



**& ==>** If both bits are 1 then only result is 1 otherwise result is 0  
**| ==>** If atleast one bit is 1 then result is 1 otherwise result is 0  
**^ ==>** If bits are different then only result is 1 otherwise result is 0  
**~ ==>** bitwise complement operator  
           1==>0 & 0==>1  
**<< ==>** Bitwise Left shift  
**>> ==>** Bitwise Right Shift

```

print(4&5) ==>4
print(4|5) ==>5
print(4^5) ==>1
  
```

Operator	Description
&	If both bits are 1 then only result is 1 otherwise result is 0
	If atleast one bit is 1 then result is 1 otherwise result is 0
^	If bits are different then only result is 1 otherwise result is 0
~	bitwise complement operator i.e 1 means 0 and 0 means 1
>>	Bitwise Left shift Operator
<<	Bitwise Right shift Operator

### bitwise complement operator(~):

We have to apply complement for total bits.

Eg: print(~5) ==>-6

### Note:

The most significant bit acts as sign bit. 0 value represents +ve number where as 1 represents -ve value.

positive numbers will be represented directly in the memory where as -ve numbers will be represented indirectly in 2's complement form.

## Shift Operators:

### << Left shift operator

After shifting the empty cells we have to fill with zero

```
print(10<<2)==>40
```

