

PYTHON ASSIGNMENT 2

Q1. What are the two values of the Boolean data type? How do you write them?

Ans. A variable of primitive data type Boolean can have two values: True and false (Boolean literals) which are special versions of 1 and 0 respectively and behave as such in arithmetic contexts.

Q2. What are the three different types of Boolean operators?

Ans. The three basic Boolean operators are: AND, OR and NOT

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 evaluates to).

Ans. List of each boolean operators 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.

TRUE IS 1 AND FALSE IS 0 If A and B Then C:

Truth table:

A B C

1 1 1

1 0 0

0 1 0

0 0 0

If: A or B Then: C

Truth table:

A B C

1 1 1

1 0 1

0 1 1

0 0 0

If NOT A

Truth table:

A C

0 1

1 0

Q4. 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
```

Q5. What are the six comparison operators?

Ans. The six comparison operators are: ==, !=, <, >, <=, and >=

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

Ans. = is an Assignment Operator it is used to assign the value of variable or expression, while == is an Equal to Operator and it is a relation operator used for comparison

```
In [3]: # example

# EQUAL TO OPERATOR

a=10

print(a)

# ASSIGNMENT OPERATOR

if (15==12):
    print('true')
else:
    print('false')
```

```
10
false
```

Q7. Identify the three blocks in this code:

```
In [ ]: 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')
```

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 [4]: spam= eval(input('Enter a number: '))
        if spam == 1:
            print('Hello')
        elif spam == 2:
            print('Howdy')
        else:
            print('Greetings!')
```

Enter a number: 32
Greetings!

Q9. If your programme is stuck in an endless loop, what keys you'll press?

Ans. If our program is stuck in an endless loop we can stop it with CTRL+C

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

Ans. Break statement stops the entire process of the loop. Continue statement only stops the current iteration of the loop. Break also terminates the remaining iterations. Continue doesn't terminate the next iterations; it resumes with the successive iterations.

```
In [5]: # use of break
        for a in range(10):
            if(a==7):
                break
            print(a)
```

0
1
2
3
4
5
6

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

0
1
2
3
4
5
6
8
9

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

```
In [7]: # range(10)

        for i in range(10):
            print(i)
```

0
1
2
3
4
5
6
7
8
9

```
In [8]: # range(0,10)

for i in range(0,10):
    print(i)
```

0
1
2
3
4
5
6
7
8
9

```
In [9]: # range(0,10,1)

for i in range(0,10,1):
    print(i)
```

0
1
2
3
4
5
6
7
8
9

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 [10]: # FOR LOOP

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

1
2
3
4
5
6
7
8
9
10

In [11]: `# WHILE LOOP`

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

1
2
3
4
5
6
7
8
9
10

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()`.