# Operators in Python





As any programming language Python supports all types of Operations. There are several categories of operators. For now we will cover Arithmetic and Comparison Operators.

- Arithmetic Operators
  - Addition (+)
  - Subtraction (-)
  - Multiplication (*)
  - Division (/)
  - Mod (%) returns reminder
  - + is also used for concatenation of strings.
- Comparison Operators - typically return boolean value (True or False)
  - Equals (==)
  - Not Equals (!=)
  - Negation (! before expression)
  - Greater Than (>)
  - Less Than (<)
  - Greater Than or Equals To (>=)
  - Less Than or Equals To (<=)

# Tasks or Exercises

Let us perform some tasks to understand more about Python Data Types.

- Create variables or objects of int, float.
- Create 2 variables i1 and i2 with values 10 and 20 respectively. Add the variables and assign the result to res_i. Check the type of res_i.

```
i1 = 10
i2 = 20
```

```
res_i = i1 + i2
```

```
print(res_i)
```

```
30
```

```
type(res_i)
```

```
int
```

- Create 2 variables f1 and f2 with values 10.5 and 15.6 respectively. Add the variables and assign the result to res_f. Check the type of f1, f2 and res_f.

```
f1 = 10.5
f2 = 15.6
res_f = f1 + f2
```

```
print(res_f)
```

```
26.1
```

```
type(f1)
```

```
float
```

```
type(res_f)
```

```
float
```

- Create 2 variables v1 and v2 with values 4 and 10.0 respectively. Add the variables and assign the result to res_v. Check the type of v1, v2 and res_v.

```
v1 = 4
v2 = 10.0
res_v = v1 + v2
```

```
print(res_v)
```

```
14.0
```

```
type(res_v)
```

```
float
```

```
# question from the class
f1 = 10.1
f2 = '20.2'
res_f = f1 + float(f2)
# throws operand related to error as there is no overloaded function +
# between float and string
print(res_f)
```

```
30.299999999999997
```

- Create object or variable s of type string for value Hello World and print on the screen. Check the type of s.

```
s = "Hello 'World'"
print(s)
```

```
Hello 'World'
```

- Create 2 string objects s1 and s2 with values Hello and World respectively and concatenate with space between them.

```
s1 = 'Hello'
s2 = 'World'
print(s1 + ' ' + s2)
```

```
Hello World
```

```
s = '{s3} {s4}'
s3 = 'Hello'
s4 = 1
print(s.format(s3=s3, s4=s4))
```

```
Hello 1
```

```
print('The result is {} {}'.format(s3,s4))
```

```
The result is Hello 1
```

- Compare whether i1 and i2 are equal and assign it to a variable res_e, then check the type of it.

```
i1 = 10
i2 = 20
```

```
res_e = i1 < i2
# Feel free to try other operations
```

```
print(res_e)
```

```
True
```

```
type(res_e)
```

```
bool
```

---

By Durga Gadiraju