Python GUI tkinter

root နှင့် Canvas

root

tkinter package ကို သုံး၍ အောက်တွင် ဖော်ပြထားသည့် code (၃)ကြောင်းဖြင့် window တစ်ခုကို စတင်တည်ဆောက်သည်။

```
import tkinter as tk
root = tk.Tk()

root.mainloop()
```

root = tk.Tk() နှင့် root.mainloop() လိုင်းနှစ်လိုင်းအကြားတွင်သာ tkinter ၏ button , label စသည့် widget တို့ကို ထည့်ရသည်။

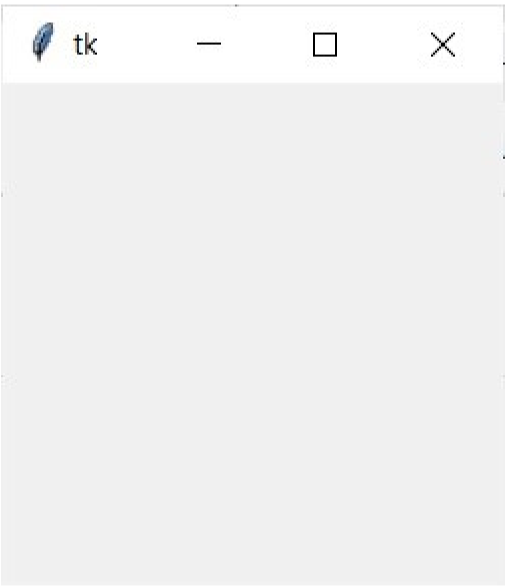
Frame, Label, Button, Checkbutton, Radiobutton, Entry Combobox စသည်တို့သည့် Basic Widgets ဖြစ်ကြသည်။ Widget များအားလုံးသည် python obejct များ ဖြစ်ကြသည်။ Widget များအားလုံးသည် root ဟု နာမည်ပေးသည့် tk.Tk() ပေါ်တွင် တည်ဆောက်နိုင်သည်။

In [1]:
```
# tkinter ကို အသုံးပြုရန် import လုပ်သည်။
import tkinter as tk

# root = tk.Tk() နှင် root.mainloop() တို့သည် tkinter ၏ အစနှင့် အဆုံး သို့
မဟုတ် mainloop() ဖြစ်သည်။
root = tk.Tk()
root.mainloop()
```

code (၃)ကြောင်းဖြင့် တည်ဆောက်လိုက်သည့် window တစ်ခုကို အောက်တွင် တွေ့မြင်နိုင်သည်။

In [2]:
```
# အောက်မှ code သည် jupyter ipython notebook တွင် ပုံများကို ဖော်ပြရန်အတွက်
ဖြစ်သည်။
from IPython.display import Image
Image(filename="root.jpg", )
```

Out[2]:



ထို့ code (၃)ကြောင်းသည် အပေါ် ပုံတွင် ဖော်ပြထားသည့် window တစ်ခုကို တည်ဆောက်ပေးသည်။ root = tk.Tk() တွင် root ဆိုသည့် နာမည် မသုံးဘဲ တခြားမိမိကြိုက်နှစ်သက်သည့် နာမည်ကို အသုံးပြုနိုင်သည်။ ဥပမာ-

In [3]:
```
import tkinter as tk
window= tk.Tk()

window.mainloop()
```

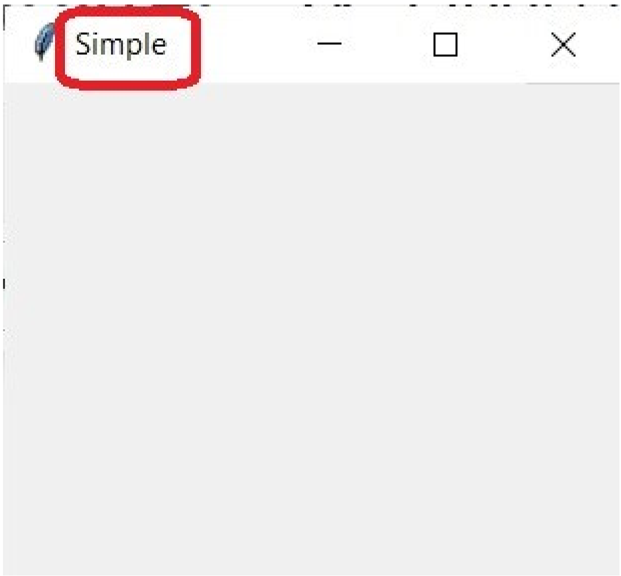
root.title() ဖြင့် window title ထည့်ခြင်း

root.title() ဖြင့် မိမိကြိုက်နှစ်သက်သည့် အမည်ထည့်သွင်းနိုင်သည်။

In [4]:
```
import tkinter as tk

root = tk.Tk()
root.title("Simple") # Simple title

root.mainloop()
```

In [5]:
```
from IPython.display import Image
Image(filename="title.jpg")
```

Out[5]:



In [6]:
```
type(root)
```

Out[6]: tkinter.Tk

root သည် tkinter တွင် အခြေခံအကျဆုံးသော အရာဖြစ်သည်။ ပန်းချီကားဘောင်နှင့်တူသည်။ ပန်းချီကားဘောင်နှင့် ကင်းဗတ်စပေါ်တွင်သာ မိမိ အလိုရှိသည့်ပုံများ ရေးဆွဲနိုင်သည်။ ထို့အတူ root ပေါ်တွင် မိမိကြိုက်နှစ်သက်သည့် widget များ ထည့်နိုင်သည်။ ကင်းဗတ်စ(Canvas) အရွယ်အစားသတ်မှတ်ပုံကို အောက်တွင်ဖော်ပြထားသည်။

Canvas

```
canvas = tk.Canvas(root, width = 900, height=400)
```

သည် root ပေါ်တွင် width = 900 နှင့် height=400 အရွယ်အစားရှိသည့် ကင်းဗတ်စ(Canvas)တစ်ခုတည်ဆောက်လိုက်ခြင်း ဖြစ်သည်။

In [7]:
```
import tkinter as tk

root = tk.Tk()
root.title("Simple")
my_canvas = tk.Canvas(root, width = 900, height=400)

root.mainloop()
```

my_canvas သည် tkinter မှ Canvas ဆိုသည့် python object တစ်ခု ဖြစ်သည်။ widget ဟုလည်းခေါ်သည်။

In [8]:
```
type(my_canvas)   # my_canvas အမျိုးအစားကို ကြည့်ရန်
```

Out[8]: tkinter.Canvas

In [9]:
```
id(my_canvas)  # my_canvas ID ကို ကြည့်ရန်
```

Out[9]: 1319392946056

help(my_canvas) ဖြင့် canvas နှင့် သက်ဆိုင်သည့် အချက်အလက်များကို ဖတ်နိုင်သည်။

ကင်းဗတ်စ(Canvas) တစ်ခုတည်ဆောက်နေစဉ် သို့မဟုတ် တည်ဆောက်ပြီးနောက်ပိုင်း မိမိအလိုရှိသည့် အရွယ်အစား(height နှင့် width)ကိုလည်းကောင်း background color, Border ကို ကိုလည်းကောင်းသတ်မှတ်ပေးနိုင်သည်။ ထို အရာများကို option ဟုခေါ်သည်။ widget တိုင်းတွင် option များစွာ ရှိသည်။

widget များထည့်သည့်အခါ သတိပြုရန်အချက်မှာ (၁) option နှင့် (၂) widget များကို မိမိ လိုချင်သည့်နေရာတွင် မှန်ကန်စွာ လွယ်ကူစွာ နေရာချနိုင်ခြင်းသည်။

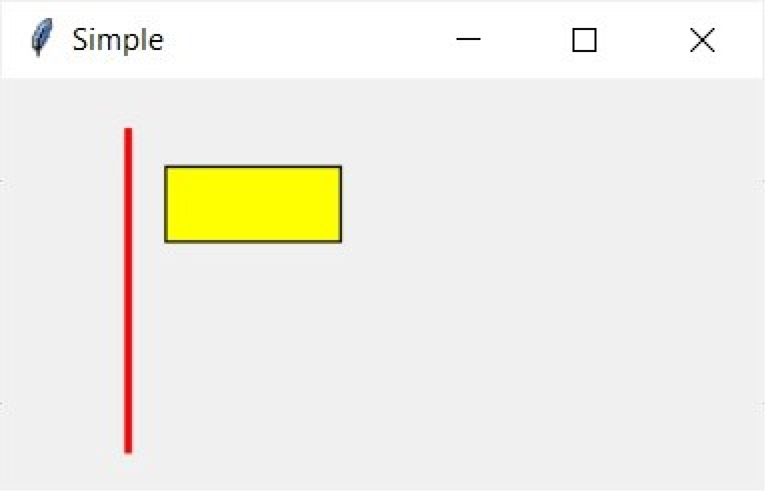
အောက်တွင် တည်ဆောက်ပြီးသည့် ကင်းဗတ်စ(Canvas) ပေါ်တွင် rectangle တစ်ခု နှင့် လိုင်း တစ်လိုင်းဆွဲပုံကိုဖော်ပြထားသည်။ rectangle တစ်ခု ရှိနေမည့် နေရာ(coordinate) နှင့် အရောင်(color) ဖြည့်လိုပါက ဖြည့်မည့် အရောင်ထည့်ပေးရသည်။ နှင့် လိုင်း ၏ coordinate နှင့် လိုင်းအရောင်ထည့် ပေးရသည်။ width=3 သည် လိုင်း အထူအတွက် ဖြစ်သည်။

ကင်းဗတ်စ(Canvas) ပေါ်တွင် လိုင်းဆွဲသည့်အခါ ထည့်ပေးရမည့် အမှတ်များမှာ .create_line(x0, y0, x1, y1, ...)

In [10]:
```
import tkinter as tk

root = tk.Tk()
root.title("Simple")
my_canvas = tk.Canvas(root, width = 300, height=160)
my_canvas.pack() # .pack() ဖြင့် နေရာချခြင်း

# rectangle တစ်ခု ဆွဲခြင်း
my_canvas.create_rectangle(65, 35, 135, 65, fill="yellow")

# လိုင်း တစ်လိုင်း ဆွဲခြင်း
my_canvas.create_line(50, 20, 50, 150, fill="red", width=3)

root.mainloop()
```

In [11]:
```
from IPython.display import Image
Image(filename="rectangle_and_line.jpg")
```

Out[11]:



root တည်ဆောက်ခြင်း, ကင်းဗတ်စ(Canvas)တည်ဆောက်ခြင်းနှင့် ကင်းဗတ်စ(Canvas) ပေါ်တွင် rectangle တစ်ခုနှင့် လိုင်း တစ်လိုင်းဆွဲပုံကို အကြမ်းရှင်းပြပြီး ဖြစ်သည်။

Code များကို ကူးယူ၍ မိမိ စိတ်ကူးရသလို အမျိုးမျိုးပြင်၍ လေ့ကျင့်နိုင်ပါသည်။

ကင်းဗတ်စ(Canvas) ပေါ်တွင် အကွက်များ သတ်မှတ်ခြင်း

canvas.grid()ဖြင့် ကင်းဗတ်စ(Canvas) ပေါ်တွင် အကွက်များ သတ်မှတ်နိုင်သည်။ မိမိ အလိုရှိသည့် အကွက်(အပိုင်း) အရေအတွက်ကို columnspan, rowspan ဖြင့် သတ်မှတ်ပေးနိုင်သည်။ columnspan=2, rowspan=2 သည် ကင်းဗတ်စ(Canvas)ကို အပိုင်း (၄)ပိုင်းဖြစ်အောက် သတ်မှတ်လိုက်ခြင်း ဖြစ်သည်။

ရှင်းလင်းစွာ မြင်နိုင်ရန်အတွက် လိုင်းများ ဆွဲကြည့်မည်။

အလယ်ကနေ ဘေးတိုက် ပိုင်းသည့် မျဉ်းကို အနီရောင် ဖြင့် ဆွဲမည်။ အနီရောင် လိုင်း၏ coordinate အမှတ်များမှာ (x1 = 0, y1 = 75, x2 = 300, y2 = 75)

```
my_canvas.create_line(0, 75, 300, 75, fill="red", width=3)
```

အလယ်ကနေ အထက်အောင် ပိုင်းသည့် မျဉ်းကို အပြာရောင် ဖြင့် ဆွဲမည်။ အပြာရောင်လိုင်း၏ coordinate အမှတ်များမှာ
(x1 = 150, y1 = 0, x2 = 150, y2 = 300)

```
my_canvas.create_line(150, 0, 150, 300, fill="blue", width=3)
```

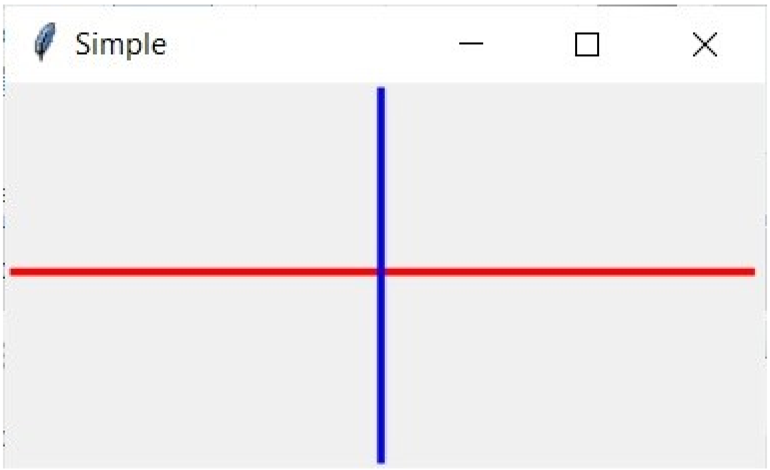
In [12]:

```
import tkinter as tk

root = tk.Tk()
root.title("Simple")
my_canvas = tk.Canvas(root, width = 300, height=150)
my_canvas.grid(columnspan=2, rowspan=2)

my_canvas.create_line(0, 75, 300, 75, fill="red", width=3)
my_canvas.create_line(150, 0, 150, 300, fill="blue", width=3)

root.mainloop()
```

In [13]:

```
from IPython.display import Image
Image(filename="columnspan_rowspan.jpg")
```

Out[13]:



အောက်တွင် for loop ဖြင့် အလျားလိုက်လိုင်းများ၏ coordinate အမှတ်များကို ပေး၍ ဆွဲပြထားသည်။ y point နှင့် သက်ဆိုင်သည့်အမှတ်များကို ပြောင်းပေးရသည်။

In [14]:

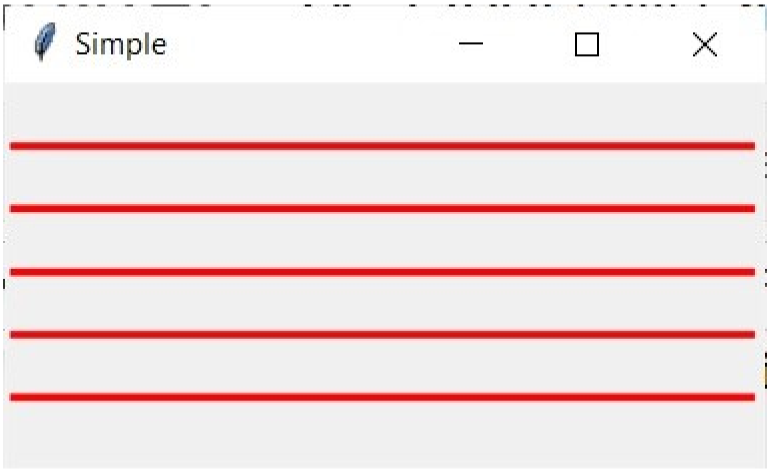
```
import tkinter as tk

root = tk.Tk()
root.title("Simple")
my_canvas = tk.Canvas(root, width = 300, height=150)
my_canvas.grid(columnspan=1, rowspan=6)

for i in range(0,150,25):
    my_canvas.create_line(0, i, 300, i, fill="red", width=3)
    print(f"(x1 = 150, y1 = {i}, x2 = 150, y2 = {i})" )

root.mainloop()
```

```
(x1 = 150, y1 = 0, x2 = 150, y2 = 0)
(x1 = 150, y1 = 25, x2 = 150, y2 = 25)
(x1 = 150, y1 = 50, x2 = 150, y2 = 50)
(x1 = 150, y1 = 75, x2 = 150, y2 = 75)
(x1 = 150, y1 = 100, x2 = 150, y2 = 100)
(x1 = 150, y1 = 125, x2 = 150, y2 = 125)
```

In [15]:

```
from IPython.display import Image
Image(filename="horizontal_line.jpg")
```

Out[15]:



Canvas နှင့် သက်ဆိုင်သည့် အရာများကို ဆက်လက် လေ့လာပါ။

1. Canvas coordinates scrollbars ကို သုံး၍ canvas အရွယ်အစားသည် window အရွယ်အစားထက် ပိုကြီးအောင် လုပ်နိုင်သည်။ window coordinate နှင့် canvas coordinate ဟူ၍ (၂)မျိုးရှိသည်။
2. The Canvas display list
3. Canvas object IDs
4. Canvas tags
5. Canvas tagOrId arguments
6. Methods on Canvas widgets
7. Canvas arc objects
8. Canvas bitmap objects
9. Canvas image objects
10. Canvas line objects
11. Canvas oval objects
12. Canvas polygon objects
13. Canvas rectangle objects
14. Canvas text objects
15. Canvas window objects

End-
Python GUI tkinter မှ root နှင့် Canvas

```
In [ ]:
```