**HAUB SCHOOL OF BUSINESS**

**SAINT JOSEPH’S UNIVERSITY**

**DSS 615: Python Programming**

**Instructor: Michael Ghen**

**Assignment week 14**

By:

Vinayak Suresh Tayshetye (10673718)

**Lists, Tuples:**

Section 2.4 Exercises 60-80 even

**60.**

['all', 'for', 'one']

**62.**

Enter name with three parts: Guido van Rossum

Guido Rossum

**64.**

Python

**66.**

Less is more.

**68.**

merry-go-round

**70.**

['around', 'the', 'clock']

**72.**

['France', 'England', 'Spain']

**74.**

a bcd

**76.**

Live let live.

**78.**

Largest Number: 8

Length: 4

Total: 16

Number list: [6, 2, 8, 0]

**80.**

dairy

**While Loop:**

Section 3.3 Exercises 17, 19, 21

**# Q17**

**m = eval(input("Enter value of M: "))**

**n = eval(input("Enter value of N: "))**

**while n != 0:**

**temp = n**

**n = m % n**

**m = temp**

**if n == 0:**

**print("Greatest common divisor: {:d}".format(m))**

**break**

**# Q19**

**x = 1**

**while x != (x \*\* 2) - 1980:**

**x += 1**

**print("Person will be {0:d}\nin the year {1:d}.".format(x, x \*\* 2))**

**# Q21**

**stronGram = 100**

**year = 0**

**while stronGram >= 1:**

**stronGram /= 2**

**year += 28**

**print("The decay time is \n{:} years.".format(year))**

**For Loop:**

Section 3.4 Exercises 51, 53, 55

**# Q51**

**cobalt = 10**

**for year in range(5):**

**cobalt = cobalt \* .88**

**print("The amount og cobalt-60 remaining\nafter five years is {:.2f} grams.".format(cobalt))**

**# Q53**

**vowels = ["a", "e", "i", "o", "u"]**

**vowel = 0**

**phrase = input("Enter a phrase: ")**

**for letter in phrase:**

**if letter in vowels:**

**vowel += 1**

**print("The phrase contains {:} vowels.".format(vowel))**

**# Q55**

**number = 1**

**for divisor in range(2, 101):**

**number += 1 / divisor**

**print("The sum 1 + 1/2 + 1/3 + ... + 1/100\nis {:.5f} to five decimal places.".format(number))**

**Functions:**

Section 4.1 Exercises 26, 28, 30

# Q26

def main():

count()

def count():

string = input("Enter a string: ")

list = []

for letter in string:

if letter not in list:

list.append(letter)

print("Number of non-overlapping occurence of substring are {:}.".format(len(list)))

main()

# Q28

def main():

input = getN()

output = fact(input)

print("{0:}! is {1:}".format(input, output))

def getN():

while True:

try:

num = int(input("Enter a positive integer: "))

except ValueError:

print("You have not Entered a Positive Number Value.")

print("Please Enter again with the Positivet Numbers.")

else:

break

return num

def fact(num):

factVal = 1

for i in range(1, (num+1)):

factVal \*= i

return(factVal)

main()

#Q30

def main():

input = getInput()

output = calSalary(input[2])

print("New salary for {0:s} {1:s}: ${2:}".format(input[0].title(), input[1].title(), output))

def getInput():

firstName = str(input("Enter first name: "))

lastName = str(input("Enter last name:"))

currentSal = int(input("Enter current salary:"))

return(firstName, lastName, currentSal)

def calSalary(sal):

if sal >= 40000:

sal = sal + 2000 + ((sal-40000) \* 0.02)

else:

sal = sal \* 1.05

return(sal)

main()