A picture containing graphical user interface

Description automatically generated

5.7: Nesting (for within a for loop)

As we saw last week, nesting involves putting a control structure (or structures) within another control structure. For example, we could nest an if statement within a for loop (maybe to print out the even numbers between 1 and 20). Another possible option is to nest a for loop within another for loop. In this section, we will be looking nesting for loops.

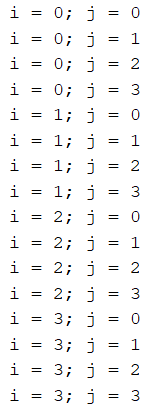
For our first example, we are going to look at how nested for loops produce their outputs. Start by defining a function called nested\_for\_example\_1() which takes no parameters. Enter the following code:

for i in range(4):

for j in range(4):

print("i = {0}; j = {1}".format(i,j))

Adapt your main() function so that you can run the above code. You should get the following output:



You will notice that each time the value for i changes, the j loop runs again. Also, you will see that the value for i doesn’t change until the j loop has finished running. This is all to do with the placement of the j loop. If it wasn’t indented and contained within the i loop, the first loop would run, then the j loop runs. However, as it is included within the i loop, it will run for each iteration of the i loop.