```python
# percipio18_while_loop.py
# Percipio video: Collections, Mapping, & Looping; The While Loop in Python
# The While Loop executes code if its condition is true
nl = '\n'
counter = 3 # sets a variable named counter to 3
while counter > 0: # if this code lines evaluates as true, then indented code will execute
    print('counting down:', counter) #
    counter -=1 # decrement statement where counter = counter minus one

while counter > 0: # this while loop will never execute
    print('Never executes code')
    print('when condition is false')

while 1: # 'while: 1' actually means 'while: true' & this is a common Python idiom for a static True.
    print('Executes at least once') # will always print
    if not counter: # CONFUSING-checks a condition: 1) if counter variable is 0, that's False, 2) 'not' false
        is true, and 3) 'if' true statement will execute code
        break # break causes the while loop to exit whic is important with a static true

# While loops can process lists containing a sequence until that sequence is empty
# A list is considered true when it has populated elements
# A list is considered false when it is empty
names = ['Tom', 'Ellen'] # This list is true with 2 elements
print('names = ', names)
while names: # this line says 'while the names variable is true', execute
    print(names.pop(), 'is going') # pop method will remove last element from the names-variable list
    print('names =', names) # I added this line
    # when names is false, while loop exits

results = [1, 0, 1] #
processed = 0 # variable to count how many results were processed
passed = 0 # variable to count how many results were passed
print('processed =', processed) # I added this line
print('passed =', passed) # I added this line
while results: # a list containing elements is true
    processed += 1 # will increment the processed variable by +1
    result = results.pop() # pop method will remove last element from the results-variable list
    if not result: # 1) '1' is a true value stored in result, 2) 'not' true value (1) is a false value, this
        condition is false & doesn't execute the 'continue' statement
        print('continue statement is run')
        continue # run only if 'if not result' statement is true, 'continue' statement will cause the loop to
            restart from the loop beginning
    passed += 1 # will increment the passed variable by +1
    print('processed =', processed) # I added this line
    print('passed =', passed) # I added this line
else: # will only execute when the 'while:results:' condition is false
    print('Processed:', processed, 'Passed:', passed) #
```

```
43   else: # will only execute when the 'while:results:' condition is false
44       print('Processed:', processed, 'Passed:', passed) #
45   '''
46   counting down: 3
47   counting down: 2
48   counting down: 1
49   Executes at least once
50   Ellen is going
51   Tom is going
52   Processed: 3 Passed: 2
53   '''
```