# Assignment 2:

# Submitted by Dr. Abhishek Hukkerikar

**Note: The answers are in Blue color**

1.What are the two values of the Boolean data type? How do you write them?

Ans 1: The two values are True, False. True can be written as == and False can be written as != (Not equal to)

2. What are the three different types of Boolean operators?

Ans2: Three different types of Boolean operators are AND, OR, NOT

3. Make a list of each Boolean operator's truth tables (i.e. every possible combination of Boolean values for the operator and what it evaluate

| AND Boolean operation / Also called as AND GATE | | |
|---|---|---|
| Input 1 | Input 2 | Result |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |
| 0 | 0 | 0 |

In the AND operation both the inputs have to be 1 for the result to be 1

| OR Boolean operation / Also called the OR gate | | |
|---|---|---|
| Input 1 | Input 2 | Result |
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

In the OR operation any one of the input has to be 1 for the output to be 1

| NOT Boolean operation / Also known as NOT gate | |
| --- | --- |
| Input 1 | Result |
| 0 | 1 |
| 1 | 0 |

Basically, the not operation will give a reciprocal of the input. Note it has only 1 input.

4. What are the values of the following expressions?

- (5 > 4) and (3 == 5)

  0 / False

- not (5 > 4)

  0 / False

- (5 > 4) or (3 == 5)

  1 / True

- not ((5 > 4) or (3 == 5))

  0/ False

- (True and True) and (True == False)

  0 / False

- (not False) or (not True)

  1 / True

5. What are the six comparison operators?

The six comparison operators in python are, less than (<), less than or equal to (<=), greater than (>), greater than or equal to (>=), equal to (=), not equal to (!=)

6. How do you tell the difference between the equal to and assignment operators? Describe a condition and when you would use one.

The assignment operator is =  and the equating operator is == . It could be understood by the code as below:

x = 10
y = 10
if x == y:
        print(' The values of both the variables are equal')

else

        print(' The values are not equal).

In the code above, we have used (=) to assign the value of 10 to variables x and y. While, we use (==) to compare if the values of x and y are equal or not.

7. Identify the three blocks in this code:

spam = 0

if spam == 10:

        print('eggs')

if spam > 5:

        print('bacon')

else:

        print('ham')

        print('spam')

I was not able to understand the question. The code appears to be fine. If the value of spam is changed from 0 to any other, the program executes accordingly.

8. Write code that prints Hello if 1 is stored in spam, prints Howdy if 2 is stored in spam, and prints Greetings! if anything else is stored in spam.

```
i = int(input('Enter the value of spam'))
if i == 1:
        print('Hello')
if i == 2:
        print('Howdy')
else:
        print('Greetings')
```

9.If your programme is stuck in an endless loop, what keys you'll press?

Escape

10. How can you tell the difference between break and continue?

The break and continue are considered to be jump statements. The break is used for immediate termination of loop. The continue statement is used to terminate the current loop and resume the control program to the next loop.

11. In a for loop, what is the difference between range(10), range(0, 10), and range(0, 10, 1)?

The loop in python is of the syntax range(start, stop, step). Thus, range(10) indicates the limit of an iteration to 10.

Range (0,10) indicates to begin the iteration from 0 and stop it at 10. Importantly, 10 is exclusive meaning, it will not be included. The same is the case in indexing also. The beginning number 0 is inclusive.

12. Write a short program that prints the numbers 1 to 10 using a for loop. Then write an equivalent program that prints the numbers 1 to 10 using a while loop.

for i in range(1,11):

      print (i, end= ' ')

--

I = 1

while(i<=10):

      print(i)

      I = I +1

13. If you had a function named bacon() inside a module named spam, how would you call it after importing spam?

Spam.bacon()