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

True and False, where True represents 1 and False represents 0

Ex.

|  |  |
| --- | --- |
| **Code** | **Output** |
| a = 10  b = (a == 10)  print(b) | True |

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

The logical operators **and**, **or** and **not** are also referred to as Boolean operators.

**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 evaluates).**

|  |  |  |
| --- | --- | --- |
| **Input** | | **Output** |
| **A** | **B** | **Y = A and B** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

|  |  |  |
| --- | --- | --- |
| **Input** | | **Output** |
| **A** | **B** | **Y = A or B** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| **A** | **Y = not A** |
| 0 | 1 |
| 1 | 0 |

**4. What are the values of the following expressions?**

(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. What are the six comparison operators?**

Python has *six* comparison operators: less than ( < ), less than or equal to ( <= ), greater than ( > ), greater than or equal to ( >= ), equal to ( == ), and 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.**

Equal to (==) is a relational operator used to check equality of the operands whereas assignment operator (=) is used to assign value to a variable

Ex.

|  |  |
| --- | --- |
| **Code** | **Output** |
| a = 10  if a == 10:  print (“Inside if block”)  else:  print (“Inside else block”)  print (“Program execution ends” | Inside if block  Program execution ends |

**7. Identify the three blocks in this code:**

spam = 0

Block 1

if spam == 10:

print('eggs')

if spam > 5:

Block 2

print('bacon')

else:

print('ham')

Block 3

print('spam')

print('spam')

**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.**

|  |  |
| --- | --- |
| **Code** | **Output** |
| spam = 3  if spam == 1:  print ("Hello")  elif spam ==2:  print("Howdy")  else:  print("Greetings!") | Greetings |

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

CTRL + C (KeyboardInterrupt)

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

The main difference between break and continue statements is that when break keyword comes, it terminates the execution of the current loop and passes the control over the next loop or main body, whereas when continue keyword is encountered, it skips the current iteration and executes the very next iteration in the loop.

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

The syntax for range in a for loop is – **range (start, end, step increment)**. Of the three parameters, end is mandatory.

range(10) – By default it assumes the start as 0 and step increment as 1. So the loop executes for 10 times for the range of numbers through 0 to 9

range(0,10) – Since, the step increment default value is 1, so the loop executes for 10 times for the range of numbers through 0 to 9

range(0,10,1) - Here all the parameter are specified and hence no default value is taken. The loop executes for 10 times for the range of numbers through 0 to 9

**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 loop** | **while loop** |
| for i in range(1,11):  print(i) | i = 0  while i<=10:  i = i + 1  print(i) |

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

spam.bacon()