

# Tkinter GUI code generator

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\*

Tkinter code generator v0.7

Window title:

	c0	c1	c2	c3	c4
r0	01: Menu	00: ---	00: ---	00: ---	00: ---
r1	00: ---	00: ---	00: ---	00: ---	00: ---
r2	01: Menu	00: ---	00: ---	00: ---	00: ---
r3	02: Button	00: ---	00: ---	00: ---	00: ---
r4	03: Canvas	00: ---	00: ---	00: ---	00: ---
r5	04: Text	00: ---	00: ---	00: ---	00: ---
r6	05: Entry	00: ---	00: ---	00: ---	00: ---
r7	06: Label	00: ---	00: ---	00: ---	00: ---
r8	07: Label (s)	00: ---	00: ---	00: ---	00: ---
r9	08: Combobox	00: ---	00: ---	00: ---	00: ---
r10	09: ComboBox (s)	00: ---	00: ---	00: ---	00: ---

Check  
Start  
Finalize

Tkinter GUI code generator  
It uses tkinter Grid geometry manager. (<https://effbot.org/tkinterbook/grid.htm>)  
You will get pure Python tkinter code.

1. Select the desired widgets\*
2. If you use rowspan/colspan press "Check" button
3. Press "Start" to generate codes
4. Set the opened options windows and Apply
5. Press "Finalize" and copy the generated code with Ctrl+C

Hint:

- rowspan must be placed under the expanded widget
- colspan 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 col.span and/or rowspan press "Check" button to check your selection. (all misplaced span selection will be reseted and common merged cells disabled)
- Canvas does not accept col./rowspan

Reset

Gmiki (2020) enromirok@gmail.com

## 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	08: Combobox	00: ---	00: ---	00: ---	00: ---
r6	09: ComboBox (s)	00: ---	00: ---	00: ---	00: ---
r7	10: OptionMenu	00: ---	00: ---	00: ---	00: ---
r8	11: OptionMenu (s)	00: ---	00: ---	00: ---	00: ---
r9	12: Radio	00: ---	00: ---	00: ---	00: ---
r10	13: Radio (s)	00: ---	00: ---	00: ---	00: ---
r11	14: Message	00: ---	00: ---	00: ---	00: ---
r12	15: Frame	00: ---	00: ---	00: ---	00: ---
r13	16: colspan <	00: ---	00: ---	00: ---	00: ---
r14	17: rowspan ^	00: ---	00: ---	00: ---	00: ---

Check  
Start  
Finalize

Tkinter GUI code generator  
It uses tkinter Grid geometry manager. (<https://effbot.org/tkinterbook/grid.htm>)  
You will get pure Python tkinter code.

1. Select the desired widgets\*

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: ---

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/col.span=5)
- If you use col.span and/or rowspan press "Check" button to check your selection.
- All misplaced span selection will be reseted
- Canvas does not accept col./rowspan

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

Related option widows open

## Button

Button options: c1, r0

Label:	Ok		
N			
Apply			

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)

## MenuButton

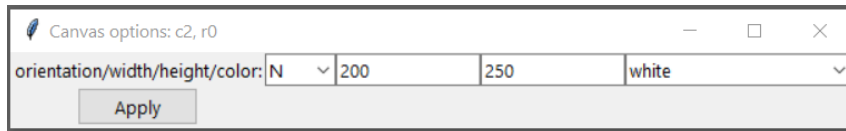
Menu options: c0, r0

Main menu:	File		
Submenu:	Close		
N <input type="button" value="v"/>			
Apply			

You can put 3 main menu (top blue line) and 3\*3 submenu

Available settings: orientation.

## Canvas



Canvas options: c2, r0

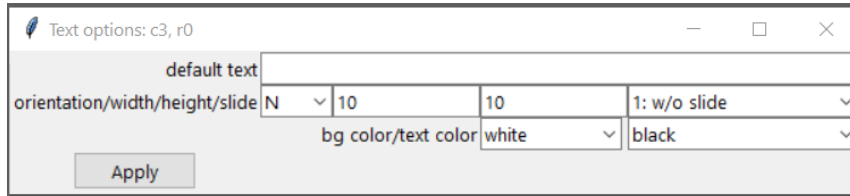
orientation/width/height/color: N 200 250 white

Apply

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



Text options: c3, r0

default text:

orientation/width/height/slide: N 10 10 1: w/o slide

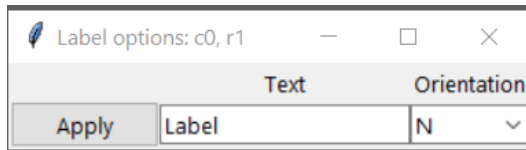
bg color/text color: white black

Apply

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



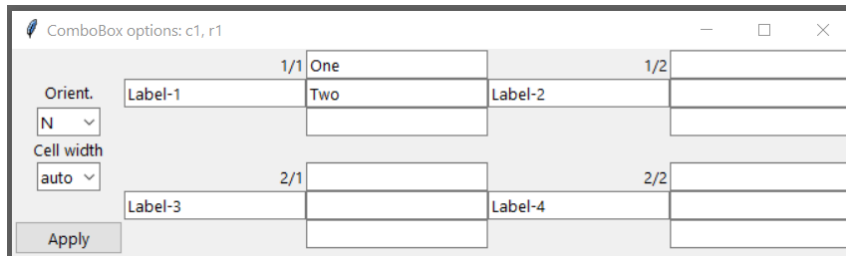
Label options: c0, r1

Text Orientation

Apply Label N

Available settings: default text and orientation

## Combobox



ComboBox options: c1, r1

Orient. N

Cell width auto

1/1 One 1/2

Label-1 Two Label-2

2/1 2/2

Label-3 Label-4

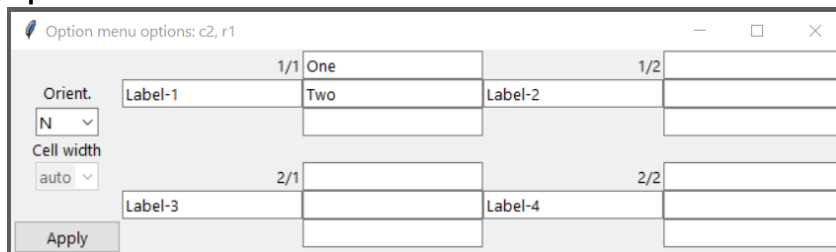
Apply

You can place up to 4 comboboxs with 3 options

Available settings:

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

## Option menu



Option menu options: c2, r1

Orient. N

Cell width auto

1/1 One 1/2

Label-1 Two Label-2

2/1 2/2

Label-3 Label-4

Apply

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)

## 4. Set the opened options windows and Apply

```
#-----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
- colspan 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 and
  - common merged cells disabled
- Canvas 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
- 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.

### Supported widgets:

menu (3x3), button (4x3), canvas, text with scroll, entry, label, combobox (4x3), optionmenu (4x3), radio (4x3), message, Universal Frame

*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: [epromirok@gmail.com](mailto:epromirok@gmail.com)*

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

Paypal



Thank you!