,... pune!!!";
>>> print(str.istitle())
False
>>>
>>> str = "THIS IS A MAGNA ELECTRONICS,... PUNE!!!";
>>> print(str.isupper())
True
>>>
>>> str = "THIS is a magna electronics,... pune!!!";
>>> print(str.isupper())
False
```

```
>>> str = "this is a magna electronics,... pune!!!";
>>> print("Length of the string: ", len(str))
Length of the string:  39
>>>
>>> str = "this is a magna electronics,... pune!!!";
>>> print(str.ljust(50, '0'))
this is a magna electronics,... pune!!!00000000000
```

```
>>> str = "THIS IS A MAGNA ELECTRONICS....PUNE!!!";
>>> print(str.lower())
this is a magna electronics....pune!!!
>>> str = "     this is a magna electronics....Pune!!!     ";
>>> print(str.lstrip())
this is a magna electronics....Pune!!!
>>> str = "88888888this is a magna electronics...pune8888888";
>>> print(str.lstrip('8'))
this is a magna electronics...pune8888888
```

```
>>> str = "this is really a magna electronics....pune!!!";
>>> print("Max character: " + max(str))
Max character: y
>>>
>>> str = "this is a magna electronics....pune!!!";
>>> print("Max character: " + max(str))
Max character: u
>>>
>>>
>>> str = "thisisamagnaelectronics....pune!!!";
>>> print("Min character: " + min(str))
Min character: !
>>>
>>> str = "thisisamagnaelectronics....pune";
>>> print("Min character: " + min(str))
Min character: .
>>>
>>> str = "thisisamagnaelectronicspune";
>>> print("Min character: " + min(str))
Min character: a
```

```
>>> str = "this is magna electronics....pune!!! this is really a magna electronics"
>>> print(str.replace("is", "was"))
thwas was magna electronics....pune!!! thwas was really a magna electronics
>>> print(str.replace("is", "was", 3))
thwas was magna electronics....pune!!! thwas is really a magna electronics
```

```
>>> str1 = "this is a magna electronics....pune!!!";
>>> str2 = "is";
>>>
>>> print(str1.rfind(str2))
5
>>> print(str1.rfind(str2, 0, 10))
5
>>> print(str1.rfind(str2, 10, 0))
-1
>>>
>>> print(str1.find(str2))
2
>>> print(str1.find(str2, 0, 10))
2
>>> print(str1.find(str2, 10, 0))
-1
```

```
>>> str1 = "this is a magna electronics....pune!!!";
>>> str2 = "is";
>>>
>>> print(str1.rindex(str2))
5
>>> print(str1.index(str2))
2
```

```
>>> str = "this is a magna electronics,... pune!!!";
>>> print(str.ljust(50, '0'))
this is a magna electronics,... pune!!!00000000000
>>>
>>> str = "this is a magna electronics....pune!!!";
>>> print(str.rjust(50, '0'))
000000000000this is a magna electronics....pune!!!
```

```
>>> str = "     this is string example....wow!!!     ";
>>> print(str.rstrip())
     this is string example....wow!!!
>>>
>>> str = "88888888this is string example....wow!!!8888888";
>>> print(str.rstrip('8'))
88888888this is string example....wow!!!
```

```
>>> str = "Line1-abcdef \nLine2-abc \nLine4-abcd";
>>> print(str.split( ))
['Line1-abcdef', 'Line2-abc', 'Line4-abcd']
>>> print(str.split(' ', 1 ))
['Line1-abcdef', '\nLine2-abc \nLine4-abcd']
```

```
>>> str = u"this2009";
>>> print(str.isdecimal());
False
>>>
>>> str = u"23443434";
>>> print(str.isdecimal());
True
```

```
>>> print(str.startswith( 'this' ))
True
>>> print(str.startswith( 'is', 2, 4 ))
True
>>> print(str.startswith( 'this', 2, 4 ))
False
```

```
>>> str = "0000000this is a magna electronics....pune!!!0000000";
>>> print(str.strip( '0' ))
this is a magna electronics....pune!!!
```

```
>>> str = "this is a magna electronics....pune!!!";
>>> print(str.swapcase())
THIS IS A MAGNA ELECTRONICS....PUNE!!!
>>>
>>> str = "THIS IS A MAGNA ELECTRONICS....PUNE!!!";
>>> print(str.swapcase())
this is a magna electronics....pune!!!
```

```
>>> str = "this is a magna electronics....pune!!!";
>>> print(str.title())
This Is A Magna Electronics....Pune!!!
```

```
>>> str = "this is a magna electronics....pune!!!";
>>> print("str.capitalize() : ", str.upper())
str.capitalize() :  THIS IS A MAGNA ELECTRONICS....PUNE!!!
```

```
>>> print(str.zfill(40))
00this is a magna electronics....pune!!!
>>> print(str.zfill(50))
000000000000this is a magna electronics....pune!!!
```

```
>>> list1 = ['physics', 'chemistry', 1997, 2000];
>>> list2 = [1, 2, 3, 4, 5 ];
>>> list3 = ["a", "b", "c", "d"]
```

```
>>> list1 = ['physics', 'chemistry', 1997, 2000];
>>> list2 = [1, 2, 3, 4, 5, 6, 7 ];
>>> print("list1[0]: ", list1[0])
list1[0]:  physics
>>> print("list2[1:5]: ", list2[1:5])
list2[1:5]:  [2, 3, 4, 5]
```

```
>>> list = ['physics', 'chemistry', 1997, 2000];
>>> print("Value available at index 2 : ")
Value available at index 2 :
>>> print(list[2])
1997
>>> list[2] = 2001;
>>> print("New value available at index 2 : ")
New value available at index 2 :
>>> print(list[2])
2001
```

```
>>> list1 = ['physics', 'chemistry', 1997, 2000];
>>> print(list1)
['physics', 'chemistry', 1997, 2000]
>>> del list1[2];
>>> print("After deleting value at index 2 : ")
After deleting value at index 2 :
>>> print(list1)
['physics', 'chemistry', 2000]
```

```
>>> list1, list2 = [123, 'xyz', 'zara'], [456, 'abc']
>>> print("First list length : ", len(list1))
First list length :  3
>>> print("Second list length : ", len(list2))
Second list length :  2
```

```
>>> list1 = [1,2,3,5,'magna',5]
>>> max(list1)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: '>' not supported between instances of 'str' and 'int'
>>>
```

```
>>> min(list1)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: '<' not supported between instances of 'str' and 'int'
```

```
>>> list1, list2 = [123, 'Magna', 'electronics'], [456, 'Pune']
>>> print("First list length : ", len(list1))
First list length :  3
>>> print("Second list length : ", len(list2))
Second list length :  2
```

```
>>> aTuple = (123, 'Magna', 'electronics', 'Pune');
>>> aList = list(aTuple)
>>> print("List elements : ", aList)
List elements :  [123, 'Magna', 'electronics', 'Pune']
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune'];
>>> aList.append( 2009 );
>>> print("Updated List : ", aList)
Updated List :  [123, 'Magna', 'electronics', 'Pune', 2009]
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune', 123];
>>> print("Count for 123 : ", aList.count(123))
Count for 123 :  2
>>> print("Count for electronics : ", aList.count('electronics'))
Count for electronics :  1
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune', 123];
>>> bList = [2009, 'manni'];
>>> aList.extend(bList)
>>> print("Extended List : ", aList)
Extended List :  [123, 'Magna', 'electronics', 'Pune', 123, 2009, 'manni']
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune'];
>>> print("Index for Magna : ", aList.index( 'Magna' ))
Index for Magna :  1
>>> print("Index for electronics : ", aList.index( 'electronics' ))
Index for electronics :  2
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune']
>>> aList.insert( 3, 2009)
>>> print("Final List : ", aList)
Final List :  [123, 'Magna', 'electronics', 2009, 'Pune']
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune'];
>>> print("A List : ", aList.pop())
A List :  Pune
>>> print("B List : ", aList.pop(2))
B List :  electronics
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune', 'Magna'];
>>> aList.remove('Magna');
>>> print("List : ", aList)
List :  [123, 'electronics', 'Pune', 'Magna']
>>> aList.remove('Pune');
>>> print("List : ", aList)
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune', 'Magna'];
>>> aList.reverse();
>>> print("List : ", aList)
List :  ['Magna', 'Pune', 'electronics', 'Magna', 123]
```

```
>>> aList = [123, 'Magna', 'electronics', 'Pune', 'Magna'];
>>> aList.sort();
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: '<' not supported between instances of 'str' and 'int'
>>> print("List : ", aList)
List :  [123, 'Magna', 'electronics', 'Pune', 'Magna']
>>> aList = ['Magna', 'electronics', 'Pune', 'Magna'];
>>> aList.sort();
>>> print("List : ", aList)
List :  ['Magna', 'Magna', 'Pune', 'electronics']
```