```
In [ ]: # TYPES OF OPERATOR
        1. Arithmatic Operator
        2. Assignment Operator
        3. Relational Operator
        4. logical Operator
        5. Unary Operator
```

Arithmatic Operator

```
In [1]: P1,Q1= 20,5
        P1+Q1

Out[1]: 25

In [2]: P1-Q1

Out[2]: 15

In [3]: P1*Q1

Out[3]: 100

In [4]: P1/Q1

Out[4]: 4.0

In [5]: P1//Q1

Out[5]: 4

In [6]: P1%Q1

Out[6]: 0

In [7]: P1**Q1

Out[7]: 3200000

In [ ]:
```

Assignment Operator

```
In [23]: x=4
         x

Out[23]: 4

In [24]: x = x + 4
         x

Out[24]: 8
```

```
In [28]: x=x-4
         x
```

```
Out[28]: 60
```

```
In [29]: x*4
```

```
Out[29]: 240
```

```
In [30]: x/4
```

```
Out[30]: 15.0
```

```
In [31]: x//4
```

```
Out[31]: 15
```

Relational operator

```
In [42]: a=7
         b=8
         print(a)
         print(b)
```

```
7
8
```

```
In [40]: a>b
```

```
Out[40]: False
```

```
In [41]: a<b
```

```
Out[41]: True
```

```
In [43]: a!=b
```

```
Out[43]: True
```

```
In [ ]: b=7
        b=8
```

```
In [44]: a==b
```

```
Out[44]: False
```

```
In [45]: a
```

```
Out[45]: 7
```

```
In [46]: b
```

```
Out[46]: 8
```

```
In [47]: a<=b

Out[47]: True

In [48]: a>=b

Out[48]: False
```

Logical Operator

```
In [59]: x=5
         y= 4

In [49]: x<8 and b<5

Out[49]: False

In [60]: x<8 and b<2

Out[60]: False

In [62]: a<8 or b>2

Out[62]: True

In [63]: a>8 or b<2

Out[63]: False

In [ ]: not x # you can reserve the operation

In [66]: x= not x
         x

Out[66]: True

In [71]: x=False
         x

Out[71]: False
```

COMPLEMENT Operator(~)

~12 # WHY WE GET -13 , FIRST WE UNDERSTAND WHAT IS COMPELMENT MEANS (RESERVE WITH BINARY FORMATE

```
In [72]: ~20

Out[72]: -21

In [73]: ~32
```

```
Out[73]:  -33

In [75]:  ~68

Out[75]:  -69
```

Binary Number System

```
In [77]:  26

Out[77]:  26

In [78]:  bin(26)

Out[78]:  '0b11010'

In [80]:  24

Out[80]:  24

In [81]:  bin(24)

Out[81]:  '0b11000'

In [83]:  15

Out[83]:  15

In [84]:  bin(15)

Out[84]:  '0b1111'

In [85]:  30

Out[85]:  30

In [86]:  bin(30)

Out[86]:  '0b11110'

In [87]:  # Example: Octal numbers
          num1 = 0o10     # Octal 10
          num2 = 0o25     # Octal 25
          num3 = 0o77     # Octal 77

          print(num1)  # Output: 8   (decimal)
          print(num2)  # Output: 21  (decimal)
          print(num3)  # Output: 63  (decimal)

        8
        21
        63

In [ ]:
```