Vidya L S 4AL16EC093 21-05-2020

## Report on python

## Operators:

- > Operators are used to perform operations on variables and values.
- > Python divides the operators in the following groups
- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Identity operators
- Membership operators
- Bitwise operators
- ➤ Arithmetic operators are used with numeric values to perform common mathematical operations. Operators are +, -, /, %, \*\* is a exponentiation, // is a symbol of floor division.

Assignment operators are used to assign values to variables. operators are  $=,+=,-=,*=,/=,\%=,//=,**=,\&=,|=,^=,>>=,<<=.$ 

- > Comparison operators are used to compare two values. Operators are ==, !=,>,<,>=,<=.
- ➤ Logical operators are used to combine conditional statements. Operators are and, or, not.
- ➤ Identity operators are used to compare the objects, not if they are equal, but if they are actually the same object, with the same memory location. Operators are' is' it returns True if both variables are same object, 'isnot' it returns True if both variables are not same object.
- Membership operators are used to test if a sequence is presented in an object. Operators are 'in' it returns True if a sequence with the specified value is present in the object, 'not in' it returns True if a sequence with the specified value is not present in the object,.
- ➤ Bitwise operators are used to compare (binary) numbers. Operators are & it sets each bit to 1, if both bits are 1, | it sets each bit to 1, if one of two bits is , ^ it sets each bit to 1 if only one of two bits is .
- > ~ it inverts all the bits, << it shifts left by pushing zeros in from the right and let the leftmost bits fall off, >> it shifts right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off.

```
Eg: 1. x = ["apple", "banana"] Output: True z = x True print("pineapple" not in x) False print(x is z) True print(x is y) print(x == y)

2. x = 5 Output: True print(x > 3 \text{ or } x < 4)
```

print(x != y)