**Chapter 5 Repetition Structures**

Q1: simple code with while functions. What is the different logic for Part 1 and Part 2? Why?

while temperature > MAX\_TEMP:

print('The temperature is too’)

Part 1:

# This program assists a technician in the process

# of checking a substance's temperature.

# MAX\_TEMP is used as a global constant for

# the maximum temperature.

MAX\_TEMP = 102.5

# The main function

def main():

# Get the substance's temperature.

temperature = float(input("Enter the substance's Celsius temperature: "))

# As long as necessary, instruct the user to

# adjust the thermostat.

while temperature > MAX\_TEMP:

print('The temperature is too high.')

print('Turn the thermostat down and wait')

print('5 minutes. Then take the temperature')

print('again and enter it.')

temperature = float(input('Enter the new Celsius temperature: '))

# Remind the user to check the temperature again

# in 15 minutes.

print('The temperature is acceptable.')

print('Check it again in 15 minutes.')

# Call the main function.

main()

Part 2:

# This program assists a technician in the process

# of checking a substance's temperature.

# MAX\_TEMP is used as a global constant for

# the maximum temperature.

MAX\_TEMP = 102.5

# The main function

def main():

# Get the substance's temperature.

temperature = float(input("Enter the substance's Celsius temperature: "))

# As long as necessary, instruct the user to

# adjust the thermostat.

while temperature > MAX\_TEMP:

print('The temperature is too high.')

print('Turn the thermostat down and wait')

print('5 minutes. Then take the temperature')

print('again and enter it.')

temperature = float(input('Enter the new Celsius temperature: '))

# Remind the user to check the temperature again

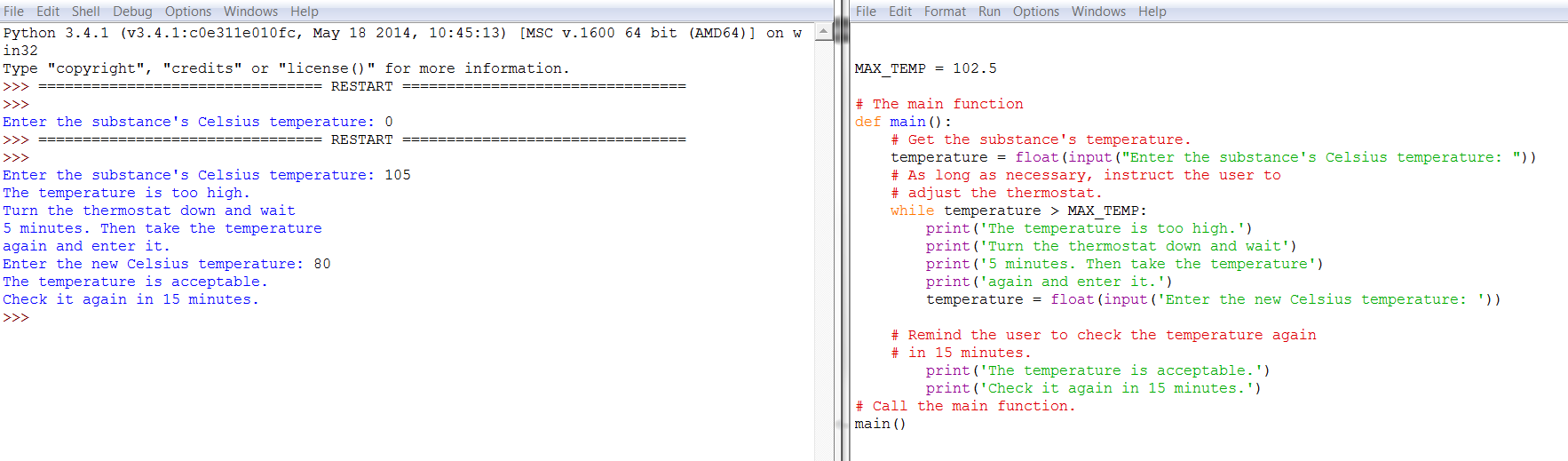
# in 15 minutes.

print('The temperature is acceptable.')

print('Check it again in 15 minutes.')

# Call the main function.

main()



Both programs do the exact same thing, the program runs until the user enters a number that is less than 102.5 degrees Celsius. Then the program will say that the temperature is okay.

Q2: simple code with if functions. What is the different logic for yellow section? Why?

#Program 5-2 (simple\_loop1.py)

# This program demonstrates a simple for loop

# that uses a list of numbers.

def main():

print('I will display the numbers 1 through 5.')

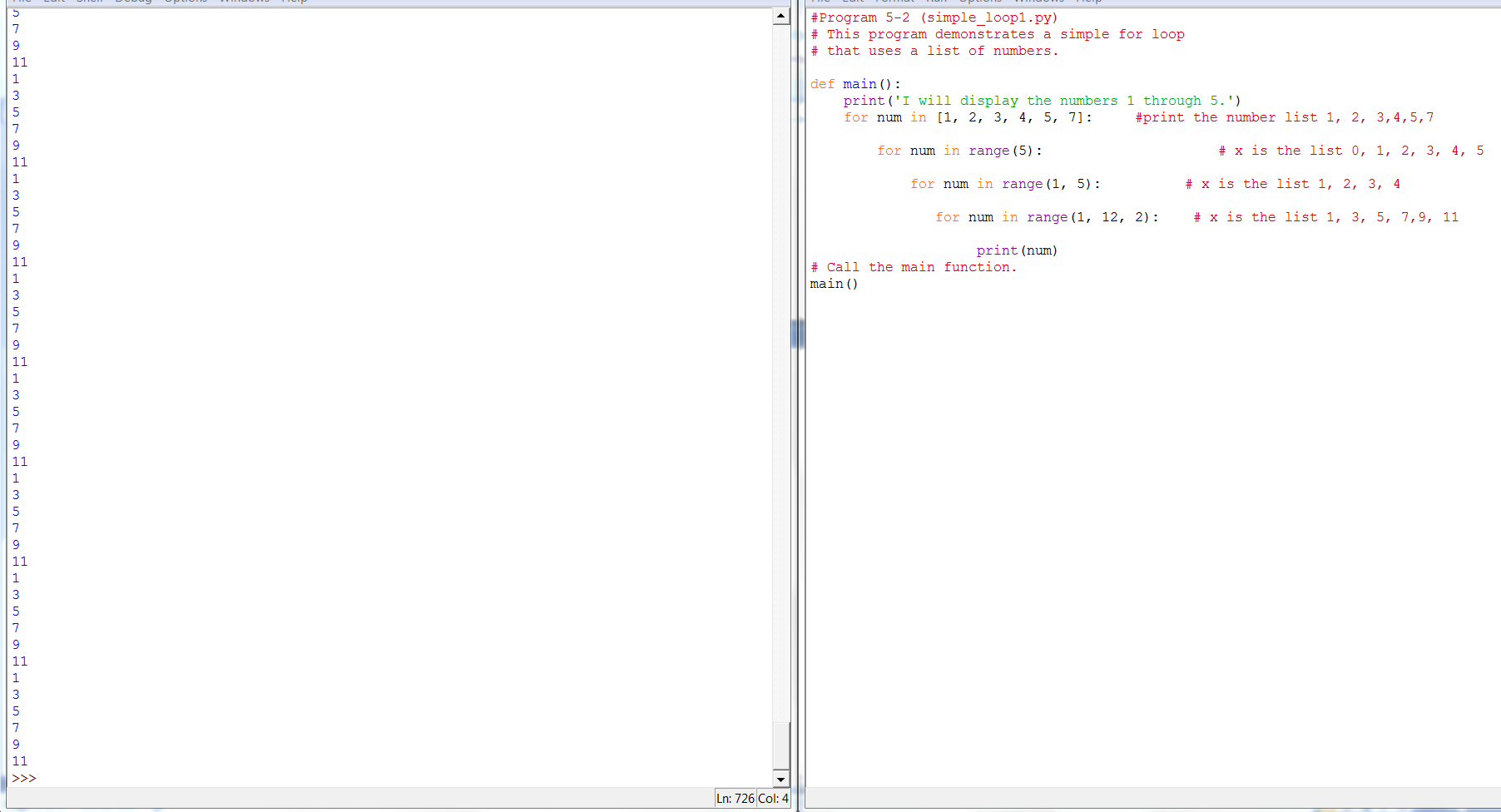
for num in [1, 2, 3, 4, 5, 7]: #print the number list 1, 2, 3,4,5,7  
 for num in range(5): # x is the list 0, 1, 2, 3, 4, 5  
 for num in range(1, 5): # x is the list 1, 2, 3, 4  
 for num in range(1, 12, 2): # x is the list 1, 3, 5, 7,9, 11

print(num)

# Call the main function.

main()

After you get this program to successfully run, capture its output and then close it!



Part 2:

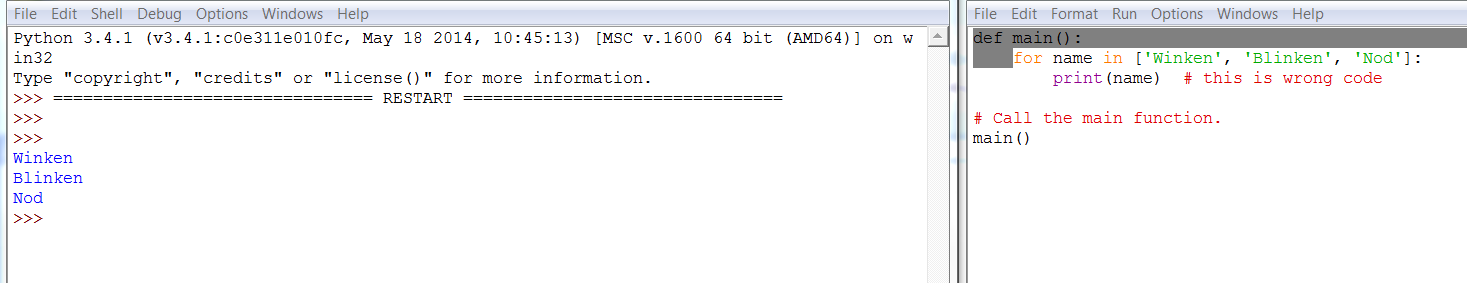
def main():

for name in ['Winken', 'Blinken', 'Nod']:

print(name) # this is wrong code

# Call the main function.

main()



Q3: random functions:

#Program 6-2 (random\_numbers2.py)

# This program displays five random

# numbers in the range of 1 through 100.

import random

def main():

for count in range(5):

# Get a random number from (1, 100]

number = random.randint(1, 100) # number from [1, 100], has 10  
 number = random.randrange(10) # number from 1 through 9, no 10, [1, 9]  
 number = random.randrange(5, 10) # the range of 5 through 9, no 10, [5, 9]  
 number = random.randrange(0, 41, 10) ) # the range [0, 41), 0, 10, 20, 30, 40

number = random.uniform(1.0, 10.0)

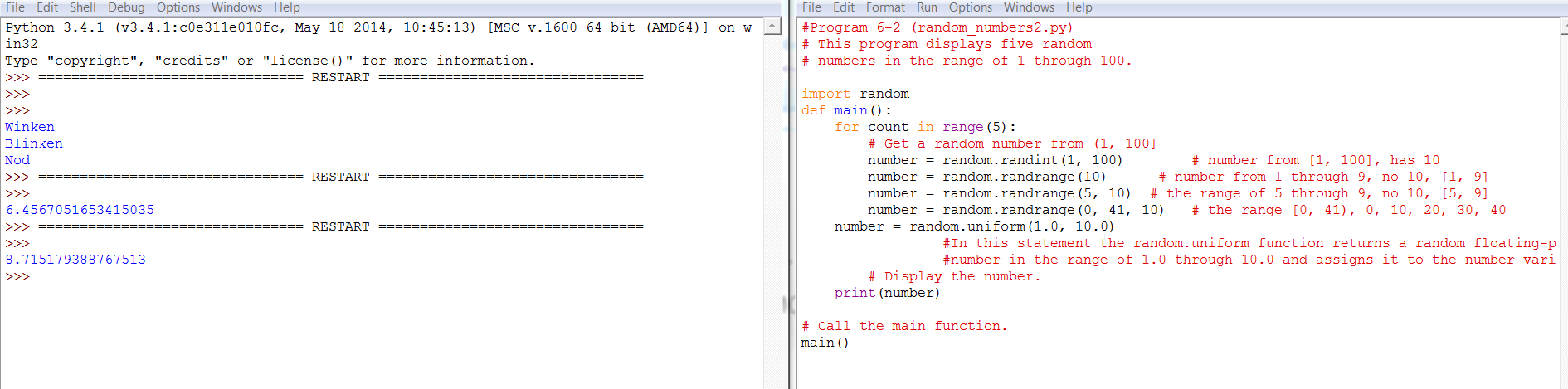
#In this statement the random.uniform function returns a random floating-point   
 #number in the range of 1.0 through 10.0 and assigns it to the number variable.

# Display the number.

print(number)

# Call the main function.

main()



## Note:

letters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'

print(letters[0:26:2]) # it is ACEGIKMOQSUWY

Result: ACEGIKMOQSUWY

full\_name = 'Patty Lynn Smith'

last\_name = full\_name[-5:] # it is Smith.  
Result: Smith