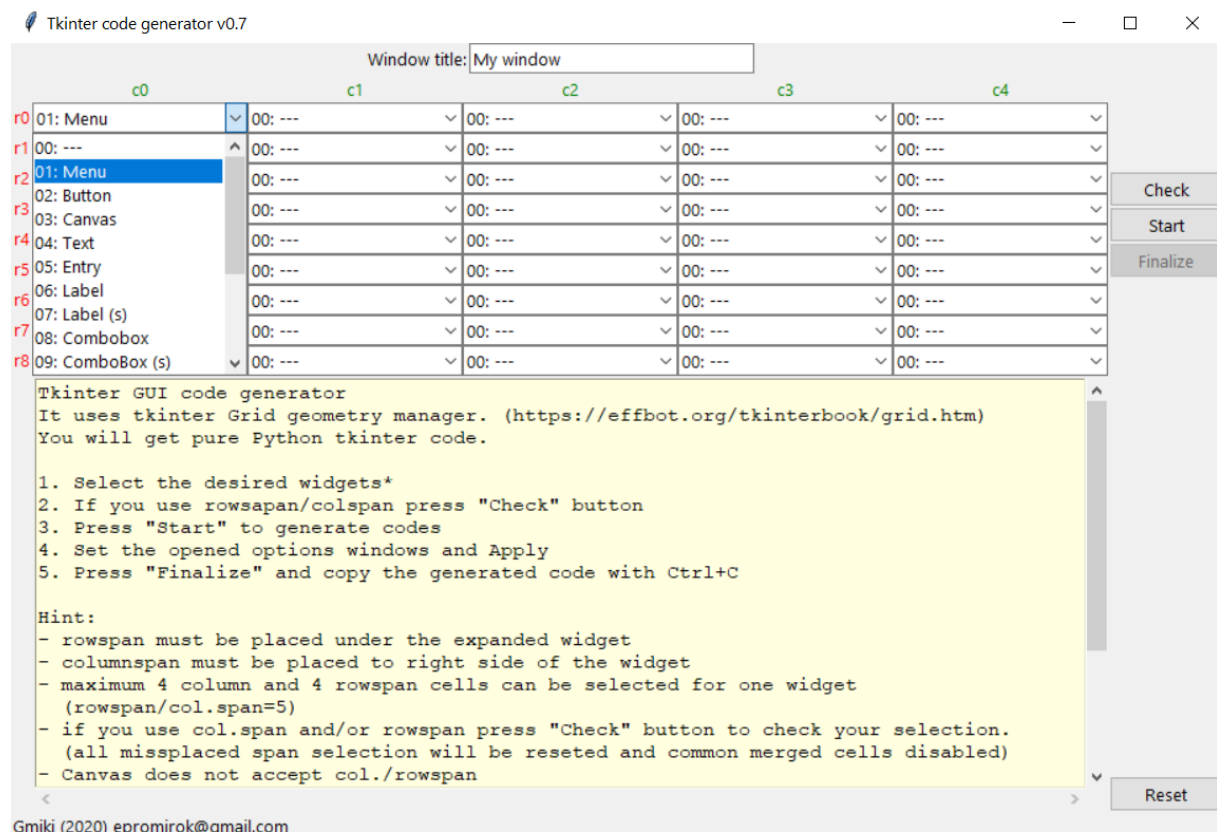


## Tkinter GUI designer

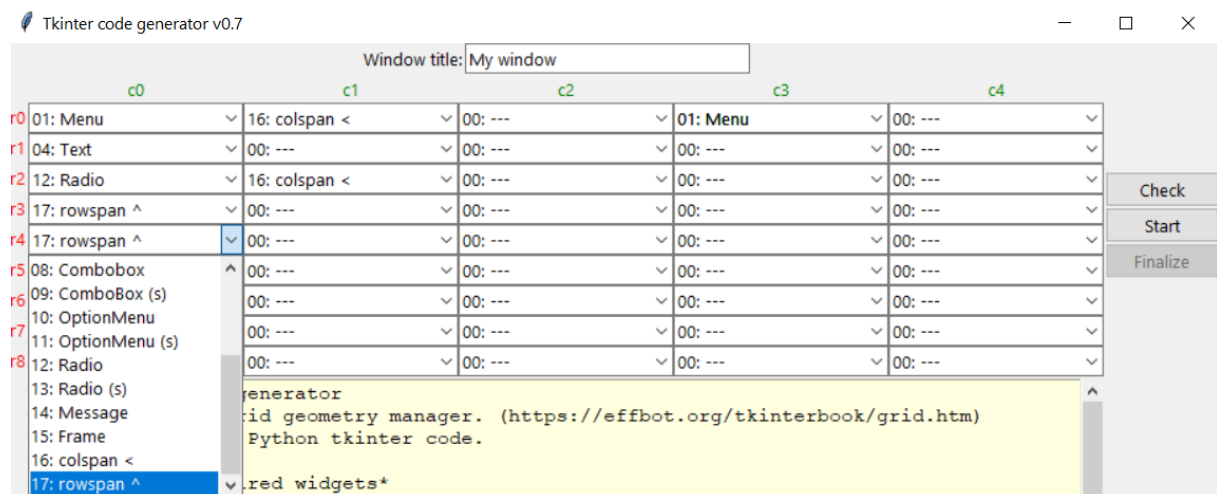
It is written in Python and you will get pure Python tkinter code.

It uses tkinter Grid geometry manager. (<https://effbot.org/tkinterbook/grid.htm>)

### 1. Select the desired widgets\*



### 2. If you use rowspan/colspan press "Check" button



Tkinter code generator v0.7

Window title:

	c0	c1	c2	c3	c4
r0	01: Menu	16: colspan <	00: ---	01: Menu	00: ---
r1	04: Text	00: ---	00: ---	00: ---	00: ---
r2	12: Radio	16: colspan <	00: ---	00: ---	00: ---
r3	17: rowspan ^	00: ---	00: ---	00: ---	00: ---
r4	17: rowspan ^	00: ---	00: ---	00: ---	00: ---
r5	00: ---	00: ---	00: ---	00: ---	00: ---
r6	00: ---	00: ---	00: ---	00: ---	00: ---
r7	00: ---	00: ---	00: ---	00: ---	00: ---
r8	00: ---	00: ---	00: ---	00: ---	00: ---

Buttons: Check, Start, Finalize

- All common merged cells will be disabled (for example c1-r3, c1-r4 on screenshot)
- Rowspan must be placed under the expanded widget
- Columnspan must be placed to right side of the widget
- Maximum 4 column and 4 rowspan cells can be selected for one widget (rowspan/colspan=5)
- If you use colspan and/or rowspan press "Check" button to check your selection.
- All misplaced span selection will be reseted
- Canvas and Menubar does not accept col./rowspan

### 3. Press "Start" to generate codes

Related option widows open

Radio options: c0, r2

Text options: c0, r1

Menu options: c3, r0

Menu options: c0, r0

### Menubar

Menubar options: c0, r0 (m)

Mainmenu (1st row) -> File Settings Tools About homepage info

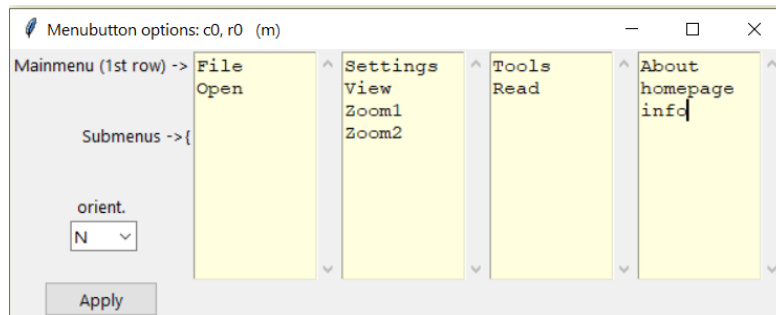
Submenus -> { Open Save Close View Zoom1 Zoom2 Read }

orient. N

Apply

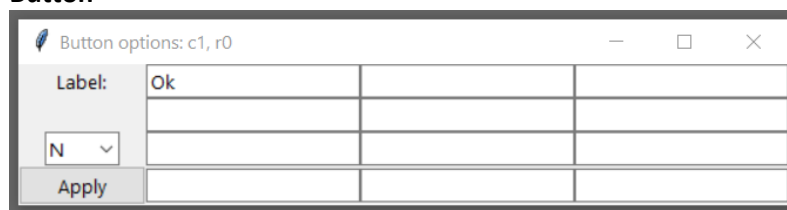
You can put 4 main menu (top blue line) and any submenu  
 First row: main menu (x4), other rows: submenu (any)  
 Available settings: separate: '-' ; checkbox: '#' (eg: #yes)

## MenuButton



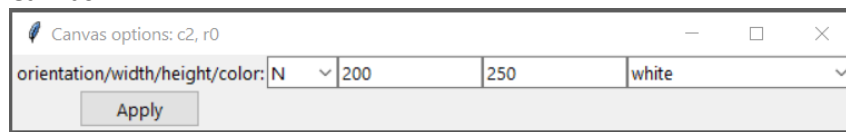
You can put 4 main menu (top row) and any submenu  
Available settings: orientation.

## Button



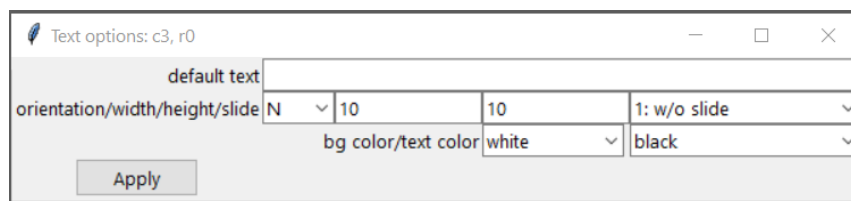
You can put 3x4 button in a frame. (min 1 button max 12button)  
Available settings: orientation. N- North, E-East, S-South, W-West, NE,SE,SW,NW, N+S (expand between max N and max S), E+W (expand between max W and max E)

## Canvas



Available settings: orientation, width, height and color  
You can select some color, select 'pick a color' to open color selector or write any standard python color code here.

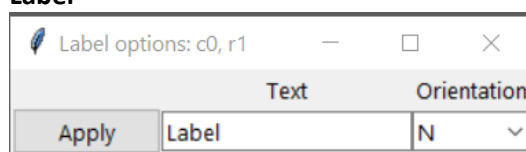
## Text



Available settings: default text, orientation, width, height, slide (without, with horizontal, with vertical or both slide) and background and text color

You can select some color, select 'pick a color' to open color selector or write any standard python color code here.

## Label



Available settings: default text and orientation

## Combobox

You can place up to 4 comboboxes with 3 options

Available settings:

Orientation, Cell width, Label (for each combobox), 3 options (for each combobox)

## Option menu

You can place up to 4 option menus with 3 options

Available settings:

Orientation, Label (for each combobox), 3 options (for each combobox)

## Radio

You can place up to 4 radio with 3 options

Available settings: Orientation, 3 options (for each combobox)

## Universal Frame

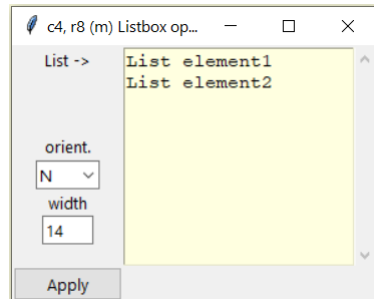
You can place any widget\* (up to 4)

\*menu (3x3), button (4x3), canvas, text with scroll, entry, label, combobox (4x3), optionmenu (4x3),radio (4x3)

Available settings:

Orientation, 4 widget selection, label text (with text: LabelFrame / without: Frame)

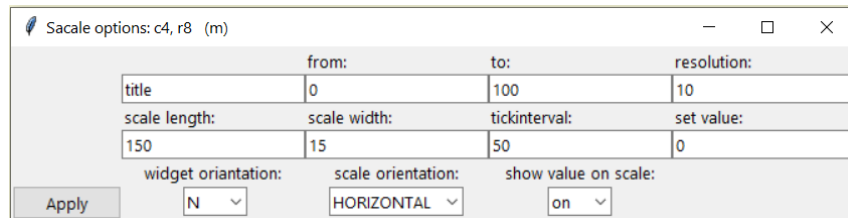
### Listbox



You can place any line into the list

Available settings: Orientation, width

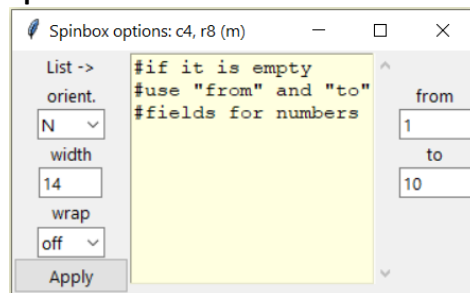
### Scale



Available settings:

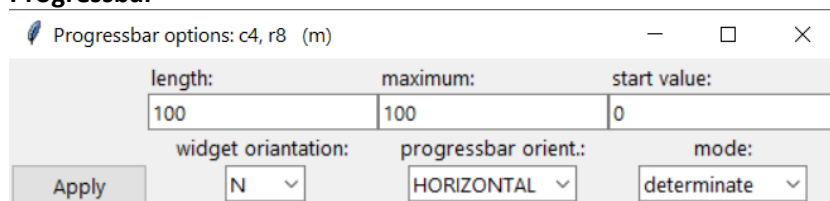
scale label, from, to, resolution, scale length, scale width, thickinterval, set value, widget orientation, scale orientation, show value

### Spinbox



Available settings: from, to, wrap, list items, width, orientation

### Progressbar



Available settings:

length, maximum, start value, widget orientation, progressbar orientation, mode

### Toolbar

Toolbar options: c4, r8 (m)

	1	2	3	4	5
image name:	addressbook	open	save	---	---
tooltip text:	Click here	Open folder	Save your file		

orient: N

Available settings:

PNG image select, Tooltip text, orientation, hint



## 4. Set the opened options windows and Apply

Tkinter code generator v0.7

Window title: My window

	c0	c1	c2	c3	c4
r0	01: Menu	02: Button	03: Canvas	04: Text	05: Entry
r1	06: Label	08: Combobox	10: OptionMenu	12: Radio	00: ---
r2	00: ---	00: ---	00: ---	00: ---	00: ---
r3	00: ---	00: ---	00: ---	00: ---	00: ---
r4	00: ---	00: ---	00: ---	00: ---	00: ---
r5	00: ---	00: ---	00: ---	00: ---	00: ---
r6	00: ---	00: ---	00: ---	00: ---	00: ---
r7	00: ---	00: ---	00: ---	00: ---	00: ---
r8	00: ---	00: ---	00: ---	00: ---	00: ---

```

#-----Entry: c4, r0-----
bmezo040=Entry(ablak)
bmezo040.grid(row=0, column=4, columnspan=1, rowspan=1, sticky=N)
#-----Button: c1, r0-----
kret01=Frame(ablak, relief='flat', borderwidth=1)
kret01.grid(row=0, column=1, columnspan=1, rowspan=1, sticky=N)
gmb010=Button(kret01, text='Ok', command=k_ablak.destroy)
gmb010.grid(row=0, column=0)
#-----Menu: c0, r0-----
kret00=Frame(ablak, relief='raised', borderwidth=1)
kret00.grid(row=0, column=0, columnspan=1, rowspan=1, sticky=N)
menu000=Menubutton(kret00, text='File')
mf000=Menu(menu000)
mf000.add_command(label='Close', command=k_ablak.destroy, state=NORMAL)
menu000.configure(menu=mf000)
menu000.grid(row=0, column=0, sticky=NW)
#-----Canvas: c2, r0-----
can020=Canvas(ablak, bg='white', height=250, width=200)
  
```

Your code is not done yet, you have to finalize it.

## 5. Press "Finalize" and copy the generated code with Ctrl+C

### Hint:

- rowspan must be placed under the expanded widget
- columnspan must be placed to right side of the widget
- maximum 4 column and 4 rowspan cells can be selected for one widget (rowspan/col.span=5)
- if you use columnspan and/or rowspan press "Check" button to check your selection.
  - all misplaced span selection will be reseted and
  - common merged cells disabled
- Canvas and Menubar does not accept column/rowspan
- Open second session of Python IDLE and use it for checking the generated code without closing the running GUI generator

v0.9.2

- widget name with (s) like 'Label (s)', 'Combobox (s)', 'Optionmenu (s)' and 'Radio (s)' means simple. It uses default parameters and does not open option window.
- Toolbar PNG files must be in ico folder. For example: If path of your file is c:\myprog\myprog.py, PNG files location is c:\myprog\ico. Sample PNG files in the package.

### **Supported widgets:**

manubar (4x any), menubutton (4x any), button (4x3), canvas, text with scroll, entry, label, combobox (4x3), optionmenu (4x3), radio (4x3), message, Universal Frame, Listbox, Scale, Spinbox, Toolbar with Tooltip text

*Note: I am not professional Python programmer, but I hope you can use it and it can help you to create Tkinter GUI much more easier.*

*If you have any comment, suggestion write me: [epromiok@gmail.com](mailto:epromiok@gmail.com)*

*[https://github.com/horrorfodrasz/Tkinter\\_GUI\\_designer](https://github.com/horrorfodrasz/Tkinter_GUI_designer)*

**If you like it you can donate me**

Paypal



Thank you!