Chapter 3:

By: James Makovics

1. Branching – using ‘if’ statements to evaluate a variable

If x == x:

1. Relational operator – Checks if a variable does or does not match another variable. The variable also can be less than or greater than another variable.

If x < y:

1. Equality operator – checks if a value is equal to another value

If x == y:

1. Boolean – A variable the validates to true or false depending on the input or change in the code

X = true

1. Operator chaining – using multiple operators to compare values without using an ‘and’ or ‘or’ statements

If x < a < b:

1. nesting – Putting multiple ‘if’ statements or loops within other ‘if’ or loop statements

if: x < 6:

if y < x:

1. > - checks if a variable is greater than another variable

If: x > y:

1. < - checks if a variable is less than another variable

If: x < y:

1. >= - checks if a variable is greater than or equal to another variable

If x >= y:

1. <= - checks if a variable is less than or equal to another variable

If x <= y:

1. == - checks if a variable is equal to another variable

If x == a:

C = true

1. and – used in an if statement to match another condition

if x == a and b == c:

1. or – used in an if statement to match one condition or the other

if x == b or x == c:

1. not – used in an if statement to compare if a statement does not match the set value

if x is not b:

1. membership operator – Compares a statement that evaluates to true or false
2. a = 'London'
3. b = 'London'
4. c = 'Paris'
5. if a is not b: print ('a is not b')
6. else: print ('a not b')
7. if a is not c: print ('a is not c')
8. else: print ('a is c')
9. if b is not c: print ('b is not c')
10. else: print ('b is c')
11. condition – a statement that is compared against a value

x\_is\_less\_than\_ten < 10

1. lower() – makes the characters in a string lowercase

example\_statement.upper()

1. upper() – makes the characters in a string uppercase

example\_statement.lower()

1. expression – using arguments to assign values to variables

x = 3 + 4 \* 9

1. else – a statement that is followed when the conditions does not meet if or elif

else:

(Continue code)

1. elif – Another conditional statement that is followed instead of if

elif 10 < x:

1. if – a condition that is followed when a variable matches the condition

if 10 > x:

1. != - compares that variables or numbers to see if their values do not match

if rock != scissors:

print(“rock beats scissors”)