

**Ajeet K. Jain, M. Narsimlu**  
(ML TEAM)- SONET, KMIT, Hyderabad

This session deals with

- Python Tokens(Building blocks)

- Single Line and Multi Line comments

- Examples on each Building block

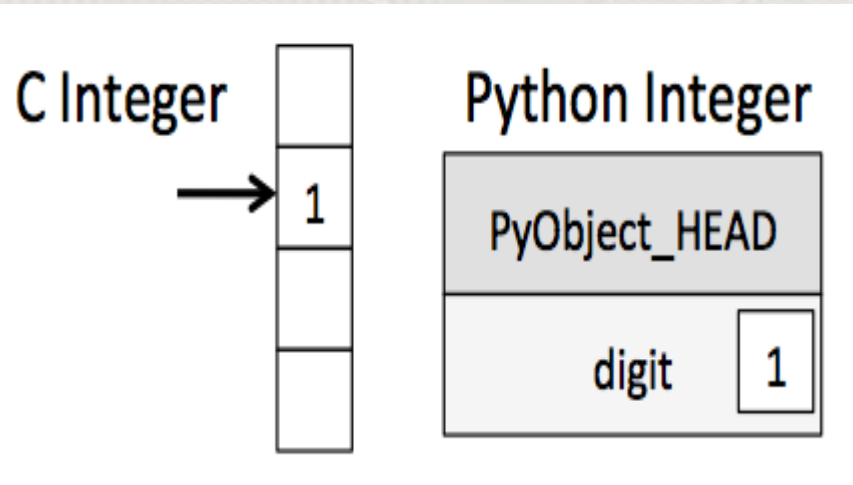
- Exercises

# Memory allocation for data types

PyObject\_HEAD is the part of the structure containing the reference count, type code, and other pieces

A C integer is essentially a label for a position in memory whose bytes encode an integer value

A Python integer is a pointer to a position in memory containing all the Python object information, including the bytes that contain the integer value.



This extra information in the Python integer structure is what allows Python to be coded so freely and dynamically.



# Memory allocation for data types



## Output

```
import sys
i1=8
i2=12996
i3=68768666868
i4=9078087087087877878787
print(sys.getsizeof(i1))
print(sys.getsizeof(i2))
print(sys.getsizeof(i3))
print(sys.getsizeof(i4))
```

IPython console

Console 2/A

```
In [4]: runfile('E:/KMIT/SONET/
NPTEL_Python_DS/Exercises/
py_variable.py', wdir='E:/KMIT/SONET/
NPTEL_Python_DS/Exercises')
28
28
32
36
```

## Memory allocation for Float data types

It does not return the size of any individual float, it returns the size of the float *class*.

Float class contains a lot more data than just any single float, so the returned size will also be much bigger

Every float object will contain a reference counter and a pointer to the type (a pointer to the float class)

Size of float is fixed in python i.e 24 bits

Example:

```
import sys
```

```
F2=8.8
```

```
F3=65.45
```

```
f1=808080807979999.99808080
```

```
print(sys.getsizeof(f1))
```

```
print(sys.getsizeof(f2))
```

```
print(sys.getsizeof(f3))
```

Output

24

# Memory allocation for data types

The size of empty string is 49 bits in spyder

The size of string object is varying based on the data which string object has

**Ex:**

```
import sys
s1=""
s2="py"
s3="Sci"
s4="python"
print(sys.getsizeof(s1))
print(sys.getsizeof(s2))
print(sys.getsizeof(s3))
print(sys.getsizeof(s4))
```

**Output**

```
49
51
52
55
```

Format specifier for int is %d  
Format specifier for float is %f  
Format specifier for string is %s

### Output

```
p1_sal=56000  
p2_sal=86550  
print("%d+%d=%d"%(p1_sal,p2_sal,p1_sal+p2_sal))
```

56000+86550=142550





## Output

```
phy_marks=76.5
maths_marks=83.65
chem_marks=65.34
tot_marks=phy_marks+maths_marks+chem_marks
print("%f+%f+%f=%f"%(phy_marks,maths_marks,chem_marks,tot_marks))
```

Name	Type	Size	Value
chem_marks	float	1	65.34
maths_marks	float	1	83.65
phy_marks	float	1	76.5
tot_marks	float	1	225.49

IPython console

Console 6/A

IPython 7.2.0 -- An enhanced Interactive Python.

```
In [1]: runfile('E:/KMIT/SONET/NPTel_Python/Exercises/py_variable.py', wdir='E:/KMIT/SONET/NPTel_Python/Exercises')
```

76.500000+83.650000+65.340000=225.490000



## Exercise-3

- 1.  $A=4$
- $B=6$
- Perform addition operation and display Format the output as “ $3+4=7$ ” using format specifier
- 2.  $f1=7.8$
- $f2=6.5$
- Perform addition operation and display the result as “ $7.80+6.50=14.30$ ”

# Conclusion

You are aware of  
Python Building blocks  
Why Python and Python Trend

We will proceed with  
Python Operators



**THANK  
YOU**