CHAPTER 2

EXERCISES

[1](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1550). What are the two values of the Boolean data type? How do you write them?

* The two values of the Boolean data type are true and false. It can be written as **True** and **False** otherwise python will throw an error message.

[2](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1551). What are the three Boolean operators?

* The three Boolean operators are **and, or and not.**

[3](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1552). Write out the truth tables of each Boolean operator (that is, every possible combination of Boolean values for the operator and what they evaluate to).

AND operator:

|  |  |
| --- | --- |
| Expression | Result |
| True and True | True |
| True and False | False |
| False and True | False |
| False and False | False |

OR operator:

|  |  |
| --- | --- |
| Expression | Result |
| True and True | True |
| True and False | True |
| False and True | True |
| False and False | False |

NOT operator:

|  |  |
| --- | --- |
| Expression | Result |
| not True | False |
| not False | True |

[4](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1553). What do the following expressions evaluate to?

(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](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1554). What are the six comparison operators?

== equal to

!= not equal to

< lesser than

> greater than

<= less than or equal to

>= greater than or equal to

[6](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1555). What is the difference between the equal to operator and the assignment operator?

* + - In assignment operator the value on the right side of the statement gets assigned to the variable on the left side. Eg: num = 10 in this example 10 is assigned to the variable num.
    - In equal to operation the value on the right side is compared to the value on the left side. If both the value matches this relational operation produces an output as True otherwise the output is False

[7](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1556). Explain what a condition is and where you would use one.

* All Boolean expressions that evaluate to any of the Boolean data types are said to be Conditions which can be used as a entry point for a block of code. If evaluated value is True the block of code gets executed, Otherwise Python skips the block of code.

[8](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1557). Identify the three blocks in this code:

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')

[9](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1558). 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.

spam = 2

if spam ==1:

print(‘Hello’)

elif spam ==2:

print(‘Howdy’)

else:

print(‘Greetings!’)

[10](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1559). What keys can you press if your program is stuck in an infinite loop?

Ctrl + C

[11](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1560). What is the difference between break and continue?

* break – when break statement is executed the program exits from the loop.
* continue – when continue statement is executed the current loop is continued from the start.

[12](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1561). What is the difference between range(10), range(0, 10), and range(0, 10, 1) in a for loop?

* + - The range() function can be called with three arguments. The first two arguments will be the start and stop values, and the third will be the step argument. The step is the amount that the variable is increased by after each iteration.
    - We can represent the range() function with either using a stop argument or start and stop argument or start, stop and step argument.
    - All the above ranges are same but it is represented differently with different number of arguments

[13](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1562). 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:

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

print(i)

While Loop:

i = 1

while i<11:

print(i)

i+=1

[14](https://automatetheboringstuff.com/2e/chapter2/#calibre_link-1563). If you had a function named bacon() inside a module named spam, how would you call it after importing spam?

spam.bacon( )