**REPORT 3**



수강과목 : 시스템성능분석

담당교수 : 서성현 교수님

학당교과 : 컴퓨터정보공학부

학당과년 : 4학년

학당과번 : 201921285

이당과름 : 정소희

 제출일자 : 2022.04.10

<응용 예제 1>

- 전체 코드

# 클래스와 함수 선언 부분

class Node() :

    def \_\_init\_\_(self) :

        self.data = None

        self.link = None

def printNodes(start):

    current = start

    if current == None:

        return

    print(current.data, end = ' ')

    while current.link != None:

        current = current.link

        print(current.data, end = ' ')

    print()

def makeSimpleLinkedList(nameEmail):

    global memory, head, current, pre

    node = Node()

    node.data = nameEmail

    memory.append(node)

    if head == None : #첫번째 노드

        head = node

        return

    if head.data[1] > nameEmail[1]:

        node.link = head

        head = node

        return

    current = head

    while current.link !=None:

        pre = current

        current = current.link

        if current.data[1] > nameEmail[1] :

            pre.link = node

            node.link = current

            return

    current.link = node

# 전역 변수 선언

memory = []

head, current, pre = None, None, None

#메인

if \_\_name\_\_ == "\_\_main\_\_":

    while True :

        name = input("이름-->")

        if name == "" or name ==None:

            break

        email = input("이메일-->")

        makeSimpleLinkedList([name,email])

        printNodes(head)

- 코드 및 실행 화면 캡처

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

<응용 예제2>

- 전체 코드

from pickletools import read\_uint1

import random

from re import L

#클래스와 함수 선언 부분

class Node() :

    def \_\_init\_\_(self) :

        self.data = None

        self.link = None

def printNodes(start):

    current = start

    if current == None:

        return

    print(current.data, end = ' ')

    while current.link != None:

        current = current.link

        print(current.data, end = ' ')

    print()

def makeLottoList(num) :

    global memory, head, current, pre

    node = Node()

    node.data = num

    memory.append(node)

    if head == None : #첫번째 노드

        head = node

        return

    if head.data > num:

        node.link = head

        head = node

        return

    current = head

    while current.link !=None:

        pre = current

        current = current.link

        if current.data > num :

            pre.link = node

            node.link = current

            return

    current.link = node

def findNumber(num):

    global memory, head, current, pre

    if head == None:

        return False

    current = head

    if current.data == num:

        return True

    while current.link != None:

        current = current.link

        if current.data == num :

            return True

    return False

# 전역 변수 선언

memory = []

head, current, pre = None, None, None

#메인

if \_\_name\_\_ == "\_\_main\_\_":

    lottoCount = 0

    while True :

        lotto = random.randint(1,45)

        if findNumber(lotto) :

            continue

        lottoCount +=1

        makeLottoList(lotto)

        if lottoCount >=6:

            break

    printNodes(head)

- 코드 및 실행 화면 캡처

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명