

CES1017 Programming Fundamentals

# Week 11: OOP 사례학습

- Graphic User Interface (GUI) 활용

Instructor: Eunil Park (pa1324@hanyang.ac.kr)



**HANYANG UNIVERSITY**



## Today's Schedule

1. 사용자 인터페이스
2. Hello World
3. Counter
4. Beer Club

## User Interface

- 사용자 인터페이스(User Interface)
  - 소프트웨어와 사용자가 의사소통하는 창구
  - 명령어 인터페이스(Command Line Interface)
  - 그래픽 사용자 인터페이스(Graphic User Interface, GUI)
- “Tkinter” (TK interface)
  - <https://wiki.python.org/moin/TkInter>
  - Python에서 GUI를 활용하는 프로그램을 만들 수 있는 도구를 모아 놓은 표준 라이브러리 모듈

## 창(window) 만들기

```
1 from tkinter import *  
2  
3 window = Tk()  
4 window.title("Hello")  
5 window.geometry("200x100")  
6 window.mainloop()
```

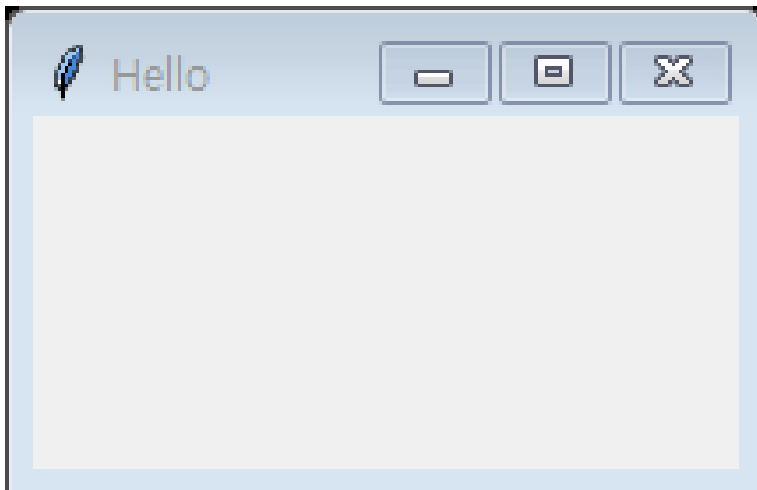
➔ Tkinter 모듈 가져오기

➔ GUI가 활동할 영역인 window 객체 만듦

➔ 창의 간판명

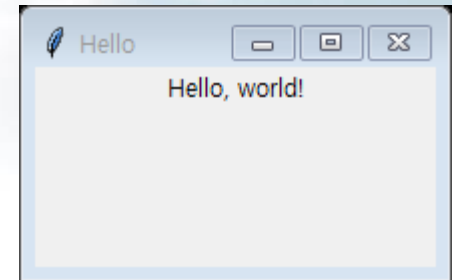
➔ 창의 크기 - 가로 200, 세로 100픽셀

➔ 창이 생기면서 이벤트 루프에 돌입



### 위젯(widget)

- 위젯(widget)
  - 버튼(button), 스크롤바(scroll bar), 메뉴(menu), 입력 상자(entry), 텍스트 상자(text box)와 같은 인터페이스 요소
- Label 위젯 객체
  - 텍스트나 이미지를 갖고 있는 위젯



```
1 from tkinter import *
2
3 window = Tk()
4 window.title("Hello")
5 window.geometry("200x100")
6 Label(window, text="Hello, world!").pack()
7 window.mainloop()
```

"Hello, world!" 문자열을 텍스트로 가지고 있는 window 소속의 위젯 객체

pack() 메소드를 호출하면 Label 위젯 객체의 창에서 텍스트의 위치를 적절하게 지정

## 프레임(Frame)

- GUI 프로그램은 이벤트 구동 (event-driven) 방식으로 작동
  - 생성된 위젯은 모두 이벤트가 발생하기를 상시 기다리고 있으므로 특정 이벤트가 발생하면 이에 맞는 반응을 하도록 프로그램을 작성해야 함

### Label 위젯

- 스스로 이벤트를 발생시켜 내용을 보여주는 기능만 가진 일방형 위젯임

### 프레임(Frame)

- 여러 위젯을 모아놓을 수 있는 컨테이너 window 자체도 프레임이지만 따로 프레임을 만들어 쓰는 것이 좋음

Frame 클래스를 상속받음  
self 는 window 안에 속한 Frame 자신임  
프레임(self) 안에 Label 객체가 있음

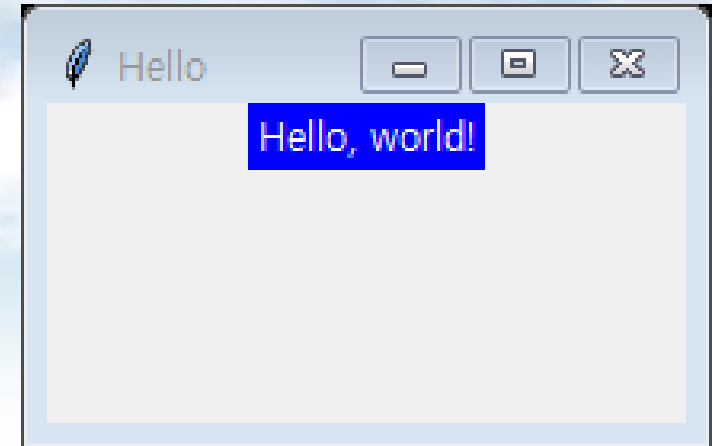
```
1 from tkinter import *
2
3 class App(Frame):
4     def __init__(self, window):
5         super().__init__(window)
6         self.pack()
7         label = Label(self)
8         label['text'] = "Hello, world!"
9         label.pack()
10
11 window = Tk()
12 window.title("Hello")
13 window.geometry("200x100")
14 App(window)
15 window.mainloop()
```



## 위젯 옵션

```
1 from tkinter import *
2
3 class App(Frame):
4     def __init__(self, window):
5         super().__init__(window)
6         self.pack()
7         label = Label(self)
8         label['text'] = "Hello, world!"
9         label['bg'] = "blue"
10        label['fg'] = "white"
11        label.pack()
12
13 window = Tk()
14 window.title("Hello")
15 window.geometry("200x100")
16 App(window)
17 window.mainloop()
```

글자의 색깔: white  
배경의 색깔: blue

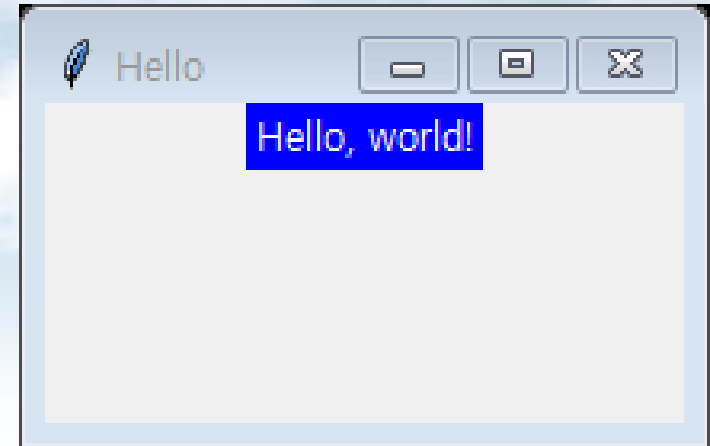


## 02. Hello World

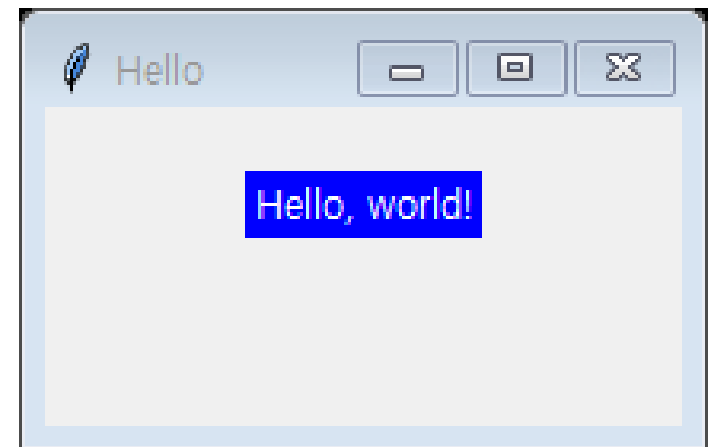


### pack() 옵션

```
1 from tkinter import *
2
3 class App(Frame):
4     def __init__(self, window):
5         super().__init__(window)
6         self.pack(padx = 20, pady = 20)
7         label = Label(self)
8         label['text'] = "Hello, world!"
9         label['bg'] = "blue"
10        label['fg'] = "white"
11        label.pack()
12
13 window = Tk()
14 window.title("Hello")
15 window.geometry("200x100")
16 App(window)
17 window.mainloop()
```



Before  
&  
After





## 02. Hello World



### 리팩토링

```
1 from tkinter import *
2
3 class App(Frame):
4     def __init__(self, window):
5         super().__init__(window)
6         self.pack(padx = 20, pady = 20)
7         Label(self, text="Hello, World!", bg = "blue", fg = "white").pack()
8
9 window = Tk()
10 window.title("Hello")
11 window.geometry("200x100")
12 App(window)
13 window.mainloop()
```

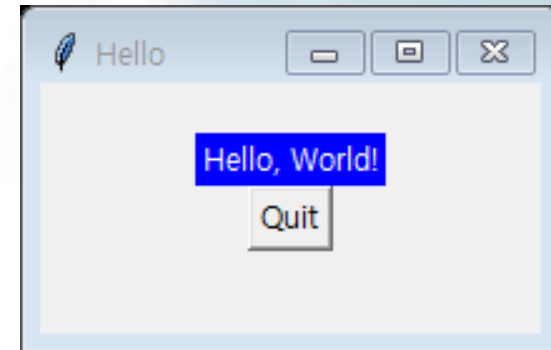
```
from tkinter import *
```

```
class App(Frame):
    def __init__(self, window):
        super().__init__(window)
        self.pack(padx = 20, pady = 20)
        label = Label(self)
        label['text'] = "Hello, world!"
        label['bg'] = "blue"
        label['fg'] = "white"
        label.pack()
```

```
window = Tk()
window.title("Hello")
window.geometry("200x100")
App(window)
window.mainloop()
```

## 버튼(Button)

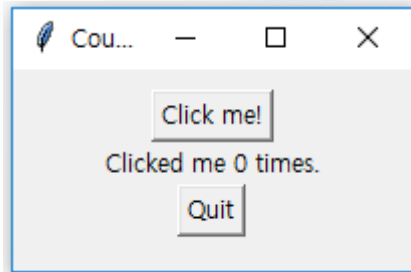
```
1 from tkinter import *
2
3 class App(Frame):
4     def __init__(self, window):
5         super().__init__(window)
6         self.pack(padx = 20, pady = 20)
7         Label(self, text="Hello, World!", bg = "blue", fg = "white").pack()
8         Button(self, text="Quit", command = self.quit).pack()
9
10 window = Tk()
11 window.title("Hello")
12 window.geometry("200x100")
13 App(window)
14 window.mainloop()
```



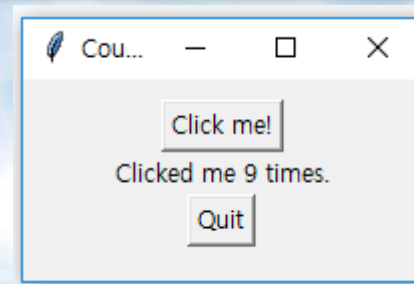
### Button 위젯

- command 옵션에는 이 버튼을 클릭했을 때 실행할 메소드의 이름을 적음
- quit 메소드는 Frame에 내장되어 있는 메소드로 호출하면 창을 닫아줌
- self.quit은 메소드 호출이 아니고, 단순히 **메소드 이름만 전달**해줌. 언젠가 이 Button을 클릭하는 이벤트가 발생하면 이 메소드가 저절로 호출됨

# 03. Counter



9회 클릭후



```
from tkinter import *
```

```
class App(Frame):
```

```
    def __init__(self, window):
        super().__init__(window)
        self.pack(padx=10, pady=10)
        self.button_clicks = 0
        self.create_widgets()
```

```
    def create_widgets(self):
```

```
        self.button = Button(self)
        self.button["text"] = "Click me!"
        self.button["command"] = self.update_count
        self.button.pack()
```

```
        self.label = Label(self)
        self.label["text"] = "Clicked me " + str(self.button_clicks) + " times."
        self.label.pack()
        Button(self, text="Quit", command = self.quit).pack()
```

```
    def update_count(self):
```

```
        self.button_clicks += 1
        self.label["text"] = "Clicked me " + str(self.button_clicks) + " times."
```

실행

```
window.title("Counter")
window.geometry("200x100")
App(window)
window.mainloop()
```

# 04. Beer Club



HANYANG  
UNIVERSITY

## Beer Club

A screenshot of a Java Swing window titled "Beer Club". The window contains a registration form with the following fields and options:

- Name:
- Email:  @hanyang.ac.kr
- Sex: ☒ male ☐ female
- Favorites:
  - ☐ Lager
  - ☒ Wheat Beer
  - ☐ Pilsners
  - ☐ Pale Ale
  - ☒ India Pale Ale
  - ☐ Stout

Below the form is a text area displaying the entered information:

```
Name: Eunil  
Email: pa1324@hanyang.ac.kr  
Sex: male  
Favorites are: Wheat Beers, India Pale Ales, ...
```

At the bottom of the window are two buttons: "Register" and "Quit".

# 04. Beer Club



HANYANG  
UNIVERSITY

## Beer Club (1/3)

```
from tkinter import *
```

```
class App(Frame):
```

```
    def __init__(self, window):  
        super().__init__(window)  
        self.pack(padx=20, pady=20)  
        self.create_widgets()
```

```
    def create_widgets(self):
```

```
        Label(self, text="Name").grid(row=0, column=0, sticky=E)  
        self.name = Entry(self, width = 10)
```

```
        self.name.grid(row=0, column = 1)
```

```
        Label(self, text = "Email").grid(row=1, column = 0, sticky=E)
```

```
        self.email = Entry(self, width = 10)
```

```
        self.email.grid(row=1, column=1)
```

```
        Label(self, text="@hanyang.ac.kr").grid(row=1, column=2, sticky = W)
```

```
        self.sex = StringVar()
```

```
        self.sex.set(None)
```

```
        Label(self, text="Sex").grid(row=2, column=0, sticky=E)
```

```
        Radiobutton(self, text='male', variable=self.sex, value='male').grid(row=2, column = 1)
```

```
        Radiobutton(self, text='female', variable=self.sex, value='female').grid(row=2, column = 2, sticky=W)
```

Beer Club

Name: Eunil  
Email: pa1324  
Sex: male  
Favorites: Wheet Beer, India Pale Ale  
Register

Name: Eunil  
Email: pa1324@hanyang.ac.kr  
Sex: male  
Favorites are: Wheet Beers, India Pale Ales, ...  
Quit



# 04. Beer Club

## Beer Club (2/3)

```
Label(self, text="Favorites").grid(row=3, column=1)
```

```
self.largers = BooleanVar()
```

```
Checkbutton(self, text="Larger", variable = self.largers).grid(row=4, column=0)
```

```
self.wheetbeer = BooleanVar()
```

```
Checkbutton(self, text="Wheet Beer", variable = self.wheetbeer).grid(row=4, column=1)
```

```
self.pilsners = BooleanVar()
```

```
Checkbutton(self, text="Pilsners", variable = self.pilsners).grid(row=4, column=2)
```

```
self.paleales = BooleanVar()
```

```
Checkbutton(self, text="Pale Ale", variable = self.paleales).grid(row=5, column=0)
```

```
self.indiapaleales = BooleanVar()
```

```
Checkbutton(self, text="India Pale Ale", variable = self.indiapaleales).grid(row=5, column=1)
```

```
self.stouts = BooleanVar()
```

```
Checkbutton(self, text="Stout", variable = self.stouts).grid(row=5, column=2)
```

```
Button(self, text="Register", command=self.write_summary).grid(row=6, column=0, columnspan=3, sticky=S)
```

```
self.summary = Text(self, width = 48, height = 10, wrap = WORD)
```

```
self.summary.grid(row = 7, column = 0, columnspan = 3, sticky = S)
```

```
Button(self, text = "Quit", command=self.quit).grid(row=8, column = 0, columnspan=3)
```

The screenshot shows a window titled "Beer Club" with a registration form. The form contains the following fields and options:

- Name:
- Email:  @hanyang.ac.kr
- Sex: ☒ male ☐ female
- Favorites section:
  - ☐ Larger
  - ☒ Wheet Beer
  - ☐ Pilsners
  - ☐ Pale Ale
  - ☒ India Pale Ale
  - ☐ Stout

Below the form is a "Register" button. A text box below the button displays the following information:

Name: Eunil  
Email: pa1324@hanyang.ac.kr  
Sex: male  
Favorites are: Wheet Beers, India Pale Ales, ...

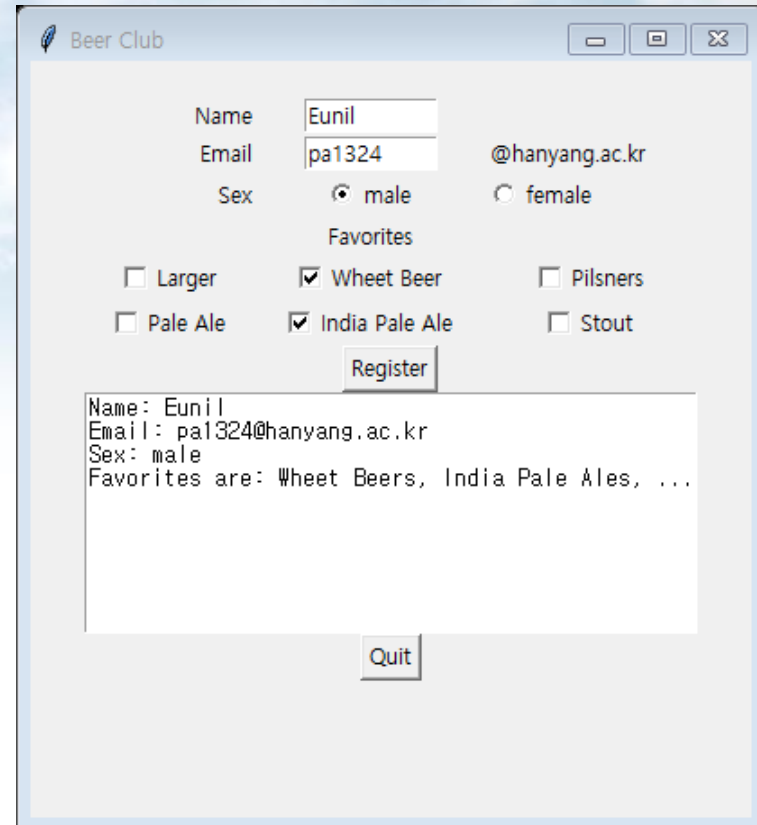
At the bottom right of the window is a "Quit" button.



## Beer Club (3/3)

```
def write_summary(self):
    summary = "Name: " + self.name.get() + "\n"
    summary += "Email: " + self.email.get() + "@hanyang.ac.kr\n"
    summary += "Sex: " + self.sex.get() + "\n"
    summary += "Favorites are: "
    if self.largers.get():
        summary += "Largers, "
    if self.wheetbeer.get():
        summary += "Wheet Beers, "
    if self.pilsners.get():
        summary += "Pilsners, "
    if self.paleales.get():
        summary += "Pale Ales, "
    if self.indiapaleales.get():
        summary += "India Pale Ales, "
    if self.stouts.get():
        summary += "Stouts, "
    summary += "..."
    self.summary.delete(0.0, END)
    self.summary.insert(0.0, summary)
```

```
window = Tk()
window.title("Beer Club")
window.geometry("400x420")
App(window)
window.mainloop()
```



The screenshot shows a Tkinter window titled "Beer Club". It contains several input fields and checkboxes. The "Name" field has "Eunil" entered, the "Email" field has "pa1324" entered, and the "Sex" field has "male" selected. Under the "Favorites" section, "Wheet Beer" and "India Pale Ale" are checked, while "Largers", "Pilsners", "Pale Ale", and "Stout" are unchecked. A "Register" button is located below the checkboxes. At the bottom of the window is a "Quit" button. A text area at the bottom displays the summary: "Name: Eunil", "Email: pa1324@hanyang.ac.kr", "Sex: male", and "Favorites are: Wheet Beers, India Pale Ales, ...".

## Summary

1. 사용자 인터페이스
2. Hello World
3. Counter
4. Beer Club

# Thanks

Week 11: OOP 사례학습 - Graphic User Interface (GUI) 활용  
Instructor: Eunil Park (pa1324@hanyang.ac.kr)

