Assignment 2 - Solutions

1. The two values of the Boolean data type are "true" and "false", which represent logical values. In programming languages, these values are typically written as keywords, such as "true" or "false" in most programming languages like Python, Java, C++, JavaScript, etc. In some languages, the Boolean values can also be represented numerically, where "true" is equivalent to 1 and "false" is equivalent to 0.

2. The three different types of Boolean operators are:

   AND operator: It is represented by the symbol "&&" or the word "AND". It returns true if both operands are true and false otherwise.

   OR operator: It is represented by the symbol "||" or the word "OR". It returns true if at least one of the operands is true and false otherwise.

   NOT operator: It is represented by the symbol "!" or the word "NOT". It returns the opposite of the operand's value. For example, if the operand is true, the NOT operator returns false, and vice versa.

3. **AND operator:**

A	B	A && B
false	false	false
false	true	false
true	false	false
true	true	true

   OR operator:

A	B	A || B
false	false	false
false	true	true
true	false	true
true	true	true

   NOT operator:

A	!A
false	true
true	false

4.
   a. (5 > 4) and (3 == 5) - False
   b. not (5 > 4) - False
   c. (5 > 4) or (3 == 5) - True

d. not ((5 > 4) or (3 == 5)) - False
e. (True and True) and (True == False) - False
f. (not False) or (not True) - True

5. The six comparison operators in Python are:

   > (greater than)
   < (less than)
   >= (greater than or equal to)
   <= (less than or equal to)
   == (equal to)
   != (not equal to)

   These operators are used to compare two values and return a Boolean value (True or False) based on whether the comparison is true or false.

6. In Python, the equal to operator is ==, and the assignment operator is =. The equal to operator is used to compare two values for equality, while the assignment operator is used to assign a value to a variable.

   To distinguish between the two operators, you can look at the context in which they are used. When the operator is used to compare two values, it is the equal to operator. When the operator is used to assign a value to a variable, it is the assignment operator.

7. If, if, else

8.

```
spam = 3

if spam == 1:
    print("Hello")
elif spam == 2:
    print("Howdy")
else:
    print("Greetings!")
```

9. If your program is stuck in an endless loop, you can press the Ctrl + C keys to stop the execution of the program.

10. break is used to exit a loop early if a certain condition is met. When break is executed, the loop is terminated immediately, and control is passed to the next statement after the loop. For example:

```
for i in range(10):
    if i == 5:
        break
    print(i)
```

continue is used to skip a particular iteration of a loop when a certain condition is met. When continue is executed, the loop immediately moves to the next iteration without executing any statements in the current iteration after the continue statement. For example:

```
for i in range(10):
    if i == 5:
        continue
    print(i)
```

11. In Python, range() is a built-in function that generates a sequence of numbers that can be used to iterate over a for loop. range(10) will generate a sequence of numbers starting from 0 up to (but not including) 10, with a step of 1. This is equivalent to range(0, 10, 1). So, all three of the following code snippets will produce the same output:

12.

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

And here's the equivalent program that prints the numbers 1 to 10 using a while loop:

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

13.

```
import spam
spam.bacon()
```