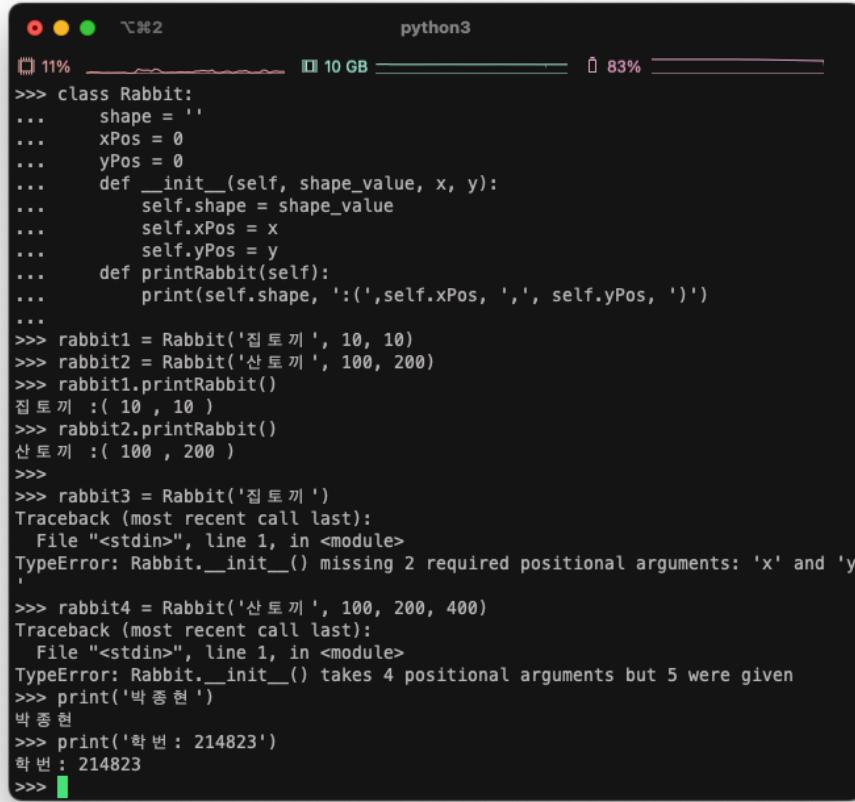


## Homework 05

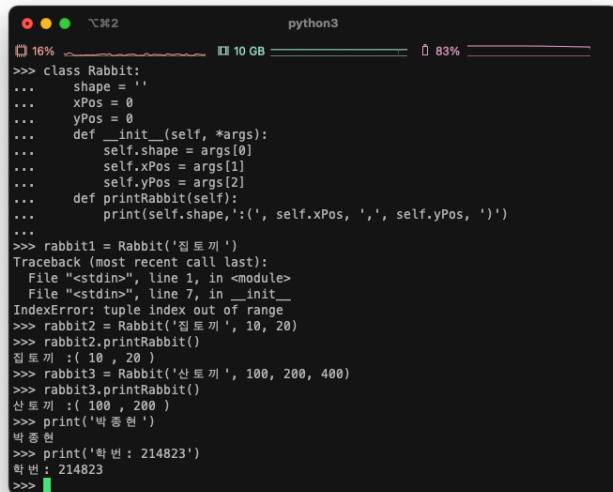
이름: 박종현

학번: 214823

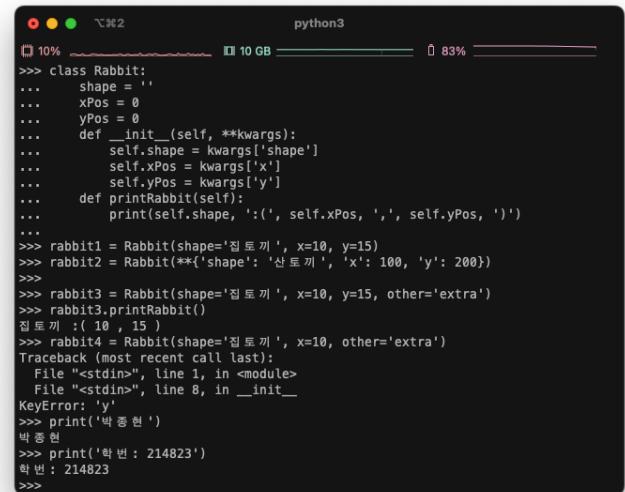
[1] 10-1. 22 27 쪽



```
python3
>>> class Rabbit:
...     shape = ''
...     xPos = 0
...     yPos = 0
...     def __init__(self, shape_value, x, y):
...         self.shape = shape_value
...         self.xPos = x
...         self.yPos = y
...     def printRabbit(self):
...         print(self.shape, ':(,self.xPos, ', self.yPos, ')')
...
>>> rabbit1 = Rabbit('집토끼', 10, 10)
>>> rabbit2 = Rabbit('산토끼', 100, 200)
>>> rabbit1.printRabbit()
집토끼 :( 10 , 10 )
>>> rabbit2.printRabbit()
산토끼 :( 100 , 200 )
>>>
>>> rabbit3 = Rabbit('집토끼')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Rabbit.__init__() missing 2 required positional arguments: 'x' and 'y'
'
>>> rabbit4 = Rabbit('산토끼', 100, 200, 400)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Rabbit.__init__() takes 4 positional arguments but 5 were given
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>
```



```
python3
>>> class Rabbit:
...     shape = ''
...     xPos = 0
...     yPos = 0
...     def __init__(self, *args):
...         self.shape = args[0]
...         self.xPos = args[1]
...         self.yPos = args[2]
...     def printRabbit(self):
...         print(self.shape, ':(, self.xPos, ', self.yPos, ')')
...
>>> rabbit1 = Rabbit('집토끼')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 7, in __init__
IndexError: tuple index out of range
>>> rabbit2 = Rabbit('집토끼', 10, 20)
>>> rabbit2.printRabbit()
집토끼 :( 10 , 20 )
>>> rabbit3 = Rabbit('산토끼', 100, 200, 400)
>>> rabbit3.printRabbit()
산토끼 :( 100 , 200 )
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>
```



```
python3
>>> class Rabbit:
...     shape = ''
...     xPos = 0
...     yPos = 0
...     def __init__(self, **kwargs):
...         self.shape = kwargs['shape']
...         self.xPos = kwargs['x']
...         self.yPos = kwargs['y']
...     def printRabbit(self):
...         print(self.shape, ':(, self.xPos, ', self.yPos, ')')
...
>>> rabbit1 = Rabbit(shape='집토끼', x=10, y=15)
>>> rabbit2 = Rabbit(**{'shape': '산토끼', 'x': 100, 'y': 200})
>>>
>>> rabbit3 = Rabbit(shape='집토끼', x=10, y=15, other='extra')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 8, in __init__
KeyError: 'y'
>>> rabbit4 = Rabbit(shape='집토끼', x=10, other='extra')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 8, in __init__
KeyError: 'y'
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>
```

## [2] 10-2. 9 10 쪽

```

python3
11% 10 GB 82%
>>> class Rabbit:
...     bag = []
...
>>> rabbit1 = Rabbit()
>>> rabbit2 = Rabbit()
>>> rabbit1.bag.append('책')
>>> rabbit2.bag.append('가위')
>>> print(rabbit1.bag)
['책', '가위']
>>> print(rabbit2.bag)
['책', '가위']
>>> print(Rabbit.bag)
['책', '가위']
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

```

python3
13% 9.5 GB 72%
>>> class Rabbit:
...     sharedBag = []
...     def __init__(self):
...         self.myBag = []
...
>>> rabbit1 = Rabbit()
>>> rabbit2 = Rabbit()
>>> rabbit1.myBag.append('책')
>>> rabbit2.myBag.append('가위')
>>> Rabbit.sharedBag.append('김밥')
>>>
>>> rabbit1.myBag
['책']
>>> rabbit2.myBag
['가위']
>>> Rabbit.sharedBag
['김밥']
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

## [3] 10-2. 15 22 쪽

```

python3
11% 10 GB 82%
>>> class Rabbit:
...     shape = ''
...     __wallet = 1000
...
>>> rabbit1 = Rabbit()
>>> rabbit1.shape = '집토끼'
>>> rabbit1.__wallet
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Rabbit' object has no attribute '__wallet'
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

```

python3
13% 10 GB 82%
>>> class Rabbit:
...     shape = ''
...     __wallet = 1000
...     def printWallet(self):
...         print(self.__wallet)
...
>>> rabbit1 = Rabbit()
>>> rabbit1.printWallet()
1000
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

```

python3
13% 10 GB 82%
>>> class Rabbit:
...     shape = ''
...     __wallet = 1000
...     def __useWallet(self, payment):
...         self.__wallet -= payment
...     def pay(self, payment):
...         self.__useWallet(payment)
...     def printWallet(self):
...         print(self.__wallet)
...
>>> rabbit1 = Rabbit()
>>> rabbit1.pay(100)
>>> rabbit1.printWallet()
900
>>> rabbit1.__useWallet(100)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Rabbit' object has no attribute '__useWallet'
>>> rabbit1.pay(100)
>>> rabbit1.printWallet()
800
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

```

python3
14% 10 GB 82%
>>> class Rabbit:
...     count = 0
...     def __init__(self):
...         Rabbit.count += 1
...     @classmethod
...     def printCount(this):
...         print('토끼 ', this.count, '마리')
...
>>> rabbit1 = Rabbit()
>>> Rabbit.printCount()
토끼 1 마리
>>> rabbit2 = Rabbit()
>>> Rabbit.printCount()
토끼 2 마리
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>>

```

```
python3
>>> class Calc:
...     @staticmethod
...     def add(a, b):
...         print(a + b)
...     @staticmethod
...     def sub(a, b):
...         print(a - b)
...
>>> Calc.add(10,20)
30
>>> Calc.sub(10,20)
-10
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>> 
```

```
python3
>>> class Rabbit:
...     shape = ''
...     def __del__(self):
...         print(self.shape, '는 없어져요 .')
...
>>> rabbit = Rabbit()
>>> rabbit.shape = '산토끼'
>>> del(rabbit)
산토끼 는 없어져요 .
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>> 
```

```
python3
>>> class Rabbit :
...     shape = ''
...     def __add__(self, other) :
...         print(self.shape, '와 ', other.shape, '가 친구가 되었습니다 .')
...
>>> rabbit1 = Rabbit()
>>> rabbit1.shape = '산토끼'
>>> rabbit2 = Rabbit()
>>> rabbit2.shape = '집토끼'
>>>
>>> rabbit1 + rabbit2
산토끼 와 집토끼 가 친구가 되었습니다 .
>>> rabbit2 + rabbit1
집토끼 와 산토끼 가 친구가 되었습니다 .
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>> 
```

[4] 10-2. 24 26 쪽

```
python3
>>> class Line:
...     length = 0
...     def __init__(self, length):
...         self.length = length
...         print(self.length, '길이의 선 생성 !')
...     def __add__(self, other):
...         return Line(self.length + other.length)
...     def __gt__(self, other):
...         return self.length > other.length
...     def __eq__(self, other):
...         return self.length == other.length
...
>>> line1 = Line(int(input('선 1의 길이 입력 : ')))
선 1의 길이 입력 : 10
10 길이의 선 생성 !
>>> line2 = Line(int(input('선 2의 길이 입력 : ')))
선 2의 길이 입력 : 20
20 길이의 선 생성 !
>>> line3 = line1 + line2
30 길이의 선 생성 !
>>> print('두 선의 길이의 합 : ', line3.length)
두 선의 길이의 합 : 30
>>>
>>> if line1 > line2:
...     print('선 1이 더 길니다 .')
... elif line1 == line2:
...     print('선 1과 선 2의 길이가 같습니다 .')
... else:
...     print('선 2가 더 길니다 .')
...
선 2가 더 길니다 .
>>> print('박종현 ')
박종현
>>> print('학번 : 214823')
학 번 : 214823
>>> 
```

## [5] 10-3. 8 11 쪽

```
python3
>>> class Rabbit:
...     shape = ''
...     xPos = 0
...     yPos = 0
...     def goto(self, x, y):
...         self.xPos = x
...         self.yPos = y
...
>>> class HouseRabbit(Rabbit):
...     owner = ''
...     def eatFeed(self):
...         print('집토끼가 모이를 먹습니다')
...
>>> class MountainRabbit(Rabbit):
...     mountain = ''
...     def eatWildglass(self):
...         print('산토끼가 들풀을 먹습니다')
...
>>> Rabbit.__dict__
mappingproxy({'__module__': '__main__', 'shape': '', 'xPos': 0, 'yPos': 0, 'goto': <function Rabbit.goto at 0x1051d9d0>, '__dict__': <attribute '__dict__' of 'Rabbit' objects>, '__weakref__': <attribute '__weakref__' of 'Rabbit' objects>, '__doc__': None})
>>> HouseRabbit.__dict__
mappingproxy({'__module__': '__main__', 'owner': '', 'eatFeed': <function HouseRabbit.eatFeed at 0x1051da020>, '__doc__': None})
>>> MountainRabbit.__dict__
mappingproxy({'__module__': '__main__', 'mountain': '', 'eatWildglass': <function MountainRabbit.eatWildglass at 0x1051d9d0>, '__doc__': None})
>>>
>>>
>>> rabbit = Rabbit()
>>> h_rabbit = HouseRabbit()
>>> m_rabbit = MountainRabbit()
>>> print(rabbit.owner)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Rabbit' object has no attribute 'owner'
>>> rabbit.eatFeed()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Rabbit' object has no attribute 'eatFeed'
>>> h_rabbit.eatFeed()
집토끼가 모이를 먹습니다
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

```
python3
>>> h_rabbit.goto(10, 20)
>>> print(h_rabbit.xPos, ',', h_rabbit.yPos)
10 , 20
>>>
>>> h_rabbit.eatWildglass()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'HouseRabbit' object has no attribute 'eatWildglass'
>>> h_rabbit.eatWildglass()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'HouseRabbit' object has no attribute 'eatWildglass'
>>> m_rabbit.eatWildglass()
산토끼가 들풀을 먹습니다
>>> print(m_rabbit.xPos, ',', m_rabbit.yPos)
0 , 0
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

## [6] 10-3. 15 쪽

```
python3
>>> class Rabbit:
...     shape = ''
...     def __init__(self, shape):
...         print('Rabbit.__init__')
...         self.shape = shape
...
>>> class HouseRabbit(Rabbit):
...     owner = ''
...     def __init__(self, shape, owner):
...         print('HouseRabbit.__init__')
...         super().__init__(shape)
...         self.owner = owner
...
>>> rabbit = Rabbit('토끼')
Rabbit.__init__
>>> h_rabbit = HouseRabbit('집토끼', '홍길동')
HouseRabbit.__init__
Rabbit.__init__
>>> h_rabbit.shape
'집토끼'
>>> h_rabbit.owner
'홍길동'
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

[7] 10-3. 18 쪽

```
python3
>>> class Rabbit:
...     def eat(self):
...         print('토끼가 밥을 먹어요')
...
>>> class HouseRabbit(Rabbit):
...     def eat(self):
...         super().eat()
...         print('토끼밥은 당근이에요')
...
>>> h_rabbit = HouseRabbit()
>>> h_rabbit.eat()
토끼가 밥을 먹어요
토끼밥은 당근이에요
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

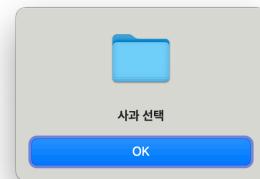
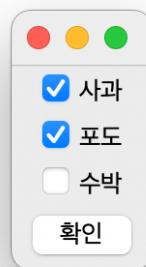
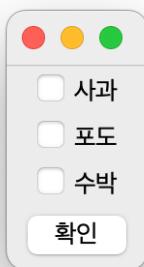
[8] 10-3. 20 21 쪽

```
python3
>>> class Rabbit:
...     def eat(self):
...         print('토끼가 밥을 먹어요')
...
>>> class HouseRabbit(Rabbit):
...     owner = '홍길동'
...     def eat(self):
...         print('토끼가 당근을 먹어요')
...
>>> class MountainRabbit(Rabbit):
...     mountain = '무등산'
...     def eat(self):
...         print('토끼가 풀을 먹어요')
...
>>> class RabbitParty(HouseRabbit, MountainRabbit):
...     def party(self):
...         print('토끼파티를 열었어요')
...
>>> p = RabbitParty()
>>> print(p.owner, ',', p.mountain)
홍길동, 무등산
>>> p.party()
토끼파티를 열었어요
>>> p.eat()
토끼가 당근을 먹어요
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

```
python3
>>> class RabbitParty(HouseRabbit, MountainRabbit):
...     def party(self):
...         print('토끼파티를 열었어요')
...
>>> RabbitParty.mro()
[<class '__main__.RabbitParty'>, <class '__main__.HouseRabbit'>, <class '__main__.MountainRabbit'>, <class '__main__.Rabbit'>, <class 'object'>]
>>>
>>> class RabbitParty(MountainRabbit, HouseRabbit):
...     def party(self):
...         print('토끼파티를 열었어요')
...
>>> RabbitParty.mro()
[<class '__main__.RabbitParty'>, <class '__main__.MountainRabbit'>, <class '__main__.HouseRabbit'>, <class '__main__.Rabbit'>, <class 'object'>]
>>> print('박종현')
박종현
>>> print('학번: 214823')
학번: 214823
>>> 
```

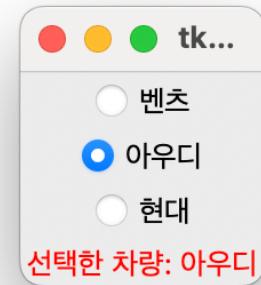
[9] 11-2. 13 14 쪽

```
python3
11% 11 GB 79%
>>> import tkinter as tk
>>> from tkinter import messagebox
>>>
>>> root = tk.Tk()
>>> var1, var2, var3 = tk.IntVar(), tk.IntVar(), tk.IntVar()
>>>
>>> def myapple():
...     if var1.get() == 0:
...         messagebox.showinfo('메시지 박스', '사과 해제')
...     else:
...         messagebox.showinfo('메시지 박스', '사과 선택')
...
>>> cb1 = tk.Checkbutton(root, text='사과', variable=var1, command=myapple)
>>> cb2 = tk.Checkbutton(root, text='포도', variable=var2)
>>> cb3 = tk.Checkbutton(root, text='수박', variable=var3)
>>>
>>> def myfunc():
...     print(var1.get(), var2.get(), var3.get())
...
>>> button1 = tk.Button(root, text='확인', command=myfunc)
>>>
>>> cb1.pack()
>>> cb2.pack()
>>> cb3.pack()
>>> button1.pack()
>>> root.mainloop()
1 0 0
0 1 1
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>> █
```



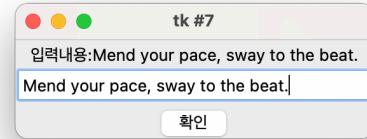
## [10] 11-2. 16 17 쪽

```
python3
>>> import tkinter as tk
>>> root = tk.Tk()
>>> label1 = tk.Label(root, text='선택한 차량 : ', fg='red')
>>>
>>> def myfunc():
...     if myvar.get() == 1:
...         label1.configure(text='선택한 차량 : 벤츠')
...     elif myvar.get() == 2:
...         label1.configure(text='선택한 차량 : 아우디')
...     else:
...         label1.configure(text='선택한 차량 : 현대')
...
>>> myvar = tk.IntVar()
>>> rb1 = tk.Radiobutton(root, text='벤츠', variable=myvar, value=1, command=myfunc)
>>> rb2 = tk.Radiobutton(root, text='아우디', variable=myvar, value=2, command=myfunc)
>>> rb3 = tk.Radiobutton(root, text='현대', variable=myvar, value=3, command=myfunc)
>>>
>>> rb1.pack()
>>> rb2.pack()
>>> rb3.pack()
>>> label1.pack()
>>> root.mainloop()
>>> print('학번: 214823')
학번: 214823
>>> 
```



## [11] 11-2. 20 쪽

```
python3
>>> import tkinter as tk
>>> root = tk.Tk()
>>> label1 = tk.Label(root, text='입력 내용 :')
>>> myEntry = tk.Entry(root, width=30)
>>>
>>> def myfunc():
...     in_text = '입력 내용 :' + myEntry.get()
...     label1.configure(text=in_text)
...
>>> bt1 = tk.Button(root, text='확인', command=myfunc)
>>>
>>> label1.pack()
>>> myEntry.pack()
>>> bt1.pack()
>>> root.mainloop()
>>> print('학번: 214823')
학번: 214823
>>> 
```



[12] 11-2. 25 30 쪽

```
python3
12% 10 GB 79%
>>> import tkinter as tk
>>> root = tk.Tk()
>>> root.geometry('300x300')
''

>>> root.resizable(width=False, height=False)
''

>>> label1 = tk.Label(root, text='마우스 이벤트')
''

>>> def clickMouse(event):
...     if event.num == 1:
...         inText = '왼쪽 버튼: (' + str(event.x) + ',' + str(event.y) + ')'
...     elif event.num == 3:
...         inText = '오른쪽 버튼: (' + str(event.x) + ',' + str(event.y) + ')'
...     else:
...         inText = '가운데 버튼: (' + str(event.x) + ',' + str(event.y) + ')'
...     label1.configure(text=inText)

''

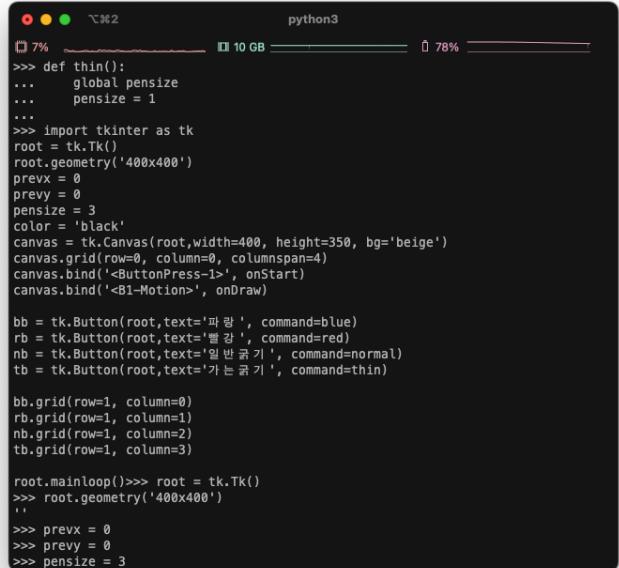
>>> label1.pack(expand=1, anchor=tk.CENTER)
''

>>> root.bind('<Button>', clickMouse)
'4319121984clickMouse'
''

>>> root.mainloop()
>>> print('辈종현')
辈종현
>>> print('학번: 214823')
학번: 214823
>>>
```



```
 11% 10 GB 78%  
python3  
=> def onStart(event):  
...     global prevx, prevy  
...     prevx = event.x  
...     prevy = event.y  
...  
>>> def onDraw(event):  
...     global prevx, prevy  
...     global canvas, pensize, color  
...     curx = event.x  
...     cury = event.y  
...     canvas.create_line(prevx, prevy, curx, cury, width=pensize, fill=color)  
...     prevx = curx  
...     prevy = cury  
...  
>>> def blue():  
...     global color  
...     color = 'blue'  
...  
>>> def red():  
...     global color  
...     color = 'red'  
...  
>>> def normal():  
...     global pensize  
...     pensize = 3  
...  
>>> def thin():  
...     global pensize  
...     pensize = 1  
...  
>>> import tkinter as tk  
root = tk.Tk()
```



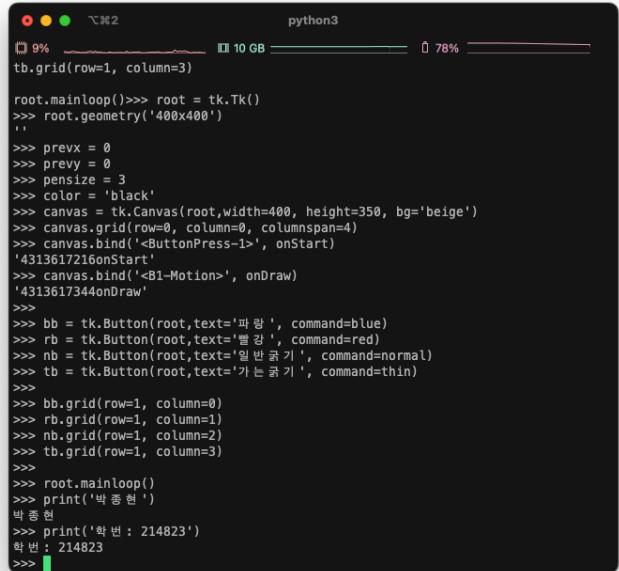
```
python3
root.geometry('400x400')
prevx = 0
prevy = 0
pensize = 3
color = 'black'
canvas = tk.Canvas(root,width=400, height=350, bg='beige')
canvas.grid(row=0, column=0, columnspan=4)
canvas.bind('<ButtonPress-1>', onStart)
canvas.bind('<B1-Motion>', onDraw)

bb = tk.Button(root,text='파랑 ', command=blue)
rb = tk.Button(root,text='빨강 ', command=red)
nb = tk.Button(root,text='일반 굵기 ', command=normal)
tb = tk.Button(root,text='가는 굵기 ', command=thin)

bb.grid(row=1, column=0)
rb.grid(row=1, column=1)
nb.grid(row=1, column=2)
tb.grid(row=1, column=3)

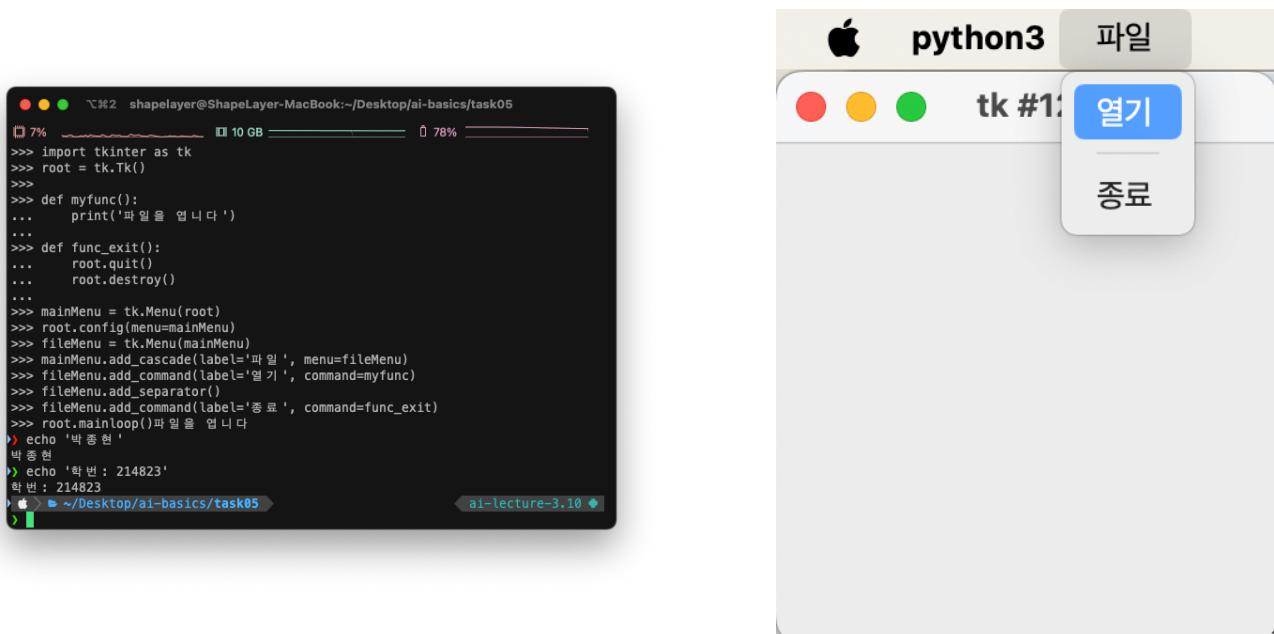
root.mainloop()>>> root = tk.TK()
>>> root.geometry('400x400')
''

>>> prevx = 0
>>> prevy = 0
>>> pensize = 3
>>> color = 'black'
>>> canvas = tk.Canvas(root,width=400, height=350, bg='beige')
>>> canvas.grid(row=0, column=0, columnspan=4)
>>> canvas.bind('<ButtonPress-1>', onStart)
'4313617216onStart'
>>> canvas.bind('<B1-Motion>', onDraw)
```





[13] 11-3. 7 쪽



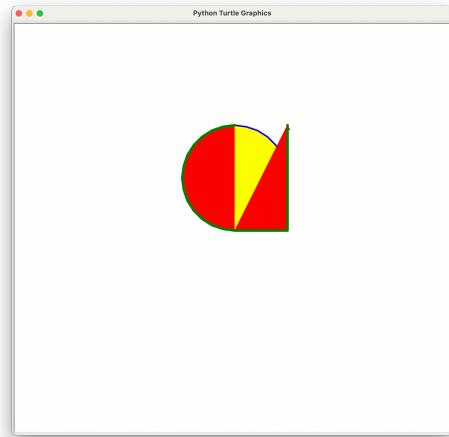
[14] 11-3. 11 12 쪽

```
 12% 10 GB 77%
>>> import tkinter as tk
>>> from tkinter import simpledialog
>>> root = tk.TK()
>>> root.geometry('200x200')
...
>>> label1 = tk.Label(root, text='입력 값 :')
>>>
>>> def myfunc():
...     value = simpledialog.askstring('문자열 입력', '문자열을 입력하세요')
...     label1.configure(text='입력 문자열 : ' + value)
...
>>> def myfunc_sec():
...     value = simpledialog.askstring('문자열 입력', '문자열을 입력하세요', show='*')
...     label1.configure(text='입력 문자열 : ' + value)
...
>>> bt1 = tk.Button(root, text='입력 받기', command=myfunc)
>>> bt2 = tk.Button(root, text='암호 입력 받기', command=myfunc_sec)
>>> label1.pack(expand=1, anchor=tk.CENTER)
>>>
>>> bt1.pack()
>>> bt2.pack()
>>> root.mainloop()
2024-07-16 18:09:21.640 python3[25456:338607] TSM AdjustCapsLockLEDForKeyTransitionHandling - _ISSetPhysicalKeyboardCapsLockLED Inhibit
> echo '박종현'
박종현
> echo '학번: 214823'
학번: 214823
ai-lecture-3.10 ◆
```



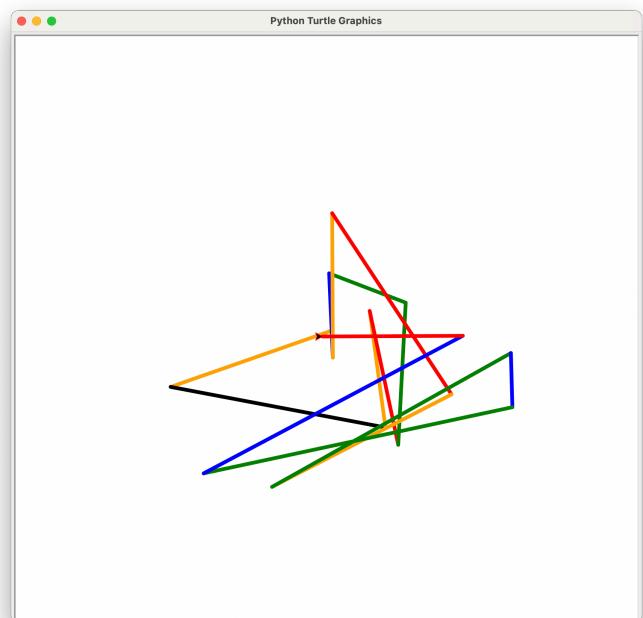
[15] 11-3. 22 쪽

```
python3
11% 10 GB 77%
>>> import turtle as t
>>>
>>> t.reset()
>>> t.pd()
>>> t.pensize(3)
>>> t.pencolor('blue')
>>> t.fillcolor('yellow')
>>> t.begin_fill()
>>> t.circle(100, 180)
>>> t.end_fill()
>>> t.pensize(5)
>>> t.pencolor('green')
>>> t.fillcolor('red')
>>> t.begin_fill()
>>> t.circle(100, 180)
>>> t.end_fill()
>>> t.begin_fill()
>>> t.fd(100)
>>> t.lt(90)
>>> t.fd(200)
>>> t.end_fill()
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>>
```



[16] 11-3. 27 31 쪽

```
python3
12% 10 GB 77%
>>> import turtle as t
>>> import random
>>>
>>> colorList = ['red', 'green', 'blue', 'black', 'orange']
>>> tMove = []
>>>
>>> def randomDraw():
...     global tMove
...     tMove.clear()
...     t.reset()
...     t.pensize()
...     for i in range(20):
...         x = random.randint(-250, 250)
...         y = random.randint(-250, 250)
...         color = random.choice(colorList)
...         tMove.append((x, y, color))
...         t.pencolor(color)
...         t.goto(x, y)
...
>>> def reDraw():
...     global tMove
...     t.reset()
...     t.pensize(5)
...     for m in tMove:
...         x = m[0]
...         y = m[1]
...         color = m[2]
...         t.pencolor(color)
...         t.goto(x, y)
...
>>> t.setup(550, 550)
>>> t.screensize(500, 500)
>>> randomDraw()
>>> reDraw()
>>> reDraw()
>>> randomDraw()
>>> reDraw()
>>> t.bye()
>>> print('박종현')
박종현
>>> print('학번 : 214823')
학번 : 214823
>>>
```



```
python3
9% 10 GB 77%
>>> import turtle as t
>>> import random
>>> colorList = ['red', 'green', 'blue', 'black', 'orange']
>>> tMove = []
>>>
>>> def randomDraw(_1, _2):
...     global tMove
...     tMove.clear()
...     t.reset()
...     t.pensize()
...     for i in range(20):
...         x = random.randint(-250, 250)
...         y = random.randint(-250, 250)
...         color = random.choice(colorList)
...         tMove.append((x, y, color))
...         t.pencolor(color)
...         t.goto(x, y)
...
>>> def reDraw():
...     global tMove
...     t.reset()
...     t.pensize(5)
...     for m in tMove:
...         x = m[0]
...         y = m[1]
...         color = m[2]
...         t.pencolor(color)
...         t.goto(x, y)
...
>>> t.setup(550, 550)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<string>", line 5, in setup
  turtle.Terminator
>>> t.onclick(randomDraw)
>>> t.onclick(reDraw, btn=3)
>>> t.mainloop()
>>> print('작성 완')
작성 완
>>> print('학번: 214823')
학번: 214823
>>> 
```

