(11주차 실습)

10장(윈도우프로그래밍) 실습

2019. 11. 12.

오늘의 실습 (10장 실습)

- 다음을 Python으로 코딩하고, 소스코드와 실행결과의 capture화면을 제출하라.
  - 10장
    1. 293쪽 SELF STUDY 10-1
    2. 301쪽 SELF STUDY 10-2
    3. 303쪽 SELF STUDY 10-3
    4. 309쪽 SELF STUDY 10-4
    5. 섭씨 <-> 화씨 변환기

- 제출 : 11. 18.(월) 자정까지

- 제출물 : 1개의 아래아한글 보고서로 제출
  - 소스코드는 실행해 볼 수 있도록 표 안에 붙여넣기로 제출
  - 실행결과는 소스코드와 실행결과를 볼 수 있게 화면을 capture하여 제출

1. 293쪽 SELF STUDY 10-1

- Code10-04.py 를 수정해서 이미지를 2개 출력해 보자.
- 그림파일의 위치 및 이름
  - GIF/cat.gif
  - GIF/cat2.gif
- 힌트 : 위젯을 가로로 나타내려면 pack(side = LEFT) 를 사용한다.

Code10-04.py

```
from tkinter import *
window = Tk()

photo = PhotoImage(file = "gif/dog.gif")
label1 = Label(window, image = photo)

label1.pack();

window.mainloop()
```

출력결과

2. 301쪽 SELF STUDY 10-2

- Code10-11.py를 실행할 때마다 그림을 임의로 뒤섞어서 나타나게 하자.
- 힌트 :
  - random모듈을 임포트하고,
    - import random
  - shuffle(리스트)함수를 사용하면 리스트를 임의로 뒤섞어준다.
    - random.shuffle(리스트명)

- Code 10-11.py
  - 소스코드는 다음 슬라이드

실행결과1

실행결과2

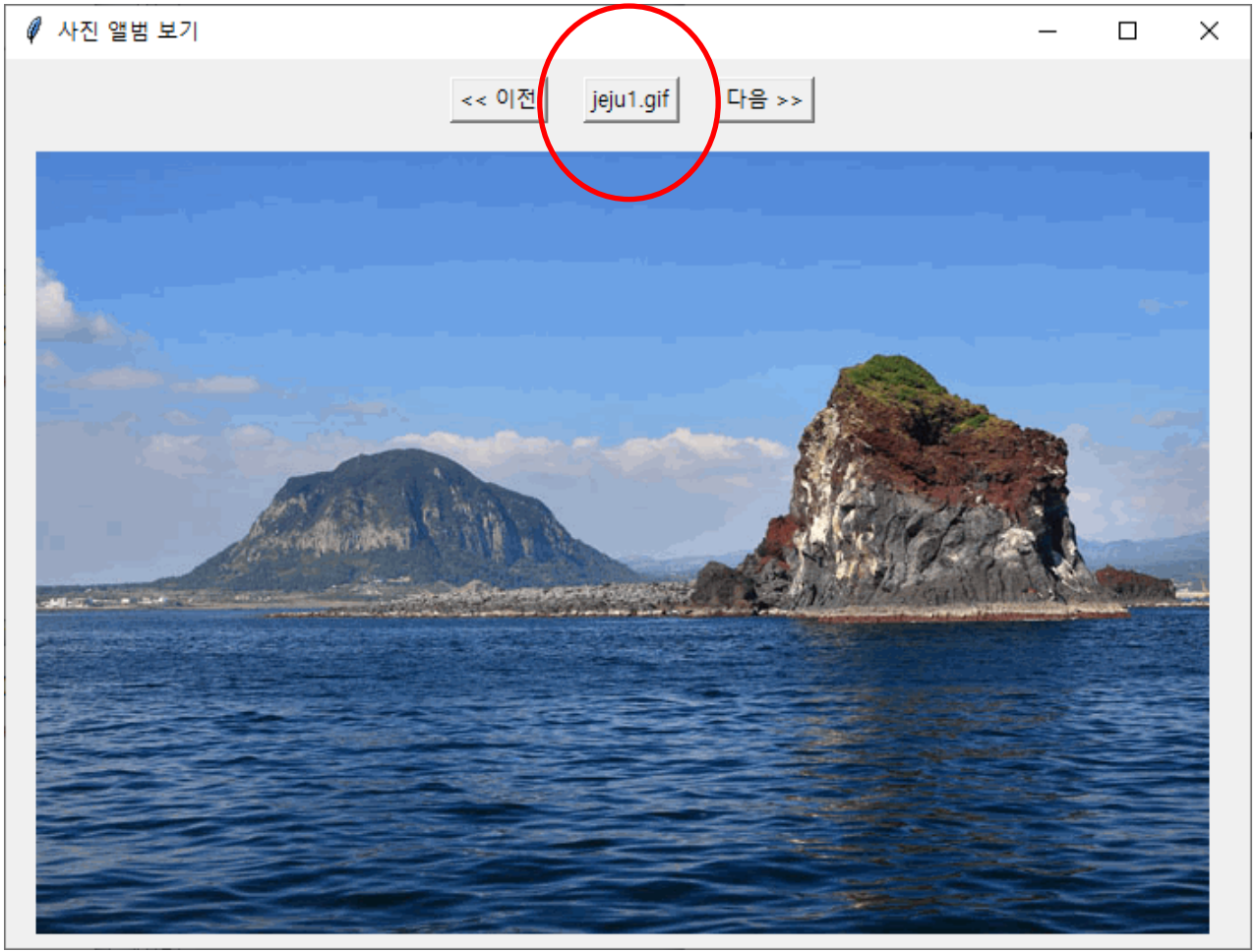
2. 301쪽 SELF STUDY 10-2

Code 10-11.py

```
from tkinter import *

## 변수 선언 부분 ##
btnList = [""] * 9
fnameList = ["froyo.gif", "gingerbread.gif", "honeycomb.gif", "icecream.gif", "jellybean.gif", "kitkat.gif", "lollipop.gif",
"marshmallow.gif", "nougat.gif"]
photoList=[None] * 9
i, k = 0, 0
xPos, yPos = 0, 0
num = 0

## 메인 코드 부분 ##
window = Tk()
window.geometry("210x210")

for i in range(0, 9) :
    photoList[i] = PhotoImage(file = "gif/" + fnameList[i])
    btnList[i] = Button(window, image = photoList[i])

for i in range(0, 3) :
    for k in range(0, 3) :
        btnList[num].place(x = xPos, y = yPos)
        num += 1
        xPos += 70
    xPos = 0
    yPos += 70

window.mainloop()
```

3. 303쪽 SELF STUDY 10-3

- Code10-12.py를 수정해서 버튼 사이에 파일명을 표기해 보자.



3. 303쪽 SELF STUDY 10-3

- Code10-12.py를 수정해서 버튼 사이에 파일명을 표기해 보자.

```
from tkinter import *
from time import *

## 전역  변수 선언 부분 ##
fnameList = ["jeju1.gif", "jeju2.gif", "jeju3.gif", "jeju4.gif", "jeju5.gif", "jeju6.gif", "jeju7.gif", "jeju8.gif", "jeju9.gif"]
photoList = [None] * 9
num = 0

## 함수 선언 부분 ##
def clickNext() :
    global num
    num += 1
    if num > 8 :
        num = 0
    photo = PhotoImage(file = "gif/" + fnameList[num])
    pLabel.configure(image = photo)
    pLabel.image = photo

def clickPrev() :
    global num
    num -= 1
    if num < 0 :
        num = 8
    photo = PhotoImage(file = "gif/" + fnameList[num])
    pLabel.configure(image = photo)
    pLabel.image=photo

## 메인 코드 부분
window = Tk()
window.geometry("700x500")
window.title("사진 앨범 보기")

btnPrev = Button(window, text = "<< 이전", command = clickPrev)
btnNext = Button(window, text = "다음 >>", command = clickNext)

photo = PhotoImage(file = "gif/" + fnameList[0])
pLabel = Label(window, image = photo)

btnPrev.place(x = 250, y = 10)
btnNext.place(x = 400, y = 10)
pLabel.place(x = 15, y = 50)

window.mainloop()
```

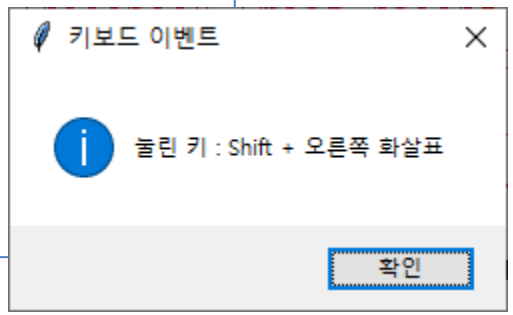
4. 309쪽 SELF STUDY 10-4

- Code10-16.py를 수정해서 shift + 화살표 키를 누르면 화살표 키가 출력되도록 해보자.
- 힌트
  - 이벤트코드는 표 10-2를 참고한다. ➔ 다음 슬라이드
  - 위젯.bind("이벤트", 함수) 코드를 한 위젯에 여러 개 작성해도 된다.

Code10-16.py

```
from tkinter import *
from tkinter import messagebox

## 함수 선언 부분 ##
def keyEvent(event) :
    messagebox.showinfo("키보드 이벤트", "눌린 키 : " + chr(event.keycode))

## 메인 코드 부분 ##
window = Tk()

window.bind("<Key>", keyEvent)

window.mainloop()
```

출력예시



4. 309쪽 SELF STUDY 10-4

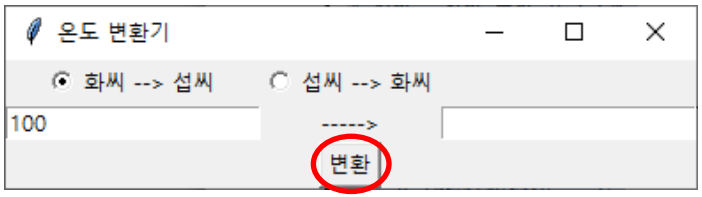
- 표 10-2 키보드 이벤트

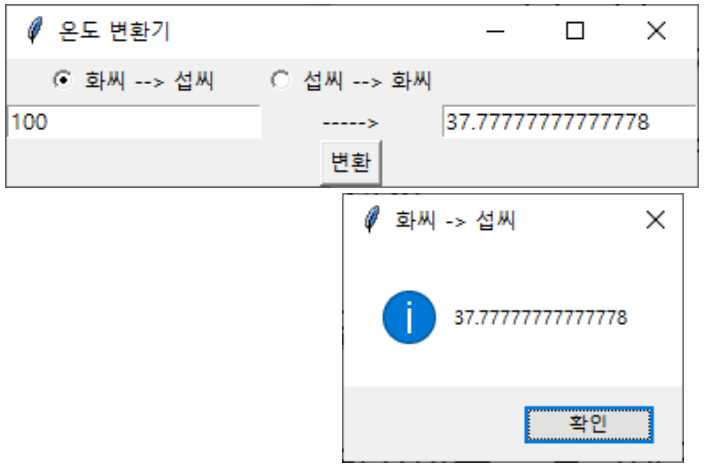
표 10-2 키보드 이벤트

키보드 작동	이벤트 코드
모든 키를 누를 때	⟨Key⟩
특수 키를 누를 때	⟨Return⟩, ⟨BackSpace⟩, ⟨Tab⟩, ⟨Shift_L⟩, ⟨Control_L⟩, ⟨Alt_L⟩, ⟨Pause⟩, ⟨Caps_Lock⟩, ⟨Escape⟩, ⟨End⟩, ⟨Home⟩, ⟨Left⟩, ⟨Right⟩, ⟨Up⟩, ⟨Down⟩, ⟨Num_Lock⟩, ⟨Delete⟩, ⟨F1⟩~⟨F12⟩ 등
일반 키를 누를 때	a~z, A~Z, 0~9, ⟨space⟩, ⟨less⟩
화살표 키와 조합	⟨Shift-Up⟩, ⟨Shift-Down⟩, ⟨Shift-Left⟩, ⟨Shift-Right⟩ 등

5. 섭씨 <-> 화씨 변환기

- 섭씨를 화씨로 바꾸는 프로그램을 작성하라.
- 요구사항
  - 1단계 라디오 버튼으로 다음을 처리
    - 섭씨 → 화씨
    - 화씨 → 섭씨
  - 2단계
    - 온도를 엔트리 위젯으로 입력/출력할 것.
  - 3단계
    - 가운데 변환버튼을 클릭하면 오른쪽의 다이알로그박스와 같은 형태의 결과 출력
- 예시





(부록) 엔트리 위젯 (1/4)

사용처

- Entry widget은 1줄 문자열을 사용자로부터 입력받을 때 사용
- 만일 여러 줄의 문자를 출력하기 원한다면 Text 위젯을 사용하여야 함
- 만일 사용자가 변경할 수 없는 한 줄 또는 여러 줄의 문자를 출력하기 원한다면 Label 위젯을 사용하여야 함

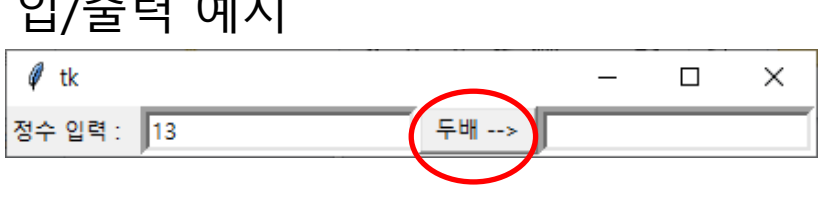
용법 w = Entry(master, option, ...)

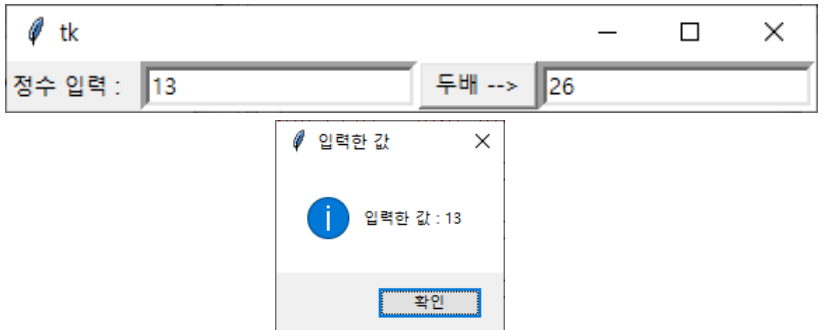
- master : 부모 윈도우
- option : 마지막 슬라이드

(부록) 엔트리 위젯 (2/4)

```
# 엔트리 위젯 실습
from tkinter import *
import tkinter.messagebox

def myfunc():
    i1 = int(E1.get()) * 2
    str1 = str(i1)
    E2.insert(0, str1)
    tkinter.messagebox.showinfo("입력한 값", "입력한 값 : " + E1.get())

root = Tk()
L1 = Label(root, text="정수 입력 :  ")
L1.pack(side = LEFT)
E1 = Entry(root, bd =5)
E1.pack(side = LEFT)
B1 = Button(root, text = " 두배 --> ", command=myfunc)
B1.pack(side = LEFT)
E2 = Entry(root, bd =5)
E2.pack(side = LEFT)
root.mainloop()
```

입/출력 예시





(부록) 엔트리 위젯 메소드 (3/4)

메소드	설명
delete(first, last=None)	Deletes characters from the widget, starting with the one at index first, up to but not including the character at position last. If the second argument is omitted, only the single character at position fir st is deleted.
get()	Returns the entry's current text as a string.
icursor (index)	Set the insertion cursor just before the character at the given index.
index (index)	Shift the contents of the entry so that the character at the given index is the leftmost visible charact er. Has no effect if the text fits entirely within the entry.
insert (index, s)	Inserts string s before the character at the given index.
select_adjust (index)	This method is used to make sure that the selection includes the character at the specified index.
select_clear()	Clears the selection. If there isn't currently a selection, has no effect.
select_from (index)	Sets the ANCHOR index position to the character selected by index, and selects that character.
select_present()	If there is a selection, returns true, else returns false.
select_range (start, end)	Sets the selection under program control. Selects the text starting at the start index, up to but not i ncluding the character at the end index. The start position must be before the end position.
select_to (index)	Selects all the text from the ANCHOR position up to but not including the character at the given in dex.
xview (index)	This method is useful in linking the Entry widget to a horizontal scrollbar.
xview_scroll (number, what)	Used to scroll the entry horizontally. The what argument must be either UNITS, to scroll by characte r widths, or PAGES, to scroll by chunks the size of the entry widget. The number is positive to scroll left to right, negative to scroll right to left.

(부록) 엔트리 위젯 옵션 (4/4)

옵션	설명
bg	The normal background color displayed behind the label and indicator.
bd	The size of the border around the indicator. Default is 2 pixels.
command	A procedure to be called every time the user changes the state of this checkbutton.
cursor	If you set this option to a cursor name (arrow, dot etc.), the mouse cursor will change to that pattern when it is ove r the checkbutton.
font	The font used for the text.
exportselection	By default, if you select text within an Entry widget, it is automatically exported to the clipboard. To avoid this expo rtation, use exportselection=0.
fg	The color used to render the text.
highlightcolor	The color of the focus highlight when the checkbutton has the focus.
justify	If the text contains multiple lines, this option controls how the text is justified: CENTER, LEFT, or RIGHT.
relief	With the default value, relief=FLAT, the checkbutton does not stand out from its background. You may set this opti on to any of the other styles
selectbackground	The background color to use displaying selected text.
selectborderwidth	The width of the border to use around selected text. The default is one pixel.
selectforeground	The foreground (text) color of selected text.
show	Normally, the characters that the user types appear in the entry. To make a .password. entry that echoes each chara cter as an asterisk, set show="*".
state	The default is state=NORMAL, but you can use state=DISABLED to gray out the control and make it unresponsive. I f the cursor is currently over the checkbutton, the state is ACTIVE.
textvariable	In order to be able to retrieve the current text from your entry widget, you must set this option to an instance of t he StringVar class.
width	The default width of a checkbutton is determined by the size of the displayed image or text. You can set this optio n to a number of characters and the checkbutton will always have room for that many characters.
xscrollcommand	If you expect that users will often enter more text than the onscreen size of the widget, you can link your entry wid get to a scrollbar.