파이썬프로그래밍 과제10 컴퓨터소프트웨어공학과 2-YA 20202296 전채린

10장 도전!프로그래밍

02. Tkinter 위젯 버튼 6개를 그리드 위치 이름으로 지정해 다음과 같이 출력하는 Tkinter 프로그램을 작성하시오.

[소스코드]

'10장도전!프로그래밍02'

'Tkinter 위젯 버튼 6개를 그리드 위치 이름으로 지정해 다음과 같이 출력하는 Tkinter 프로그램을 작성하시오.'

'버튼의 가로 길이는 30으로 지정'

''' 다음 구문을 사용해 6개 버튼의 모양과 표시되는 이름을 결정

rtype = ['flat', 'groove', 'raised', 'ridge', 'solid', 'sunken']

for i, t in enumerate(rtype): '''

from tkinter import \*

rtype = ['flat', 'groove', 'raised', 'ridge', 'solid', 'sunken']

win = Tk()

win.title('그리드에 배치한 버튼의 다양한 모습')

for i, t in enumerate(rtype):

for r in range(3):

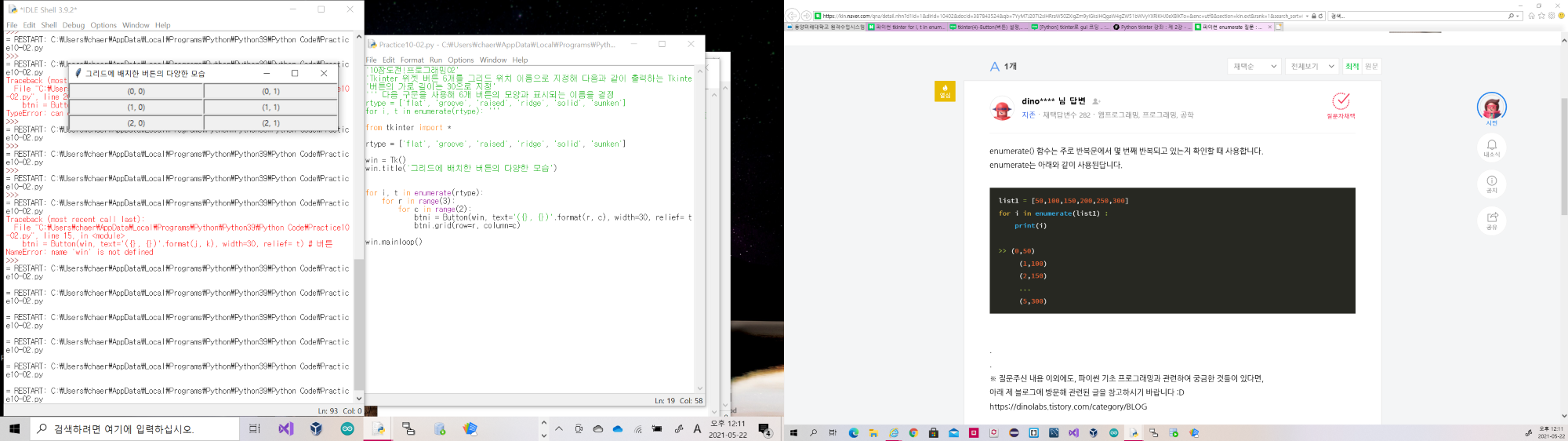
for c in range(2):

btni = Button(win, text='({}, {})'.format(r, c), width=30, relief= t) # 버튼

btni.grid(row=r, column=c)

win.mainloop()

[실행결과]



04. Tkinter 윈도 전체에 표시된 버튼 '인사 이벤트 처리'를 누르면 Tkinter의 이벤트 처리 방식 두 가지가 모두 실행되는 Tkinter 프로그램을 작성하시오.

[소스코드]

'10장도전!프로그래밍04'

"Tkinter 윈도 전체에 표시된 버튼 '인사 이벤트 처리'를 누르면 Tkinter의 이벤트 처리 방식 두 가지가 모두 실행되도록 구현한다."

'즉, 버튼을 누르면 두 가지 이벤트 처리에 대한 결과가 텍스트 모드로 출력되는 프로그램을 작성하시오.'

''' 버튼을 누르면 첫 줄은 메소드 bind()로 연결한 핸들러에서 출력하고, 두 번째 줄은 인자 command에 연결한 핸들러에서 출력'

bind: 안녕!

command: 안녕! '''

''' 버튼 btn은 다음으로 배치해 윈도 전체에 표시

btn.pack(expand=True, fill='both') '''

from tkinter import \*

def click1(e):

print('bind: 안녕!')

def click2():

print('command: 안녕!')

win = Tk()

win.title('반가워요, Tkinter!')

win.geometry('300x200')

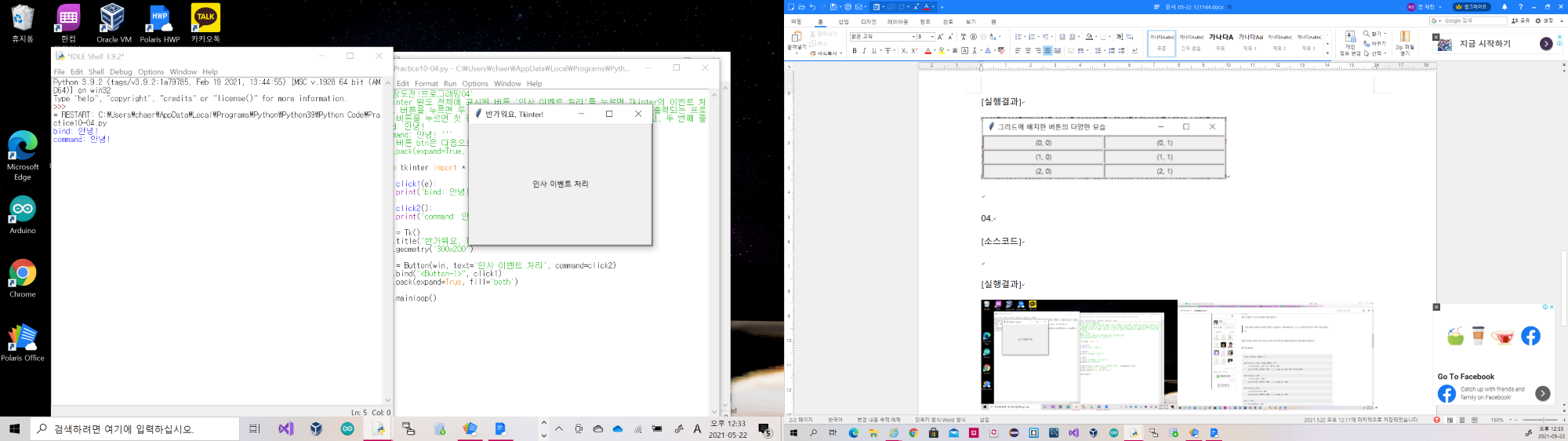
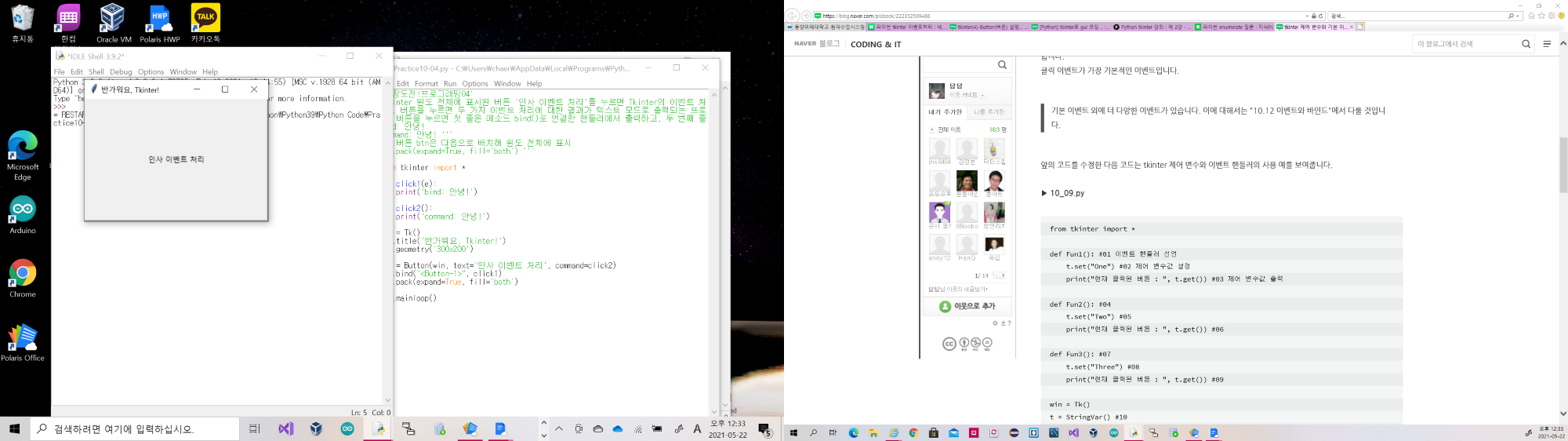
btn = Button(win, text='인사 이벤트 처리', command=click2)

btn.bind("<Button-1>", click1)

btn.pack(expand=True, fill='both')

win.mainloop()

[실행결과]



06. 마우스를 누른 체 이동하면 붉은색의 작은 원이 계속 그려지는 pygame 프로그램을 작성하시오.

[소스코드]

'10장도전!프로그래밍06'

'다음을 참고해 마우스를 누른 체 이동하면 붉은색의 작은 원이 계속 그려지는 pygame 프로그램을 작성하시오.'

'마우스를 누른 채 이동하는 위치를 리스트 mpos에 모두 저장'

''' 변수 mdown은 마우스를 누른 채 이동하는 동안 True인 변수이며,

메인 루프의 이벤트 처리 부분에서 다음 코드를 활용

for event in pygame.event.get():

if event.type == pygame.QUIT:

done = True

elif event.type == pygame.MOUSEBUTTONDOWN:

mdown = True

elif event.type == pygame.MOUSEMOTION:

if mdown:

mpos.append(event.pos)

elif event.type == pygame.MOUSEBUTTONUP:

mdown = False '''

''' 리스트 mpos의 모든 원소에 대해 원을 그림

pygame.draw.circle(screen, RED, [x, y], 2) '''

import pygame

pygame.init() # 초기화

RED = [255, 0, 0]

WHITE = [255, 255, 255]

SIZE = [600, 500] # 윈도 크기

screen = pygame.display.set\_mode(SIZE)

pygame.display.set\_caption('마우스로 그림 그리')

# 위치 좌표 리스트

mpos = []

# 화면 수정에 사용될 시계 저장

clock = pygame.time.Clock()

# 메인 루프

done = False

mdown = False

while True:

for event in pygame.event.get():

if event.type == pygame.QUIT:

done = True

elif event.type == pygame.MOUSEBUTTONDOWN:

mdown = True

elif event.type == pygame.MOUSEMOTION:

if mdown:

mpos.append(event.pos) # 리스트에 추가

elif event.type == pygame.MOUSEBUTTONUP:

mdown = False

# 배경색을 흰색으로

screen.fill(WHITE)

for pos in mpos:

# 그리기

pygame.draw.circle(screen, RED, pos, 5)

pygame.display.update()

# 초당 수정될 프레임 수 지정, 초당 20 프레임 화면이 수정됨

clock.tick(20)

pygame.quit()

[실행결과]

