

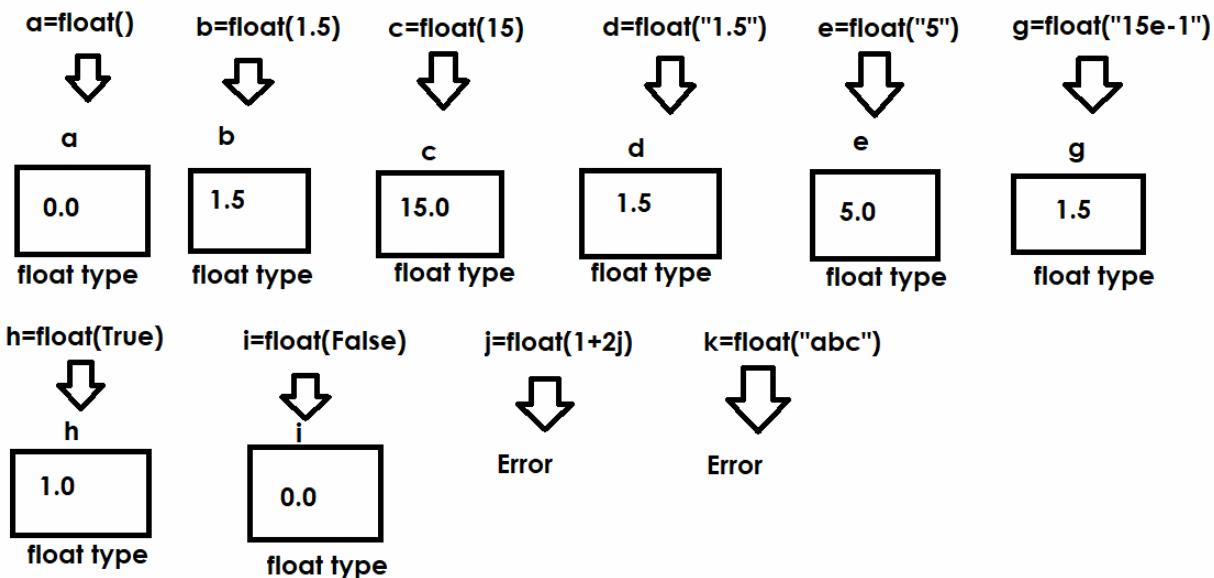
float() function

float() is a predefined function in python

This function is used to perform the following conversions

1. Float to float
2. Int to float
3. String to float
4. Bool to float

Syntax: float(value=0.0)



Example:

```
a=float()  
b=float(1.5)  
c=float(15)  
d=float("1.5")  
e=float("1.5e1")  
f=float("15e-1")  
g=float("15")  
h=float(True)  
i=float(False)  
print(a,b,c,d,e,f,g,h,i,sep="\n")  
#j=float("ab") Error
```

Output

```
0.0  
1.5  
15.0  
1.5  
15.0  
1.5  
15.0  
1.0  
0.0
```

Example

```
# Write a program to find area of triangle  
# area=0.5*base*height  
  
#input  
base=float(input("Input Base of Triangle :"))  
height=float(input("Input Height of Triangle :"))  
  
#Process  
area=0.5*base*height  
  
#Output  
print("Area of triangle is ",area)
```

Output

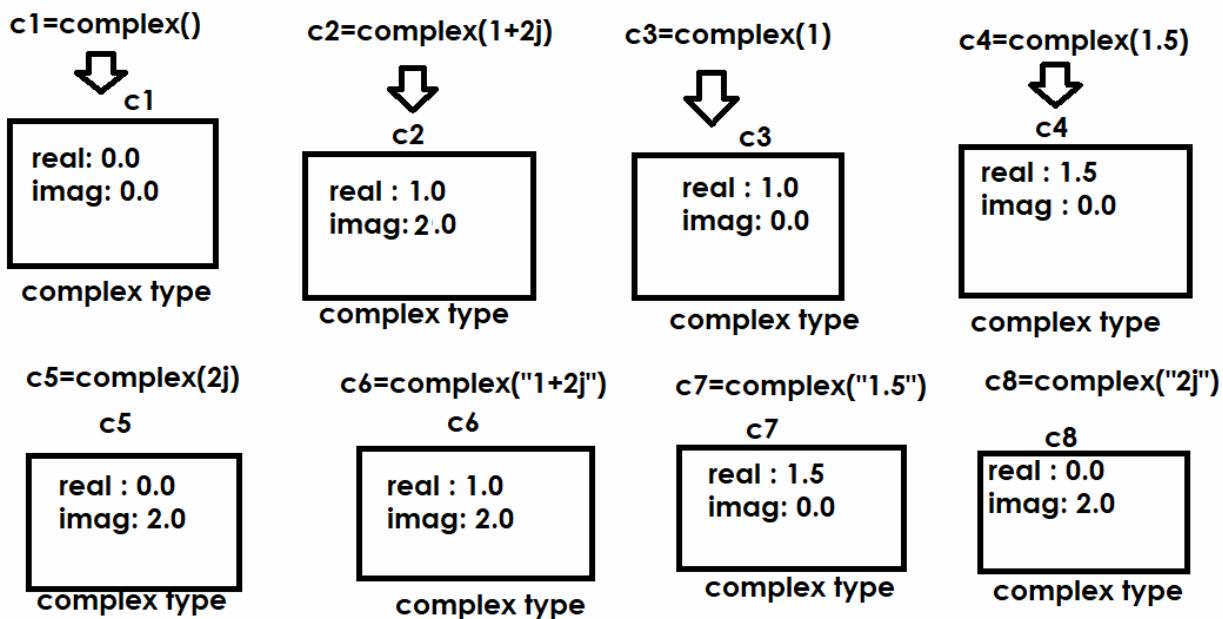
```
Input Base of Triangle :1.2  
Input Height of Triangle :1.3  
Area of triangle is  0.78
```

complex() function

complex() is a predefined function in python. This function is used to perform the following conversions

1. Complex to complex
2. Int to complex
3. Float to complex
4. Boolean to complex
5. String to complex

Syntax: <variable-name>=complex(value)



Example:

```

c1=complex()
c2=complex(1+2j)
c3=complex("1+2j")
c4=complex("2j")
c5=complex("1")
c6=complex(1)
c7=complex(1.5)
c8=complex(True)
c9=complex(False)
print(c1,c2,c3,c4,c5,c6,c7,c8,c9,sep="\n")
#c10=complex("ab")

```

Output

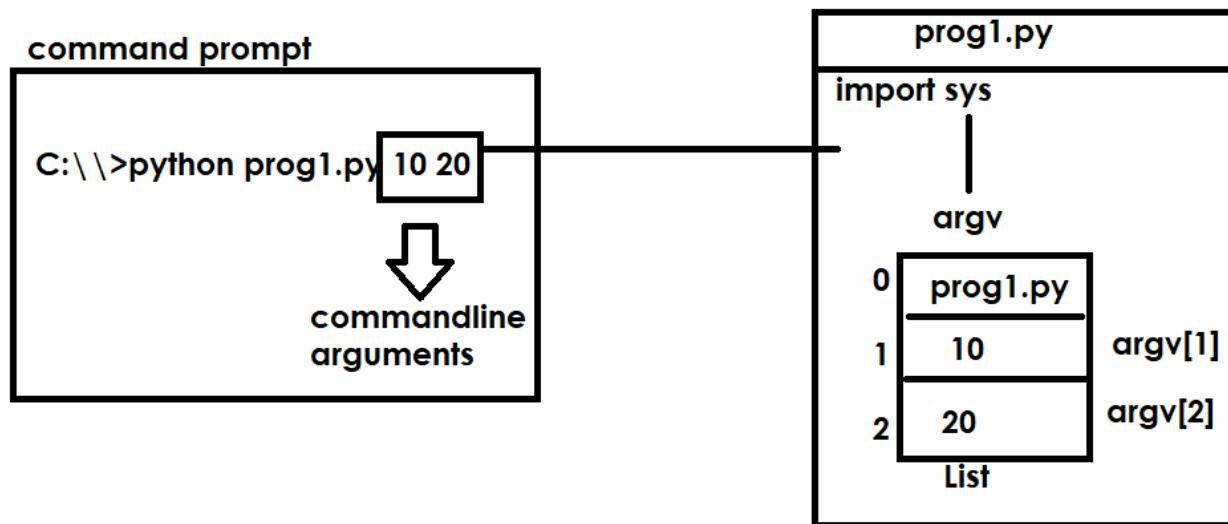
```
0j  
(1+2j)  
(1+2j)  
2j  
(1+0j)  
(1+0j)  
(1.5+0j)  
(1+0j)  
0j
```

Command line arguments

What are command line arguments?

The arguments or values send from command prompt to program are called command line arguments

Command line arguments are of type string



What is argv?

“`argv`” is a predefined variable exist in “`sys`” module

Command line arguments are stored inside `argv` variable.

“`argv`” is variable of type list

Inside this list all values are stored in “string” type.

Example:

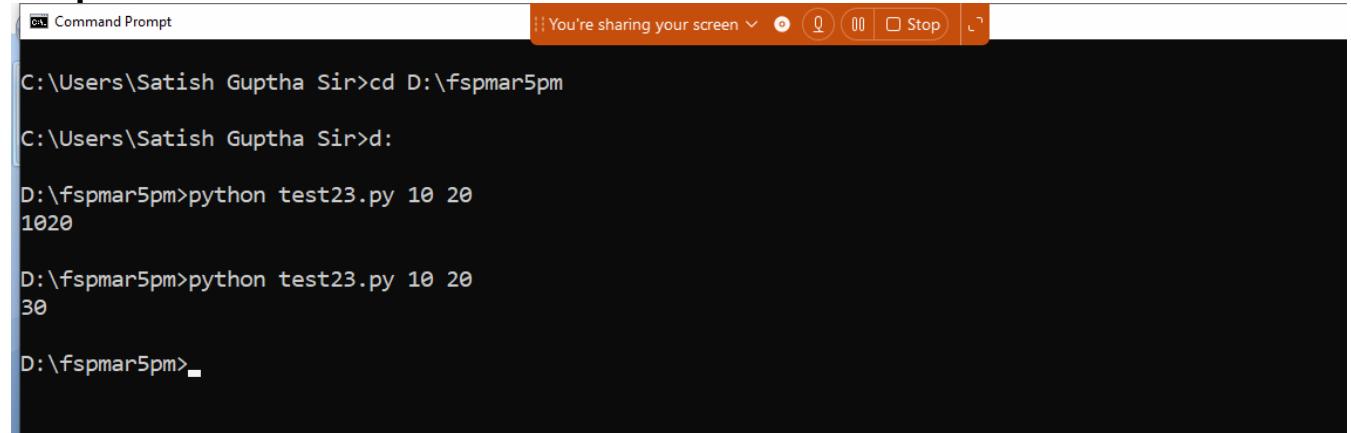
```
# Write a program to print sum of two integers  
# input two integers from command prompt as command
```

```
# line arguments

import sys

res=int(sys.argv[1])+int(sys.argv[2])
print(res)
```

Output



The screenshot shows a Windows Command Prompt window. The title bar says "Command Prompt". The status bar at the top right says "You're sharing your screen". The command line shows the user navigating to D:\fspmar5pm, running python test23.py, and then entering the numbers 10 and 20. The output is 1020.

```
C:\Users\Satish Gupta Sir>cd D:\fspmar5pm
C:\Users\Satish Gupta Sir>d:
D:\fspmar5pm>python test23.py 10 20
1020

D:\fspmar5pm>python test23.py 10 20
30

D:\fspmar5pm>
```

Example:

```
# Write a program to add two complex numbers
# input complex number from command prompt as command line
arguments
```

```
import sys

c1=complex(sys.argv[1])
c2=complex(sys.argv[2])
c3=c1+c2
print(c1,c2,c3)
```



The screenshot shows a Windows Command Prompt window. The command line shows the user navigating to D:\fspmar5pm, running python test24.py, and then entering the complex numbers 1+2j and 1+3j. The output is (1+2j) (1+3j) (2+5j). A second run shows the numbers 1 and 2, resulting in (1+0j) (2+0j) (3+0j).

```
D:\fspmar5pm>python test24.py 1+2j 1+3j
(1+2j) (1+3j) (2+5j)

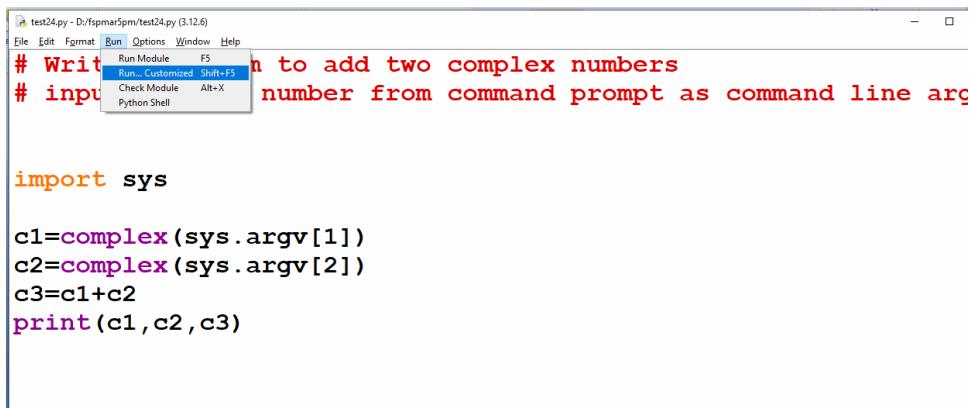
D:\fspmar5pm>python test24.py 1 2
(1+0j) (2+0j) (3+0j)

D:\fspmar5pm>
```

How to give command line arguments from IDLE (OR) how to test program with command line argument using IDLE?

⇒ Run

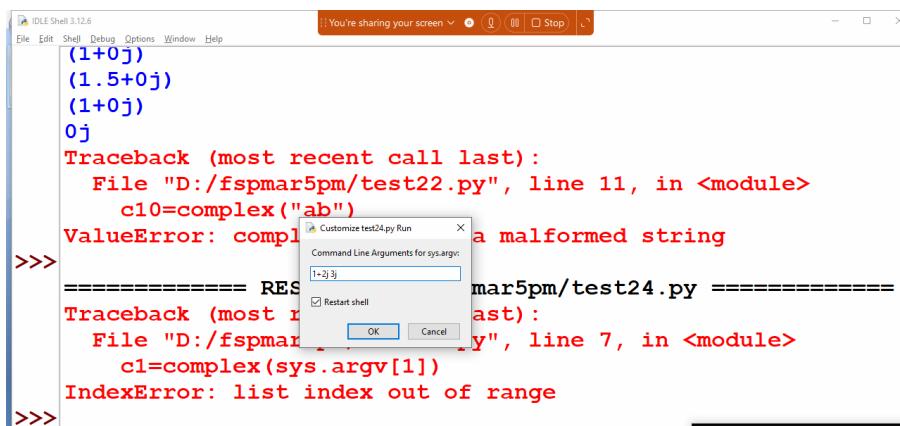
- Run Customized



```
# Write a program to add two complex numbers
# input number from command prompt as command line arguments

import sys

c1=complex(sys.argv[1])
c2=complex(sys.argv[2])
c3=c1+c2
print(c1,c2,c3)
```



```
(1+0j)
(1.5+0j)
(1+0j)
0j
Traceback (most recent call last):
  File "D:/fspmar5pm/test24.py", line 11, in <module>
    c10=complex("ab")
ValueError: complex() arg 1 must be a string or a tuple of two strings, not float
>>> ===== RESTART: D:/fspmar5pm/test24.py =====
Traceback (most recent call last):
  File "D:/fspmar5pm/test24.py", line 7, in <module>
    c1=complex(sys.argv[1])
IndexError: list index out of range
```

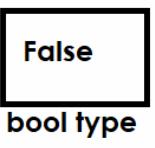
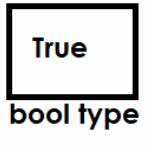
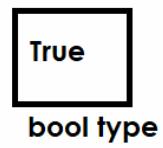
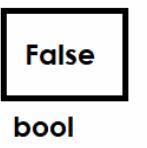
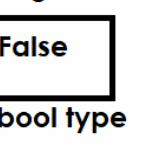
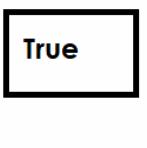
bool()

It is a predefined function in python.

This function is used to perform the following conversions

1. Bool to bool
2. Int to bool
3. Float to bool
4. String to bool

Syntax: <variable-name>=bool(value)

$a = \text{bool}()$	$b = \text{bool}(\text{True})$	$c = \text{bool}(1)$	$d = \text{bool}(\text{False})$	$e = \text{bool}(0)$	$f = \text{bool}(500)$
 a False bool type	 b True bool type	 c True bool type	 d False bool	 e False bool type	 f True bool type
$g = \text{bool}(1.5)$ g	$h = \text{bool}(0.0)$ h	$i = \text{bool}(1+2j)$ i	$j = \text{bool}("A")$ j	$k = \text{bool}("")$ k	$l = \text{bool}("False")$ l