

## Assignment-2

1. True or false, Boolean operators are written with respect to comparison using AND, OR and NOT operations.

2. AND, OR and NOT

3. **AND**

TRUE+TRUE=TRUE  
TRUE+FALSE=FALSE  
FALSE+FALSE=FALSE  
FALSE+TRUE=FALSE

**OR**

TRUE+TRUE=TRUE  
TRUE+FALSE=TRUE  
FALSE+FALSE=FALSE  
FALSE+TRUE=TRUE

**NOT**

TRUE→FALSE  
FALSE→TRUE

4. (5>4) and (3==5)→False  
not(5>4)→ False  
(5>4) or(3==5)→True  
not(5>4) or(3==5)→False  
(True and True) and (True==False)→ False  
(not false) or (not true)→True

5. =,>,<,>=,<=,!=

6. If we want to give a value to any variable we use equals i.e a=10, b=15 etc  
If we want to do any comparison we use == operator→ i.e a="sagar"  
If type(a)==str:  
Print true  
Here sagar is assigned to a, and the word sagar is compared to string or not.

7. **First Block**

```
if spam == 10:  
    print('eggs')
```

**Second Block**

```
if spam>5  
    print('bacon')
```

**Last Block**

```
Print('ham')  
Print('ham')  
Print('ham')
```

Out put will be ham, spam, spam every time for this question since spam=0.

8.

```
spam=int(input("Enter value of spam"))
if spam==1:
    print('hello')
elif spam==2:
    print('Howdy')
else:
    print('Greetings')
```

9. Ctrl+C which will interrupt the keyboard input. Or restart the kernel.

10. `l=[1,2,3,4,5,6,7,8,9]`

```
for i in l:
    if i==4:
        break:
```

Here break statement is used to terminate the loop after 4.

```
l=[1,2,3,4,5,6,7,8,9]
for i in l:
    if i==4:
        continue:
```

Here continue statement is used to iterate the loop after 4.

Break will break the loop after condition.  
Continue will iterate the loop after condition.

11. All the 3 yields same result , i.e all the values between 0 to 10 excluding 10.

12. **For Loop**

```
for i in range(1,11):
    print(i)
```

**While Loop**

```
i=1
while i<=10:
    print(i)
    i=i+1
```

13.

```
✓ [48] def bacon():
0s      print("spam")
```

```
✓ [49] bacon()
```

spam