

# Chapter 13 미니 프로젝트

[실습예제: 고객 정보 입력 창 구현하기 - ch13\_mini\_project\_01.py]

---본문 618쪽

```
from tkinter import Tk, Frame, PanedWindow, ttk, messagebox
from tkinter.ttk import Labelframe, Label, Entry, Button
import tkinter.font as tkFont

win = Tk()
win.geometry('%dx%d+%d+%d' %(800, 600, 5, 5))

topFrame = Frame(win)
topFrame.pack(side='top')
topFrame.config(width=800, height=100, background="#eee")

panedwindow=PanedWindow(relief="raised", bd=0)
panedwindow.pack(expand=True)

leftFrame = Frame(win)
leftFrame.pack(side='left')
leftFrame.config(width=200, height=400, background="#eee")

rightFrame = Frame(win)
rightFrame.pack(side='right')
rightFrame.config(width=600, height=470, background="green")

panedwindow.add(leftFrame)
panedwindow.add(rightFrame)

bottomFrame = Frame(win)
bottomFrame.pack(side='bottom')
bottomFrame.config(width=800, height=30, background="blue")
```

*# leftFrame 에 Entry 위젯 추가*

label\_frame = LabelFrame(leftFrame, text='기본 정보 입력')

label\_frame.pack()

lbl\_name = Label(label\_frame, text="성명:")

lbl\_phone = Label(label\_frame, text="전화번호:")

lbl\_email = Label(label\_frame, text="이메일:")

entry\_name = Entry(label\_frame)

entry\_phone = Entry(label\_frame)

entry\_email = Entry(label\_frame)

lbl\_name.grid(row=0, column=0)

entry\_name.grid(row=0, column=1, padx=5, pady=5)

lbl\_phone.grid(row=1, column=0)

entry\_phone.grid(row=1, column=1, padx=5, pady=5)

lbl\_email.grid(row=2, column=0)

entry\_email.grid(row=2, column=1, padx=5, pady=5)

*# bottomFrame 에 Button 위젯 추가하기*

panedwindow=PanedWindow(bottomFrame, relief="raised", bd=0)

panedwindow.pack()

btn\_output = Button(panedwindow, text="전체보기")

btn\_input = Button(panedwindow, text="입력")

btn\_search = Button(panedwindow, text="검색")

btn\_modify = Button(panedwindow, text="수정")

btn\_delete = Button(panedwindow, text="삭제")

btn\_backup = Button(panedwindow, text="파일백업")

btn\_load = Button(panedwindow, text="파일로드")

	<pre> panedwindow.add(btn_output) panedwindow.add(btn_input) panedwindow.add(btn_search) panedwindow.add(btn_modify) panedwindow.add(btn_delete) panedwindow.add(btn_backup) panedwindow.add(btn_load)  # 앱 타이틀 라벨 추가 fontStyle = tkFont.Font(family="궁서체", size=28) lbl_title = Label(topFrame, text="고객 관리 시스템", font=fontStyle) lbl_title.place(relx=0.5, rely=0.5, anchor='center')  if __name__ == '__main__':     win.mainloop() </pre>
--	---

[실습예제: 고객 목록 테이블 추가하기- ch13\_mini\_project\_10.py]

-----본문 621쪽

	<pre> from tkinter import Tk, Frame, PanedWindow, ttk, messagebox from tkinter.ttk import Labelframe, Label, Entry, Button import tkinter.font as tkFont  win = Tk() win.geometry('%dx%d+%d+%d' %(800, 600, 5, 5))  topFrame = Frame(win) topFrame.pack(side='top') topFrame.config(width=800, height=100, background="#eee")  panedwindow=PanedWindow(relief="raised", bd=0) panedwindow.pack(expand=True) </pre>
--	--

```
leftFrame = Frame(win)
leftFrame.pack(side='left')
leftFrame.config(width=200, height=400, background="#eee")

rightFrame = Frame(win)
rightFrame.pack(side='right')
rightFrame.config(width=600, height=470, background="green")

panedwindow.add(leftFrame)
panedwindow.add(rightFrame)

bottomFrame = Frame(win)
bottomFrame.pack(side='bottom')
bottomFrame.config(width=800, height=30, background="blue")

# leftFrame 에 Entry 위젯 추가
label_frame = LabelFrame(leftFrame, text='기본 정보 입력')
label_frame.pack()

lbl_name = Label(label_frame, text="성명:")
lbl_phone = Label(label_frame, text="전화번호:")
lbl_email = Label(label_frame, text="이메일:")

entry_name = Entry(label_frame)
entry_phone = Entry(label_frame)
entry_email = Entry(label_frame)

lbl_name.grid(row=0, column=0)
entry_name.grid(row=0, column=1, padx=5, pady=5)
lbl_phone.grid(row=1, column=0)
entry_phone.grid(row=1, column=1, padx=5, pady=5)
lbl_email.grid(row=2, column=0)
```

```
entry_email.grid(row=2, column=1, padx=5, pady=5)

# bottomFrame 에 Button 위젯 추가하기
panedwindow=PanedWindow(bottomFrame, relief="raised", bd=0)
panedwindow.pack()

btn_output = Button(panedwindow, text="전체보기")
btn_input = Button(panedwindow, text="입력")
btn_search = Button(panedwindow, text="검색")
btn_modify = Button(panedwindow, text="수정")
btn_delete = Button(panedwindow, text="삭제")
btn_backup = Button(panedwindow, text="파일백업")
btn_load = Button(panedwindow, text="파일로드")

panedwindow.add(btn_output)
panedwindow.add(btn_input)
panedwindow.add(btn_search)
panedwindow.add(btn_modify)
panedwindow.add(btn_delete)
panedwindow.add(btn_backup)
panedwindow.add(btn_load)

# rightFrame 에 목록 추가하기
header_list = ['no', 'name', 'phone number', 'e-mail']
data_list = [
    (1,'kim', '010-2222-1111', 'kim@comstudy21.or.kr'),
    (2,'lee', '010-2222-2222', 'lee@comstudy21.or.kr'),
    (3,'park', '010-2222-3333', 'park@comstudy21.or.kr'),
    (4,'kang', '010-2222-4444', 'kang@comstudy21.or.kr')
]
```

```
tree = ttk.Treeview(rightFrame, columns=header_list, show="headings")
tree.pack()

for i, col in enumerate(header_list):
    tree.heading(col, text=col.title())
for i, item in enumerate(data_list):
    tree.insert("", 'end', iid='IID%d' % (i), values=item)

# TreeView 의 각 필드 너비 설정
tree.column(0, width=50)
tree.column(1, width=80)
tree.column(2, width=150)
tree.column(3, width=250)

# TreeView 의 높이 설정
tree.config(height=22)

# 앱 타이틀 라벨 추가
fontStyle = tkFont.Font(family="궁서체", size=28)
lbl_title = Label(topFrame, text="고객 관리 시스템", font=fontStyle)
lbl_title.place(relx=0.5, rely=0.5, anchor='center')

if __name__ == '__main__':
    win.mainloop()
```

```
from tkinter import Tk, Frame, PanedWindow, ttk, messagebox
from tkinter.ttk import Labelframe, Label, Entry, Button
import tkinter.font as tkFont

def deleteTreeData() :
    children = tree.get_children()
    if children != '():
        for item in children:
            tree.delete(item)

def refreshTreeData(data_list) :
    for i, item in enumerate(data_list):
        tree.insert("", 'end', iid='IID%d' % (i), values=item)

def refreshTreeview(data_list) :
    deleteTreeData()
    refreshTreeData(data_list)

# 이벤트 핸들러
def outputEvtHandler():
    #print("전체 보기 ...")
    refreshTreeview(data_list)

seq = 5
def inputEvtHandler():
    name = entry_name.get()
    phone = entry_phone.get()
    email = entry_email.get()
    if name=="" or phone=="" or email=="":
        messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')
        return

    global seq
    data_list.append((seq, name, phone, email))
    #tree.insert("", 'end', values=(seq, name, phone, email))
    seq += 1

    refreshTreeview(data_list)
```

```

def searchEvtHandler():
    #print("검색 ...")
    sname = entry_name.get()
    if sname == "" :
        messagebox.showinfo('경고','이름을 입력 하고 검색 하세요!')
        return

    #search_data_list = [(2,'lee', '010-1111-1111', 'lee@comstudy21.or.kr')]
    search_data_list = []
    for data in data_list :
        try :
            idx = data.index(sname)
            search_data_list.append(data)
        except :
            pass
    if search_data_list == [] :
        messagebox.showinfo('경고', '찾는 정보가 없습니다!')
        return

    #print(search_data_list)
    refreshTreeview(search_data_list)

def modifyEvtHandler():
    print("수정 ...")
    sname = entry_name.get()
    phone = entry_phone.get()
    email = entry_email.get()
    if sname == "" or phone == "" or email == "":
        messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')
        return

    idx = -1
    for data in data_list :
        try :
            idx = data.index(sname)
            data_list[idx] = (data[0], sname, phone, email)
            refreshTreeview(data_list)

```



	<pre>        return     except :         pass      print(idx)     if idx == -1 :         messagebox.showinfo('경고', '찾는 정보가 없습니다!')  def deleteEvtHandler():     print("삭제 ...")     sname = entry_name.get()     if sname == "":         messagebox.showinfo('경고', '이름을 입력 하고 검색 하세요!')     return      idx = -1     for data in data_list:         try:             idx = data.index(sname)             del data_list[data_list.index(data)]             refreshTreeview(data_list)         except:             pass      if idx == -1 :         messagebox.showinfo('경고', '찾는 정보가 없습니다!')  def backupEvtHandler():     print("파일로 백업 ...")  def loadEvtHandler():     print("파일로 로드 ...")</pre>
--	---

```
win = Tk()
win.geometry('%dx%d+%d+%d' %(800, 600, 5, 5))

topFrame = Frame(win)
topFrame.pack(side='top')
topFrame.config(width=800, height=100, background="#eee")

panedwindow=PanedWindow(relief="raised", bd=0)
panedwindow.pack(expand=True)

leftFrame = Frame(win)
leftFrame.pack(side='left')
leftFrame.config(width=200, height=400, background="#eee")

rightFrame = Frame(win)
rightFrame.pack(side='right')
rightFrame.config(width=600, height=470, background="green")

panedwindow.add(leftFrame)
panedwindow.add(rightFrame)

bottomFrame = Frame(win)
bottomFrame.pack(side='bottom')
bottomFrame.config(width=800, height=30, background="blue")

# leftFrame 에 Entry 위젯 추가
label_frame = LabelFrame(leftFrame, text='기본 정보 입력')
label_frame.pack()

lbl_name = Label(label_frame, text="성명:")
lbl_phone = Label(label_frame, text="전화번호:")
lbl_email = Label(label_frame, text="이메일:")

entry_name = Entry(label_frame)
entry_phone = Entry(label_frame)
entry_email = Entry(label_frame)

lbl_name.grid(row=0, column=0)
```

```

entry_name.grid(row=0, column=1, padx=5, pady=5)
lbl_phone.grid(row=1, column=0)
entry_phone.grid(row=1, column=1, padx=5, pady=5)
lbl_email.grid(row=2, column=0)
entry_email.grid(row=2, column=1, padx=5, pady=5)

# bottomFrame 에 Button 위젯 추가하기
panedwindow=PanedWindow(bottomFrame, relief="raised", bd=0)
panedwindow.pack()

btn_output = Button(panedwindow, text="전체보기", command=outputEvtHandler)
btn_input = Button(panedwindow, text="입력", command=inputEvtHandler)
btn_search = Button(panedwindow, text="검색", command=searchEvtHandler)
btn_modify = Button(panedwindow, text="수정", command=modifyEvtHandler)
btn_delete = Button(panedwindow, text="삭제", command=deleteEvtHandler)
btn_backup = Button(panedwindow, text="파일백업", command=backupEvtHandler)
btn_load = Button(panedwindow, text="파일로드", command=loadEvtHandler)

panedwindow.add(btn_output)
panedwindow.add(btn_input)
panedwindow.add(btn_search)
panedwindow.add(btn_modify)
panedwindow.add(btn_delete)
panedwindow.add(btn_backup)
panedwindow.add(btn_load)

# rightFrame 에 목록 추가하기
header_list = ['no', 'name', 'phone number', 'e-mail']
data_list = [
(1,'kim', '010-2222-1111', 'kim@comstudy21.or.kr'),
(2,'lee', '010-2222-2222', 'lee@comstudy21.or.kr'),
(3,'park', '010-2222-3333', 'park@comstudy21.or.kr'),
(4,'kang', '010-2222-4444', 'kang@comstudy21.or.kr')
]

tree = ttk.Treeview(rightFrame, columns=header_list, show="headings")

```

	<pre> tree.pack()  for i, col in enumerate(header_list):     tree.heading(col, text=col.title()) for i, item in enumerate(data_list):     tree.insert('', 'end', iid='IID%d' %i, values=item)  # TreeView 의 각 필드 너비 설정 tree.column(0, width=50) tree.column(1, width=80) tree.column(2, width=150) tree.column(3, width=250)  # TreeView 의 높이 설정 tree.config(height=22)  # 앱 타이틀 라벨 추가 fontStyle = tkFont.Font(family="궁서체", size=28) lbl_title = Label(topFrame, text="고객 관리 시스템", font=fontStyle) lbl_title.place(relx=0.5, rely=0.5, anchor='center')  if __name__ == '__main__':     win.mainloop() </pre>
--	--

[실습예제: DB 연동 - ch13\_mini\_project\_20.py]

-----본문 633쪽

	<pre> from tkinter import Tk, Frame, PanedWindow, ttk, messagebox from tkinter.ttk import Labelframe, Label, Entry, Button import tkinter.font as tkFont  from member_dao import *  def init_entry(name, phone, email) :     entry_name.delete(0, len(name))     entry_phone.delete(0, len(phone)) </pre>
--	---

	<pre> entry_email.delete(0, len(email))  def deleteTreeData() :     children = tree.get_children()     if children != '0':         for item in children:             tree.delete(item)  def refreshTreeData(data_list) :     for i, item in enumerate(data_list):         tree.insert('', 'end', iid='IID%d' % (i), values=item)  def refreshTreeview(data_list) :     deleteTreeData()     refreshTreeData(data_list)  # 이벤트 핸들러 def outputEvtHandler():     #print("전체 보기 ...")     global data_list     data_list = select_all()     refreshTreeview(data_list)  def inputEvtHandler():     name = entry_name.get()     phone = entry_phone.get()     email = entry_email.get()     if name=="" or phone=="" or email=="" :         messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')         return      insert((name,phone,email))     data_list = select_all() </pre>
--	--

```
refreshTreeview(data_list)
```

```
init_entry(name, phone, email)
```

```
def searchEvtHandler():
```

```
    #print("검색 ...")
```

```
    sname = entry_name.get()
```

```
    if sname == "" :
```

```
        messagebox.showinfo('경고','이름을 입력 하고 검색 하세요!')
```

```
        return
```

```
    search_data_list = select(sname)
```

```
    if search_data_list == [] :
```

```
        messagebox.showinfo('경고', '찾는 정보가 없습니다!')
```

```
        return
```

```
    #print(search_data_list)
```

```
    refreshTreeview(search_data_list)
```

```
def modifyEvtHandler():
```

```
    print("수정 ...")
```

```
    sname = entry_name.get()
```

```
    phone = entry_phone.get()
```

```
    email = entry_email.get()
```

```
    if sname == "" or phone == "" or email == "":
```

```
        messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')
```

```
        sname.focus()
```

```
        return
```

```
    idx = -1
```

```
    for data in data_list :
```

```
        try :
```

```

        idx = data.index(sname)
        update((sname, phone, email, data[0]) )
        refreshTreeview(select_all())
        init_entry(sname, phone, email)
        return
    except :
        pass

print(idx)
if idx == -1 :
    messagebox.showinfo('경고', '찾는 정보가 없습니다!')

```

```
def deleteEvtHandler():
```

```

    #print("삭제 ...")
    sname = entry_name.get()
    if sname == "":
        messagebox.showinfo('경고', '이름을 입력 하고 검색 하세요!')
        return

```

```

    idx = -1
    for data in data_list:
        try:
            idx = data.index(sname)
            delete(data[0])
            refreshTreeview(select_all())
            init_entry(sname,"","")
            return
        except:
            pass

```

```

    if idx == -1 :

```

```
messagebox.showinfo('경고', '찾는 정보가 없습니다!')
```

```
def backupEvtHandler():  
    print("파일로 백업 ...")
```

```
def loadEvtHandler():  
    print("파일로 로드 ...")
```

```
create_table()
```

```
win = Tk()  
win.geometry('%dx%d+%d+%d' %(800, 600, 5, 5))
```

```
topFrame = Frame(win)  
topFrame.pack(side='top')  
topFrame.config(width=800, height=100, background="#eee")
```

```
panedwindow=PanedWindow(relief="raised", bd=0)  
panedwindow.pack(expand=True)
```

```
leftFrame = Frame(win)  
leftFrame.pack(side='left')  
leftFrame.config(width=200, height=400, background="#eee")
```

```
rightFrame = Frame(win)  
rightFrame.pack(side='right')  
rightFrame.config(width=600, height=470, background="green")
```

```
panedwindow.add(leftFrame)  
panedwindow.add(rightFrame)
```



```
bottomFrame = Frame(win)
bottomFrame.pack(side='bottom')
bottomFrame.config(width=800, height=30, background="blue")

# leftFrame 에 Entry 위젯 추가
label_frame = LabelFrame(leftFrame, text='기본 정보 입력')
label_frame.pack()

lbl_name = Label(label_frame, text="성명:")
lbl_phone = Label(label_frame, text="전화번호:")
lbl_email = Label(label_frame, text="이메일:")

entry_name = Entry(label_frame)
entry_phone = Entry(label_frame)
entry_email = Entry(label_frame)

lbl_name.grid(row=0, column=0)
entry_name.grid(row=0, column=1, padx=5, pady=5)
lbl_phone.grid(row=1, column=0)
entry_phone.grid(row=1, column=1, padx=5, pady=5)
lbl_email.grid(row=2, column=0)
entry_email.grid(row=2, column=1, padx=5, pady=5)

# bottomFrame 에 Button 위젯 추가하기
panedwindow=PanedWindow(bottomFrame, relief="raised", bd=0)
panedwindow.pack()

btn_output = Button(panedwindow, text="전체보기", command=outputEvtHandler)
btn_input = Button(panedwindow, text="입력", command=inputEvtHandler)
btn_search = Button(panedwindow, text="검색", command=searchEvtHandler)
btn_modify = Button(panedwindow, text="수정", command=modifyEvtHandler)
```

```

btn_delete = Button(panedwindow, text="삭제", command=deleteEvtHandler)
btn_backup = Button(panedwindow, text="파일백업", command=backupEvtHandler)
btn_load = Button(panedwindow, text="파일로드", command=loadEvtHandler)

panedwindow.add(btn_output)
panedwindow.add(btn_input)
panedwindow.add(btn_search)
panedwindow.add(btn_modify)
panedwindow.add(btn_delete)
panedwindow.add(btn_backup)
panedwindow.add(btn_load)

# rightFrame 에 목록 추가하기
header_list = ['no', 'name', 'phone number', 'e-mail']
"""data_list = [
(1,'kim', '010-2222-1111', 'kim@comstudy21.or.kr'),
(2,'lee', '010-2222-2222', 'lee@comstudy21.or.kr'),
(3,'park', '010-2222-3333', 'park@comstudy21.or.kr'),
(4,'kang', '010-2222-4444', 'kang@comstudy21.or.kr')
]"""

data_list = select_all()

tree = ttk.Treeview(rightFrame, columns=header_list, show="headings")
tree.pack()

for i, col in enumerate(header_list):
    tree.heading(col, text=col.title())
for i, item in enumerate(data_list):
    tree.insert("", 'end', iid='IID%d' % (i), values=item)

# TreeView 의 각 필드 너비 설정

```

	<pre> tree.column(0, width=50) tree.column(1, width=80) tree.column(2, width=150) tree.column(3, width=250)  # TreeView 의 높이 설정 tree.config(height=22)  # 앱 타이틀 라벨 추가 fontStyle = tkFont.Font(family="궁서체", size=28) lbl_title = Label(topFrame, text="고객 관리 시스템", font=fontStyle) lbl_title.place(relx=0.5, rely=0.5, anchor='center')  if __name__ == '__main__':     win.mainloop() </pre>
--	---

다음은 지금까지 구현한 미니 프로젝트의 전체 소스 코드 내용입니다.

**[실습예제: 파일 입출력 기능 추가 - ch13\_mini\_project\_21.py]**

	<pre> from tkinter import Tk, Frame, PanedWindow, ttk, messagebox from tkinter.ttk import Labelframe, Label, Entry, Button import tkinter.font as tkFont  from member_dao import *  def init_entry(name, phone, email) :     entry_name.delete(0, len(name))     entry_phone.delete(0, len(phone))     entry_email.delete(0, len(email)) </pre>
--	---

```

def deleteTreeData() :
    children = tree.get_children()
    if children != '0':
        for item in children:
            tree.delete(item)

def refreshTreeData(data_list) :
    for i, item in enumerate(data_list):
        tree.insert('', 'end', iid='IID%d' % (i), values=item)

def refreshTreeview(data_list) :
    deleteTreeData()
    refreshTreeData(data_list)

# 이벤트 핸들러
def outputEvtHandler():
    #print("전체 보기 ...")
    global data_list
    data_list = select_all()
    refreshTreeview(data_list)

def inputEvtHandler():
    name = entry_name.get()
    phone = entry_phone.get()
    email = entry_email.get()
    if name==" or phone==" or email==" :
        messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')
        return

    insert((name,phone,email))
    data_list = select_all()
    refreshTreeview(data_list)

    init_entry(name, phone, email)

```

```

def searchEvtHandler():
    #print("검색 ...")
    sname = entry_name.get()
    if sname == "" :
        messagebox.showinfo('경고', '이름을 입력 하고 검색 하세요!')
        return

    search_data_list = select(sname)

    if search_data_list == [] :
        messagebox.showinfo('경고', '찾는 정보가 없습니다!')
        return

    #print(search_data_list)
    refreshTreeview(search_data_list)

def modifyEvtHandler():
    print("수정 ...")
    sname = entry_name.get()
    phone = entry_phone.get()
    email = entry_email.get()
    if sname == "" or phone == "" or email == "":
        messagebox.showinfo('알림', '이름, 전화번호, 이메일을 모두 입력 하세요!')
        sname.focus()
        return

    idx = -1
    for data in data_list :
        try :
            idx = data.index(sname)
            update((sname, phone, email, data[0]) )
            refreshTreeview(select_all())

```

```
        init_entry(sname, phone, email)
        return
    except :
        pass

    print(idx)
    if idx == -1 :
        messagebox.showinfo('경고', '찾는 정보가 없습니다!')
```

```
def deleteEvtHandler():
    #print("삭제/ ...")
    sname = entry_name.get()
    if sname == "":
        messagebox.showinfo('경고', '이름을 입력 하고 검색 하세요!')
        return
```

```
    idx = -1
    for data in data_list:
        try:
            idx = data.index(sname)
            delete(data[0])
            refreshTreeview(select_all())
            init_entry(sname, "", "")
            return
        except:
            pass
```

```
    if idx == -1 :
        messagebox.showinfo('경고', '찾는 정보가 없습니다!')
```

```
# pickle 모듈을 불러온다.
import pickle

def backupEvtHandler():
    print("파일로 백업 ...")
    with open('data_list.pickle', 'wb') as f:
        pickle.dump(data_list, f)

def loadEvtHandler():
    print("파일로 로드 ...")
    global data_list
    with open('data_list.pickle', 'rb') as f:
        data_list = pickle.load(f)
    refreshTreeview(data_list)

create_table()

win = Tk()
win.geometry('%dx%d+%d+%d' %(800, 600, 5, 5))

topFrame = Frame(win)
topFrame.pack(side='top')
topFrame.config(width=800, height=100, background="#eee")

panedwindow=PanedWindow(relief="raised", bd=0)
panedwindow.pack(expand=True)

leftFrame = Frame(win)
leftFrame.pack(side='left')
leftFrame.config(width=200, height=400, background="#eee")

rightFrame = Frame(win)
rightFrame.pack(side='right')
```

```
rightFrame.config(width=600, height=470, background="green")

panedwindow.add(leftFrame)
panedwindow.add(rightFrame)

bottomFrame = Frame(win)
bottomFrame.pack(side='bottom')
bottomFrame.config(width=800, height=30, background="blue")

# leftFrame 에 Entry 위젯 추가
label_frame = LabelFrame(leftFrame, text='기본 정보 입력')
label_frame.pack()

lbl_name = Label(label_frame, text="성명:")
lbl_phone = Label(label_frame, text="전화번호:")
lbl_email = Label(label_frame, text="이메일:")

entry_name = Entry(label_frame)
entry_phone = Entry(label_frame)
entry_email = Entry(label_frame)

lbl_name.grid(row=0, column=0)
entry_name.grid(row=0, column=1, padx=5, pady=5)
lbl_phone.grid(row=1, column=0)
entry_phone.grid(row=1, column=1, padx=5, pady=5)
lbl_email.grid(row=2, column=0)
entry_email.grid(row=2, column=1, padx=5, pady=5)

# bottomFrame 에 Button 위젯 추가하기
panedwindow=PanedWindow(bottomFrame, relief="raised", bd=0)
panedwindow.pack()
```



```
btn_output = Button(panedwindow, text="전체보기", command=outputEvtHandler)
btn_input = Button(panedwindow, text="입력", command=inputEvtHandler)
btn_search = Button(panedwindow, text="검색", command=searchEvtHandler)
btn_modify = Button(panedwindow, text="수정", command=modifyEvtHandler)
btn_delete = Button(panedwindow, text="삭제", command=deleteEvtHandler)
btn_backup = Button(panedwindow, text="파일백업", command=backupEvtHandler)
btn_load = Button(panedwindow, text="파일로드", command=loadEvtHandler)
```

```
panedwindow.add(btn_output)
panedwindow.add(btn_input)
panedwindow.add(btn_search)
panedwindow.add(btn_modify)
panedwindow.add(btn_delete)
panedwindow.add(btn_backup)
panedwindow.add(btn_load)
```

*# rightFrame 에 목록 추가하기*

```
header_list = ['no', 'name', 'phone number', 'e-mail']
"""data_list = [
(1,'kim', '010-2222-1111', 'kim@comstudy21.or.kr'),
(2,'lee', '010-2222-2222', 'lee@comstudy21.or.kr'),
(3,'park', '010-2222-3333', 'park@comstudy21.or.kr'),
(4,'kang', '010-2222-4444', 'kang@comstudy21.or.kr')
]"""
```

```
data_list = select_all()
```

```
tree = ttk.Treeview(rightFrame, columns=header_list, show="headings")
tree.pack()
```

```
for i, col in enumerate(header_list):
```

```
tree.heading(col, text=col.title())
for i, item in enumerate(data_list):
    tree.insert("", 'end', iid='IID%d' %i, values=item)

# TreeView 의 각 필드 너비 설정
tree.column(0, width=50)
tree.column(1, width=80)
tree.column(2, width=150)
tree.column(3, width=250)

# TreeView 의 높이 설정
tree.config(height=22)

# 앱 타이틀 라벨 추가
fontStyle = tkFont.Font(family="궁서체", size=28)
lbl_title = Label(topFrame, text="고객 관리 시스템", font=fontStyle)
#lbl_title.pack(side="bottom", anchor="center")
lbl_title.place(relx=0.5, rely=0.5, anchor='center')
#lbl_title.config(background="red")

if __name__ == '__main__':
    win.mainloop()
```