

ch6

2019년 7월 24일

1 tkinter 모듈 소개

Tkinter는 Tcl/Tk에 대한 파이썬 Wrapper로서 Tcl/Tk를 파이썬에 사용할 수 있도록 한 Lightweight GUI 모듈이다.

Tcl은 Tool Command Language의 약자로서 일종의 프로그래밍 언어이며, Tk는 크로스 플랫폼에 사용되는 일종의 GUI 툴킷이다.

Tkinter는 타 GUI 프레임워크나 툴킷에 비해 지원되는 위젯들이 부족하고 UI도 그렇게 예쁘지 않다는 단점이 있지만,

Python 설치시 기본적으로 내장되어 있는 파이썬 표준 라이브러리이기 때문에 쉽고 간단한 GUI 프로그램을 만들 때 활용될 수 있다.

다음과 같은 방법으로 위젯들을 화면에 배치할 수 있다.

```
In [ ]: from tkinter import *  
        root = Tk()  
  
        lbl = Label(root, text="이름")  
        lbl.pack()  
  
        txt = Entry(root)  
        txt.pack()  
  
        btn = Button(root, text="OK")  
        btn.pack()  
  
        root.mainloop()
```

1.1 Place (혹은 absolute)

위젯을 위치를 절대 좌표로 정하는 것으로, 윈도우 크기 변경에 따라 위젯들이 변경되지 않으므로 많이 사용되지 않는다.

위젯.place() 메서드를 사용한다.

1.2 Pack

위젯들을 부모 위젯에 모두 패킹하여 불필요한 공간을 없앤다.

위젯.pack() 메서드를 사용한다.

1.3 Grid

위젯들을 테이블 레이아웃에 배치하는 것으로 지정된 row, column에 위젯을 놓는다.

위젯.grid() 메서드를 사용한다.

```
In [1]: from tkinter import *  
        root = Tk()  
  
        lbl = Label(root, text="이름")  
        lbl.grid(row=0, column=0) # 0, 0에 '이름' 놓기  
        txt = Entry(root)  
        txt.grid(row=0, column=1) # 0, 1에 '텍스트 상자' 놓기  
        btn = Button(root, text="OK", width=15)  
        btn.grid(row=1, column=1) # 1, 1에 버튼 놓기  
  
        root.mainloop()
```

2 tkinter 모듈 위젯

2.1 Button

단순한 버튼

2.2 Label

텍스트 혹은 이미지 표시

2.3 CheckButton

체크박스

2.4 Entry

단순한 한 라인 텍스트 박스

2.5 ListBox

리스트 박스

2.6 RadioButton

옵션 버튼

2.7 Message

Label과 비슷하게 텍스트 표시.

Label과 달리 자동 래핑 기능이 있다.

2.8 Scale

슬라이스 바

2.9 Scrollbar

스크롤 바

2.10 Text

멀티 라인 텍스트 박스로서 일부 Rich Text 기능 제공

2.11 Menu

메뉴 Pane

2.12 Menubutton

메뉴 버튼

2.13 Toplevel

새 윈도우를 생성할 때 사용.

Tk()는 윈도우를 자동으로 생성하지만 추가로 새 윈도우
혹은 다이얼로그를 만들 경우 Toplevel를 사용한다

2.14 Frame

컨테이너 위젯. 다른 위젯들을 그룹화할 때 사용

2.15 Canvas

그래프와 점들로 그림을 그릴 수 있으며, 커스텀 위젯을 만드는데 사용될 수도 있다

```
In [3]: from tkinter import *
        from tkinter.ttk import *

        class MyFrame(Frame):
            def __init__(self, master):
                Frame.__init__(self, master)

                self.master = master
                self.master.title("고객 입력")
                self.pack(fill=BOTH, expand=True)

                # 성명
                frame1 = Frame(self)
                frame1.pack(fill=X)

                lblName = Label(frame1, text="성명", width=10)
                lblName.pack(side=LEFT, padx=10, pady=10)

                entryName = Entry(frame1)
                entryName.pack(fill=X, padx=10, expand=True)

                # 회사
                frame2 = Frame(self)
                frame2.pack(fill=X)

                lblComp = Label(frame2, text="회사명", width=10)
                lblComp.pack(side=LEFT, padx=10, pady=10)

                entryComp = Entry(frame2)
                entryComp.pack(fill=X, padx=10, expand=True)
```

```

# 특징
frame3 = Frame(self)
frame3.pack(fill=BOTH, expand=True)

lblComment = Label(frame3, text="특징", width=10)
lblComment.pack(side=LEFT, anchor=N, padx=10, pady=10)

txtComment = Text(frame3)
txtComment.pack(fill=X, pady=10, padx=10)

# 저장
frame4 = Frame(self)
frame4.pack(fill=X)
btnSave = Button(frame4, text="저장")
btnSave.pack(side=LEFT, padx=10, pady=10)

def main():
    root = Tk()
    root.geometry("600x550+100+100")
    app = MyFrame(root)
    root.mainloop()

if __name__ == '__main__':
    main()

```

3 tkinter 모듈 이벤트

```

In [ ]: from tkinter import *

def keyPressed(event):
    # 키보드 문자하나 출력
    print(event.char)

```

```

root = Tk()

frame = Frame(root, width=100, height=100)
# Key 이벤트 바인딩
frame.bind('<Key>', keyPressed)
frame.place(x=0, y=0)

# 키보드 포커를 갖게 한다
frame.focus_set()

root.mainloop()

```

```

In [15]: from tkinter import *
         from tkinter import messagebox

root = Tk()

# 버튼 클릭 이벤트 핸들러
def okClick():
    name = txt.get()
    messagebox.showinfo("이름", name)

lbl = Label(root, text="이름")
lbl.grid(row=0, column=0)
txt = Entry(root)
txt.grid(row=0, column=1)

# 버튼 클릭 이벤트와 핸들러 정의
btn = Button(root, text="OK", command=okClick)

btn.grid(row=1, column=1)

root.mainloop()

```

4 실습: 이메일 전송 프로그램 만들기

```
In [1]: from tkinter import *
        from tkinter import messagebox
        from tkinter.ttk import *
        import smtplib
        from email.mime.text import MIMEText

def SendMail(mail_from, mail_pw, mail_to, mail_title, mail_body):
    smtp = smtplib.SMTP('smtp.gmail.com', 587)
    smtp.ehlo() # say Hello
    smtp.starttls() # TLS 사용시 필요
    smtp.login(mail_from, mail_pw)

    msg = MIMEText(mail_body)
    msg['Subject'] = mail_title
    msg['To'] = mail_to
    smtp.sendmail(mail_from, mail_to, msg.as_string())

    smtp.quit()

class MyFrame(Frame):
    def __init__(self, master):
        Frame.__init__(self, master)

        self.master = master
        self.master.title("메일 작성하기")
        self.pack(fill=BOTH, expand=True)
        str1 = StringVar()
        str2 = StringVar()
        str3 = StringVar()
        str4 = StringVar()

        # 받는 이
        frame1 = Frame(self)
```

```

frame1.pack(fill=X)

lblTo = Label(frame1, text="받는 이", width=10)
lblTo.pack(side=LEFT, padx=10, pady=10)

entryTo = Entry(frame1, textvariable=str1)
entryTo.pack(fill=X, padx=10, expand=True)

# 제목
frame2 = Frame(self)
frame2.pack(fill=X)

lblTitle = Label(frame2, text="제목", width=10)
lblTitle.pack(side=LEFT, padx=10, pady=10)

entryTitle = Entry(frame2, textvariable=str2)
entryTitle.pack(fill=X, padx=10, expand=True)

# 내용
frame3 = Frame(self)
frame3.pack(fill=BOTH, expand=True)

lblBody = Label(frame3, text="내용", width=10)
lblBody.pack(side=LEFT, anchor=N, padx=10, pady=10)

txtBody = Text(frame3)
txtBody.pack(fill=X, pady=10, padx=10)

# 보내는 사람 ID
frame4 = Frame(self)
frame4.pack(fill=BOTH, expand=True)

lblFromID = Label(frame4, text="GMAIL ID", width=10)
lblFromID.pack(side=LEFT, anchor=N, padx=10, pady=10)

entryFromID = Entry(frame4, textvariable=str3)

```



```

entryFromID.pack(fill=X, padx=10, expand=True)

# 보내는 사람 비밀번호
frame5 = Frame(self)
frame5.pack(fill=BOTH, expand=True)

lblFromPW = Label(frame5, text="GMAIL PW", width=10)
lblFromPW.pack(side=LEFT, anchor=N, padx=10, pady=10)

entryFromPW = Entry(frame5, textvariable=str4, show="*")
entryFromPW.pack(fill=X, padx=10, expand=True)

# 보내기
frame6 = Frame(self)
frame6.pack(fill=X)
btnSend = Button(frame6, text="보내기", command= lambda: \
                sendClick(str3.get(), str4.get(), str1.get(), \
                str2.get(), txtBody.get(1.0, END)))
btnSend.pack(side=LEFT, padx=10, pady=10)

def sendClick(sendFromID, sendFromPW, sendTo, sendTitle, sendBody):
    if sendFromID and sendFromPW and sendTo and sendTitle and sendBody:
        SendMail(sendFromID, sendFromPW, sendTo, sendTitle, sendBody)
        messagebox.showinfo("알림", "메일 전송 완료")
    else:
        messagebox.showinfo("알림", "메일 전송 실패")

def main():
    root = Tk()
    root.geometry("600x600+100+100")
    app = MyFrame(root)
    root.mainloop()

if __name__ == '__main__':
    main()

```