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

Ans. Two values of Boolean data type is True and false. We write true as 1 and False as 0

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

Ans. Three different types of Boolean operators are AND, OR, and 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 ).**

|  |  |  |  |
| --- | --- | --- | --- |
| INPUT | AND | OR | NOT |
| 1&1 | 1 | 1 | 0 |
| 1&0 | 0 | 1 | 1 |
| 0&1 | 0 | 1 | 1 |

Ans.

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

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

**not (5 > 4)**

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

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

**(True and True) and (True == False)**

**(not False) or (not True)**

**5. What are the six comparison operators?**

Ans. (i) Equality operator (==): It checks if two values are equal and returns true if they are, or false if they are not. For example, 5 == 5 evaluates to true.

(ii)Inequality operator (!=): It checks if two values are not equal and returns true if they are not equal, or false if they are equal. For example, 5 != 3 evaluates to true.

(iii)Less than operator (<): It checks if the value on the left is less than the value on the right and returns true if it is, or false otherwise. For example, 3 < 5 evaluates to true.

(iv)Greater than operator (>): It checks if the value on the left is greater than the value on the right and returns true if it is, or false otherwise. For example, 5 > 3 evaluates to true

(v)Less than or equal to operator (<=): It checks if the value on the left is less than or equal to the value on the right and returns true if it is, or false otherwise. For example, 3 <= 5 evaluates to true.

(vi)Greater than or equal to operator (>=): It checks if the value on the left is greater than or equal to the value on the right and returns true if it is, or false otherwise. For example, 5 >= 5 evaluates to true.

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

Ans. Equal to operator (‘==’) used to comparison, comparing the equality between the two values it checks the values to the both sides of the operator and gives the output in the Boolean form either true or false.

Assignment operator (‘=’) used to assign a value to a variable. It assigns the value on the right side of the operator.

**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')**

**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.**

Ans. spam=int (input (enter a value”)

if spam==1:

print=(“Hello”)

elif spam==2:

print(“Howdy”)

else:

print(“Greetings!”)

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

**Ctrl+C**: This is the most common way to interrupt a running program in the command line. Pressing Ctrl+C sends an interrupt signal (SIGINT) to the program, causing it to terminate. This should break the loop and stop the program from running.

**Ctrl+Break**: On some systems, such as Windows, you may use Ctrl+Break instead of Ctrl+C to interrupt the program. This key combination sends a break signal, similar to SIGINT, causing the program to stop.

**Ctrl+D**: On Unix-based systems, including Linux and macOS, pressing Ctrl+D sends an end-of-file (EOF) signal to the program's standard input. This can be used to terminate the program gracefully, but it may not work in all cases, especially if the program is stuck in an infinite loop.

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

Ans. **Break**:- When the "break" statement is encountered, the loop is immediately terminated, and the program continues to execute from the next statement after the loop. It is often used to prematurely exit a loop based on certain conditions .

**Continue:-**  When continue statement encountered, the loop jumps to the next iteration, ignoring any code below the "continue" statement within the current iteration.

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

Ans. **range(10):**This syntax specifies that the loop should iterate from 0 to 9 (10 is excluded).The starting point is implicitly assumed as 0, and the step value is implicitly assumed as 1.

**range(0, 10):**This syntax explicitly specifies the start and stop values for the loop. The loop will iterate from the start value (0) to the stop value (10 is excluded).The step value is implicitly assumed as 1.

**range(0, 10, 1):**This syntax explicitly specifies the start, stop, and step values for the loop. The loop will iterate from the start value (0) to the stop value (10 is excluded) in increments of the step value (1).In this case, the step value is explicitly specified as 1, but since it is the default, it can be omitted

**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.**

ans. for i in range(1,11):

print(i)

**and**

nm=1

while nm<=10:

print(nm)

nm+=1

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

Ans. import spam

spam.bacon()