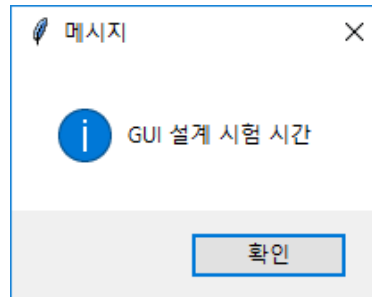
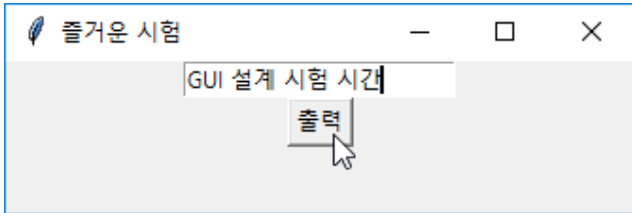


1. 다음 빈 칸을 기술하라. (11 개 x 2 점 = 22 점)

- (1) 라디오 버튼 위젯들 간의 동기화를 위해서는 라디오 버튼들이 공유하는 (제어변수)를 생성해야 한다.
- (2) 윈도우 시작, 종료, 활성화 등의 이벤트는 (윈도우 관리자(window manager))가 이벤트를 생성하며, 사용자가 윈도우 객체의 (protocol()) 메서드를 사용하여 기본 처리 동작을 재정의 할 수 있다.
- (3) 인터넷 주소에서 호스트 내의 프로세스를 구분하는 번호는 (포트 번호)이다.
- (4) TCP 서버에서 클라이언트의 연결 요청을 수락하는 socket.accept() 메서드는 클라이언트와 통신을 위한 새로운 (연결 소켓)과 (클라이언트의 주소)를 리턴 한다.
- (5) 소켓 채팅 프로그램에서 데이터 수신 시 블록킹되는 문제 해결을 위해, 데이터 수신 동작을 (스레드)로 실행한다.
- (6) 위젯의 사용자 지정 이벤트는 위젯의 (bind()) 메서드를 사용하여 지정한다.
- (7) 파일 대화상자를 위한 함수들은 (tkinter.filedialog) 모듈에 정의되어 있다.
- (8) 파이썬에서 (람다(Lambda)) 함수는 이름이 없는 한 줄 함수이다.
- (9) 2 차원 테이블의 행과 열로 구분하여 위젯을 표시할 때 위젯의 (grid()) 메서드를 사용한다.

2. 다음은 버튼을 누르면 텍스트 입력 값을 메시지 창에 출력하는 프로그램의 실행 예이다. 프로그램의 빈 곳을 완성하여라. (12 점)



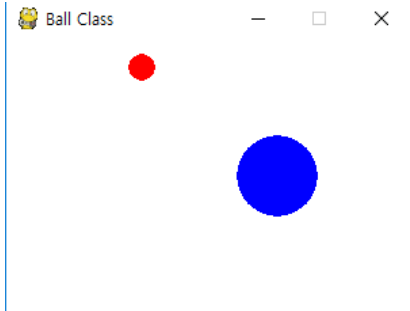
```

from tkinter import *
import tkinter.messagebox

def output_msg():
    tkinter.messagebox.showinfo("메시지", ( e.get() ) )           # 1) 메시지 창에 출력

( app = Tk() )                                               # 2) tkinter 응용 윈도우 생성
app.title("즐거운 시험")
e = Entry(app)
( e.pack() )                                                 # 3) 입력 위젯을 윈도우에 연결
( Button(app, text = "출력", command = output_msg).pack() )
                                                                # 4) 버튼 위젯을 생성하고, 윈도우에 연결
app.mainloop()
    
```

3. 다음은 2 개의 공을 그리는 프로그램의 실행 예이다. 프로그램의 빈 곳을 완성 또는 질문에 답을 하여라. (12 점)



```
import pygame
WHITE = [255, 255, 255]
RED = [255, 0, 0]
BLUE = [0, 0, 255]
class Ball():
    ( def __init__(self, px, py, cr, sz): )           # 1) 생성자 정의
        self.x = px
        self.y = py
        self.color = cr
        self.size = sz
    def draw(self, screen):
        pygame.draw.circle(screen, self.color, [self.x, self.y], self.size )
pygame.init()
size = [300,200]
screen = pygame.display.set_mode(size) # 2) 리턴되는 객체는? ( pygame.Surface )
pygame.display.set_caption("Ball Class")
done=False
clock=pygame.time.Clock()
redBall = Ball(100, 20, RED, 10)
blueBall = Ball(200, 100, BLUE, 30)
while done==False:
```

```
for event in pygame.event.get():    # 3) 동작의 의미는? ( 이벤트 큐 값들 읽기 )
    if event.type == pygame.QUIT:
        done=True
screen.fill(WHITE)
redBall.draw(screen)
blueBall.draw(screen)
( clock.tick(20) )                # 4) 20 fps 이내로 루프 실행 제어
pygame.display.flip()
pygame.quit ()
```