In [ ]:  1.What are the two values of the Boolean data type? How do you write them?

In [2]:  The two values of the Boolean data type are **True and False.**
Ex:
print(10==10)
print(10==9)

True
False

In [ ]:  2. What are the three different types of Boolean operators?

In [ ]:  The three basic boolean operators are: AND, OR, and NOT.

In [ ]:  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 ).

In [ ]:
```
For OR:          For AND:          For NOT:

                                    1 0
                                    0 1
1 0 1            1 0 0
0 0 0            0 1 0
1 1 1            1 1 1
0 1 1            0 0 0
```

In [ ]:  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)

In [ ]:  (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**

In [ ]: `5. What are the six comparison operators?`

In [ ]:
```
The six comparison operators are:
a.Equal                    x == y
b.Not equal                x != y
c.Greater than             x > y
d.Less than                x < y
e.Greater than or equal to x >= y
f.Less than or equal to    x <= y
```

In [ ]: `6.How do you tell the difference between the equal to and assignment operators?Describe a condition and when you would use one.`

In [5]:
```
equal to (==) is used to compare values and assignment (==) is used to assign value.
ex:
print(5==6)
ex:
a=5
print(a)
```

```
False
5
```

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

In [ ]:
```
The three blocks are everything inside the if statement and the lines print('bacon') and print('ham').

print('eggs')
if spam > 5:
    print('bacon')
else:
```

```
        print('ham')
print('spam')
```

In [ ]: 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.

In [8]:
```python
spam = int(input("Enter your number: "))
if spam == 1:
    print('Hello')
elif spam == 2:
    print('Howdy')
else:
    print('Greetings!')
```

Enter your number: 2
Howdy

In [ ]: 9.If your programme **is** stuck **in** an endless loop, what keys you'll press?

In [ ]: We have to press CTRL+C.

In [ ]: 10. How can you tell the difference between **break and** continue?

In [ ]: Break statement stops the entire process of the loop. Continue statement only stops the current iteration of the loop.

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

In [17]:
```python
No differences between them all three range functions provide same output.
print(list(range(10)))
print(list(range(0,10)))
print(list(range(0,10,1)))
```

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [ ]: 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.

In [1]:
```python
using for loop:
for i in range(1,11):
```

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

```
1
2
3
4
5
6
7
8
9
10
1
2
3
4
5
6
7
8
9
10
```

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

In [ ]: This function can be called with spam.bacon().