

Assignmet-2

1.What are the two values of the Boolean data type? How do you write them?
Ans- A variable of the primitive data type Boolean can have two values: true and false.

```
In [1]: #Eg:
a=True
b=False
print(a)
print(b)

True
False

In [ ]:
```

2. What are the three different types of Boolean operators?
Ans- Three different types of Boolean Operator is "AND","OR" ,"NOT" .

```
In [ ]:
```

Q3. 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).
Ans-True and True is True. True and False is False. False and True is False. False and False is False. True or True is True. True or False is True. False or True is True. False or False is False. not True is False, not False is True.

```
In [ ]:
```

True is 1 and False is 0

Truth Table for AND A B output 0 0 0 1 0 1 0 0 1 1 1 Truth Table for OR A B output 0 0 0 1 1 1 0 1 1 1 1 Truth Table for NOT A output 0 1 1 0

```
In [ ]:
```

4. What are the values of the following expressions?

```
In [2]: print((5>4) and (3==5))
print(not(5>4))
print((5>4) or (3==5))
print(not(5>4) or (3==5))
print((True and True) and (True==False))
print((not(False))or(not(True)))

False
False
True
False
False
True

In [ ]:
```

Q5. What are the six comparison operators?
Ans . ==, !=, <, >, <=, and >=

```
In [ ]:
```

Q6. How do you tell the difference between the equal to and assignment operators?Describe a condition and when you would use one.

```
In [3]: #eg.
# Equal To Operator
if(2==3):
    print("True")
else:
    print("False")
#Assignment operator
c=1 #here we have used assignment operator(=) to assign value of c which is 1
print("c =",c)

False
c = 1

In [ ]:
```

Q7. Identify the three blocks in this code:

```
In [4]: spam = 0
if spam == 10:
    print('eggs') #Block 1
if spam > 5:
    print('bacon') #Block 2
else:
    print('ham') #Block 3
    print('spam')
    print('spam')

ham
spam
spam

In [ ]:
```

Q8. 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.

```
In [5]: spam = int(input("Input a no. "))
if spam == 1:
    print("Hello")
elif spam == 2:
    print("Howdy")
else:
    print("Greetings!")

Input a no.1
Hello

In [ ]:
```

Q9.If your programme is stuck in an endless loop, what keys you'll press?
Ans . If program is stuck in endless loop we will press ctrl+c

```
In [ ]:
```

Q10. How can you tell the difference between break and continue?

```
In [6]: # use of break
for i in range(10):
    if(i==7):
        break
    print(i)

print('Breaked')
#use of continue
for i in range(10):
    if(i==7):
        continue
    print(i)

0
1
2
3
4
5
6
Breaked
0
1
2
3
4
5
6
8
9

In [ ]:
```

Ans . The break statement will move the execution outside and just after a loop. The continue statement will move the execution to the start of the loop.

```
In [ ]:
```

Q11. In a for loop, what is the difference between range(10), range(0, 10), and range(0, 10, 1)?

```
In [7]: for i in range(10):
print(i)
print("xxxxxxxxxxxxxxxxxxxxxxxxxxxx")
for i in range(0,10):
    print(i)
print("xxxxxxxxxxxxxxxxxxxxxxxxxxxx")
for i in range(0,10,1):
    print(i)

0
1
2
3
4
5
6
7
8
9
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
0
1
2
3
4
5
6
7
8
9
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
0
1
2
3
4
5
6
7
8
9

In [ ]:
```

From the above output we can conclude that they all do the same thing. The range(10) call ranges from 0 up to (but not including) 10, range(0, 10) tells the loop to start at 0, and range(0, 10, 1) tells the loop to increase the variable by 1 on each iteration.

```
In [ ]:
```

Q12. 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.

```
In [8]: #Use of For Loop
print("For Loop")
for i in range(1,11):
    print(i)
#Use of While Loop
print("While Loop")
a =1
while a <= 10:
    print(a)
    a+=1

For Loop
1
2
3
4
5
6
7
8
9
10
While Loop
1
2
3
4
5
6
7
8
9
10

In [ ]:
```

Q13. If you had a function named bacon() inside a module named spam, how would you call it after importing spam?
Ans- This function can be called with spam.bacon().

```
In [ ]:
```