def FunctionName(num1, num2):

result = (num1+num2)/2

return result

def computeNprintAverage(num1, num2) :

result = (num1 + num2) / 2

print (result)

#함수정의

def printAddress():

print("서울특별시 종로루 1번지")

print("파이썬 빌딩 7층")

print("홍길동")

#실행문

print("주소출력")

printAddress()

print("출력 끝")

#실행문

print("주소출력")

printAddress()

#미리 정의되지 않아서 오류 발생

#함수정의

def printAddress():

print("서울특별시 종로루 1번지")

print("파이썬 빌딩 7층")

print("홍길동")

#실행문

print("출력 끝")

#함수의 호출 (2/3)

#함수정의

#전달받은 인자를 정의한 변수에 저장

def computeAverage(num1,num2): #parameter

result=(num1+num2)/2

return result

#두 수에 대한 평균 구하는 프로그램

#전달인자, 인수, argument 2,6

val=computeAverage(2,6)

print("평균 =",val)

#예제 2

# 함수 정의 (원의 면적 구하는 함수)

def calculate\_area (radius):

area = 3.14 \* radius\*\*2

return area

# 본 프로그램

n = float(input("숫자 입력: "))

c\_area = calculate\_area(n)

print(c\_area)

#함수정의

def concateStr(str1, str2):

constr= str1+str2

print(constr)

return #반환값 없음, 있어도 돼고 없어도됨

#함수호출 문장

concateStr('str1','str2')

#지역변수: 함수 안에서 선언되는 변수,함수 안에서만 작동

#전역변수: 함수 외부에서 선언되는 변수

a=1

def vartest(a):

a=a+1

varest(a)

print(a)

#지역변수

#매개변수는 무조건 지역변수!! result

#전역변수 : r, area

#함수정의

def calculateArea (): #radius 지역변수

result = 3.14 \* r\*\*2 # 바깥에서 만든 r 변수를 함수 내에서 사용

return result #반환

#실행

r = float(input("원의 반지름: "))

#전역변수로 r 이라는 변수생성

area = calculateArea()

print(area)

#지역변수

#매개변수는 무조건 지역변수!! : result, radius

#전역변수: r, area

#함수정의

def calculateArea (radius): #radius 지역변수

result = 3.14 \* radius\*\*2 #

return result #반환

#실행

r = float(input("원의 반지름: "))

area = calculateArea(r) #함수를 area로 받았음

print(result)

#result 를 직접 사용할 수 있다고 생각했음.

#함수 안에서 만든 변수는 바깥에서 사용할 수 없음 .

#result 정의되지 않았다.

import random

def printColor ():

print(color)

color = random.choice(["black", "yellow", "blue", "blue"])

printColor()

#함수 안에서 전역 변수 변경(2/3)

def calculateArea (radius):

area = 3.14 \* radius\*\*2 # 전역변수 area에 계산값을 저장하려고 했다!

#새로운 지역변수 area 생성되었음

return #리턴하지 않고 내려옴

area = 0 #전역변수 area =0 로 새롭게 정의 되었음

r = float(input("원의 반지름: "))

calculateArea(r)

print(area)

#함수 안에서 전역 변수 변경(3/3)

def calculateArea (radius):

global area #global을 사용하여 전역 변수에 값을 저장한다고 알려야 한다.

area = 3.14 \* radius\*\*2 # 전역변수 area에 계산값을 저장하려고 했다!

#새로운 전역변수 area 생성되었음

return #리턴하지 않고 내려옴

area = 0 # ?? 여기서 왜 0 이 무시되었는지?

r = float(input("원의 반지름: "))

calculateArea(r)

print(area)

#원의 반지름: 10

#314.0

def greet(name, msg="별일없죠?"):

#디폴트인수 defualt argument , 맨 마지막에 위치시킴

print("안녕 ", name + ', ' + msg)

greet("영희", "숙제 했어요?" )

greet("영희")

def sayMyself(name, age, korean=True ): #한국에 사는 것이 디폴트 값

print("나의 이름은 %s입니다." % name)

print("나이는 %d살입니다." % age)

if korean:

print("서울에 삽니다.")

else:

print("외국에 삽니다.")

#실행문

sayMyself("나이화", 21)

print()

sayMyself("김한국", 22, False)

#결과

안녕 영희, 숙제 했어요?

안녕 영희, 별일없죠?

나의 이름은 나이화입니다.

나이는 21살입니다.

서울에 삽니다.

나의 이름은 김한국입니다.

나이는 22살입니다.

외국에 삽니다.

#키워드 인수

def calc(x, y, z):

return (x\*1)+(y\*2)+(z\*3)

calc(y=1,x=3,z=4)

calc(y=2,x=1,z=3)

calc(x=2,z=1,y=3)

calc(x=2,y=1,z=3)

calc(2,1,3)

>>> calc(y=1,x=3,z=4)

17

>>> calc(y=2,x=1,z=3)

14

>>> calc(x=2,z=1,y=3)

11

>>> calc(x=2,y=1,z=3)

13

>>> calc(2,1,3)

13

#함수정의 연습

def volume(width,length,height):

volume=width\*length\*height

return volume

volume(4,4,5)

#함수정의 연습(4/4)

#함수정의

def hello():

result= "Hello!"

return result

def hello():

return "Hello!"

def repeat(n):

result = 'Ha' \*n

return result

def repeat(n):

return 'Ha' \*n

#반환값이 있을 것임

def maxList(lst):

maxVal=lst[0]

for n in lst:

if maxVal < n: #크지 않다면

maxVal = n # n 바꿔치기

return maxVal

#Lab-ex1

#입력 반환값이 모두 없는 경우

def printAddress():

print("서울 특별시 서대문구 이화여대길")

print("호크마 교양 대학")

print("나이화")

>>> printAddress()

서울 특별시 서대문구 이화여대길

호크마 교양 대학

나이화

#Lab9-ex2

#parameter 한개 있고, return 값이 없는 함수 정의 및 호출

#입력 있는 경우 , 반환 값은 없는 경우

#printAddress 함수 정의

def printAddress(name):

print("서울특별시 서대문구 이화여대길")

print("호크마 교양 대학")

print(name)

>>> printAddress("나이화")

서울특별시 서대문구 이화여대길

호크마 교양 대학

나이화

#사용자에게 이름을 입력받음

inname=input("이름을 입력: ")

#입력받은 이름을 printAddress 함수를 호출하며 전달

printAddress(inname)

이름을 입력하시오: 나이화

서울특별시 서대문구 이화여대길

호크마 교양 대학

나이화

#Lab9-ex3

#parameter 한개 있고 return 값이 있는 함수 정의 및 호출

#입력이 있는 경우, 반환 값이 있는 경우

#caculateArea 함수 정의

#반지름을 입력받아 원의 면적을 구해 이를 return(반환)

def caculateArea(radius):

area=3.14\*radius\*\*2

return area

>>> val=caculateArea(5.0)

>>> print(val)

78.51

#사용자에게 반지름 입력받아

inradius=float(input("반지름을 입력하시오: "))

#원의 넓이 구하는 함수 호출(입력받은 값 전달)

#calculateArea에게 면적을 반환받아 area에 저장

area=caculateArea(inradius)

# area 에 있는 값 출력

print("반지름이 %0.2f인 원의 area")

반지름을 입력하시오: 6

113.04

#Lab9-ex4

#parameter 두개, return 값이 있는 함수 정의 및 함수 호출 연습

#getSum 함수 정의

#시작값고 종료값을 파라미터로 받아 시작값부터 종료값 까지의 숫자를 더해

#그 결과를 반환하는 함수 작성

def getSum(start, stop):

#start 부터 stop 까지의 값을 더하고 그 결과를 반환

sum=0 #누적할 변수 초기화

#종료값까지 더해야 하므로 stop+1 미만까지 반복

while start <= stop:

sum+=start

start+=1

#for n in range(start, stop+1):

# sum+=n

return sum

#1~10까지의 합을 구해서 그 결과를 출력 (getSum 함수 호출)

# getSum 함수 호출, 시작값으로 1 종료값으로 10 전달

# getSum 호출 후 반환 받은 값을 result 에 저장

result = getSum(1,10)

print(result)

#lab9-1 (별표 라인 함수)

#별표 라인을 출력하는 starLine 함수 정의 작성

#별표 몇 개 출력할 지 그 개수를 파라미터로 받게 한다.

def starLine(n): #n개의 수만큼 별표 찍는 코드 작성

print("\*"\*n)

#10개의 별을 찍기 위해 starLine 함수를 호출

#함수 호출할 때 10을 전달

>>> starLine(10)

\*\*\*\*\*\*\*\*\*\*

#lab9-2 (별표 사각형 함수 출력 함수)

#별표 사각형을 출력하는 rectangle 함수 정의 작성

#한 변의 길이를 파라미터로 받게 한다.

#반환할 것은 없음

def rectangle(n):

#한 라인을 별표를 찍는 코드 작성

i=0

while i < n:

print("\*"\*n)

i+=1

#for I in range(n):

# print(line)

#사용자한테 한 변의 길이를 정수로 입력받아

#사각형을 출력하는 rectangle 함수 호출

width=int(input(“한 변의 길이 입력: ”))

rectangle(width)

한 변의 길이 입력: 4

\*\*\*\*

\*\*\*\*

\*\*\*\*

\*\*\*\*

#Lab9-2(2) (별표 출력 함수2)

#별표 사각형을 출력하는 rectangle 함수 정의 작성

#한 라인에 찍을 별표 개수, 그것을 몇 라인 찍을 지 개수를 파라미터로 받게 한다.

#반환할 것은 없음

def rectangle(n, m):

#한 라인을 별표로 찍는 코드 작성

line = "\* "\* n

#m 라인만큼 출력

for i in range(m):

print(line)

#사용자에게 한 변의 길이와 높이를 정수로 입력받아

#사각형을 출력하는 rectangle 함수 호출

width =int(input("한 변의 길이 입력: "))

height = int(input("높이 입력: "))

rectangle(width, height)

한 변의 길이 입력: 4

높이 입력: 5

\* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \*

---------------------------------------------

#Lab9-3 (n! 계산 함수)

#factorial 함수 정의 작성

#factorial 함수 계산할 n을 파라미터로 받는다.

#n! 계산한 결과 반환

def factorial(n):

fac=1

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

fac\*=i

return fac

#팩토리얼 계산할 값을 사용자에게 입력 받아

#팩토리얼을 계산하는 factorial 함수 호출

n=int(input("팩토리얼을 구할 정수를 입력: "))

num=factorial(n)

#%d는 %s로 해도 되지만, 정수일 경우 %d로 사용 ok

print("%d!은 %d입니다."%(n,num))

#Lab9-4 (n! 계산 함수) (사용자가 종료할 때까지 반복)

#factorial 함수 정의 작성

#factorial 함수 계산할 n을 파라미터로 받는다.

#n! 계산한 결과 반환

def factorial(n):

result = 1

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

result += i

return result

#팩토리얼 계산할 값을 사용자에게 입력 받아

#팩토리얼을 계산하는 factorial 함수 호출

#사용자가 음의 정수 입력하면 양의 정수 입력하라는 메세지 출력

#사용자가 0을 입력하면 팩토리얼 계산 종료

while True:

print()

num = int(input("팩토리얼 구할 정수를 입력(0은 종료): "))

if num == 0:

print("프로그램을 종료합니다.")

break

elif num < 0:

print("양의 정수를 입력하세요.")

continue

result = factorail(num)

print("%d!은 %d입니다" %(num, result))

# Lab9-5 (도형 그리기)

import turtle as t

#다각형을 그리기 함수 정의 작성

#파라미터로 받은 n각형을 그리는 함수

def drawPolagon(n):

t.clear()

for i in range(n):

t.forward(100)

t.right(360/n)

#별 그리기 함수 정의 작성

def drawStar():

t.clear()

for i in range(5):

t.forward(100)

t.right(144)

#메뉴 출력하는 함수 정의 작성

def showMenu():

print("\*\*\*\*\*\*\* 메뉴 \*\*\*\*\*\*\*")

print("1. 삼각형 그리기")

print("2. 사각형 그리기")

print("3. 오각형 그리기")

print("4. 별 그리기")

print("0. 종료")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

#본 프로그램

while True:

showMenu() #메뉴 출력하는 함수 호출

menu = int(input("번호 입력: "))

#원하는 도형 그릴 번호 선택

if menu == 0: break

elif menu < 0 or menu >4 : continue

elif menu ==1: drawPloygon(3)

elif menu ==2: drawPloygon(4)

elif menu ==3: drawPloygon(5)

elif menu ==4: drawStar()

#Lab9-6 (문자열 -> 숫자변환)

x = input("숫자1: ")

y = input("숫자2: ")

z = input("숫자3: ")

#문자열을 숫자로 변환하여 덧셈 연산

result = int(x) + int(y) + int(z)

print(result)

#Lab9-7 (여러 줄을 한꺼번에 \n)

#아래와 같이 출력되도록 school 변수에 문자열을 저장 (\n 사용)

school = "(우) 03760\n서울특별시 서대문구 이화여대길 52\n이화여자대학교 공과대학 소프트웨어학부"

print(school)

(우) 03760

서울특별시 서대문구 이화여대길 52

이화여자대학교 공과대학 소프트웨어학부

#Lab9-8 (print 문의 끝 위치)

x = input("숫자1: ")

y = input("숫자2: ")

z = input("숫자3: ")

print(x, end=", ")

print(y, end=", ")

print(z, end=" ")

숫자1: 1

숫자2: 2

숫자3: 3

1, 2, 3

#Lab9-9 (list에 저장된 숫자들의 합)

list1 = [10, 20, 30]

list2 = ["100", "200", "300"]

#list1의 숫자들 합을 위한 변수 초기화

sum1 = 0

for item in list1:

sum1 += item

#list2의 숫자들 합을 위한 변수 초기화

sum2 = 0

for item in list2:

# list2 의 item은 문자열이므로 숫자로 변환 후 덧셈

sum2 += int(item)

print("list1의 합은 %d 입니다." %sum1)

print("list2의 합은 %d 입니다." %sum2)

list1의 합은 60 입니다.

list2의 합은 600 입니다.

#Lab9-10 (문자열 내의 개별 문자)

school = "이화여자대학교 공과대학"

subjects = ["국어", "영아", "수학", "물리"]

#문자열 슬라이싱과 인덱싱 연산

shortSchool = school[ :3] + school[4] + school[-5: ]

#subject에서 앞 글자만 모아 저장할 변수 초기화

shortSubjects = ""

#subjects 아이템을 반복해서 과목명을 가져온다

for i in range(len(subjects)):

shortSubjects += (subjects[i])[0] #각 과목의 앞 글자 추출

#과목 뒤에 쉼표 작성, 마지막 과목 뒤에는 쉼표를 붙이지 않음

if i < len(subjects)-1 :

shortSubjects += ", "

#결과 출력

print(shortSchool)

print(shortSubjects)

이화여대 공과대학

국, 영, 수, 물