Python 3 - Tkinter Toplevel

Toplevel widgets work as windows that are directly managed by the window manager. They do not necessarily have a parent widget on top of them.

Your application can use any number of top-level windows.

Syntax

Here is the simple syntax to create this widget −

```
w = Toplevel ( option, ... )
```

Parameters

options − Here is the list of most commonly used options for this widget. These options can be used as key-value pairs separated by commas.

Sr.No.	Option & Description
1	**bg** The background color of the window.
2	**bd** Border width in pixels; default is 0.
3	**cursor** The cursor that appears when the mouse is in this window.
4	**class_** Normally, text selected within a text widget is exported to be the selection in the window manager. Set exportselection = 0 if you don't want that behavior.
5	**font** The default font for text inserted into the widget.
6	**fg** The color used for text (and bitmaps) within the widget. You can change the color for tagged regions; this option is just the default.
7	**height** Window height.
8	**relief** Normally, a top-level window will have no 3-d borders around it. To get a shaded border, set the bd option larger that its default value of zero, and set the relief option to one of the constants.
9	**width** The desired width of the window.

Methods

Toplevel objects have these methods −

Sr.No.	Methods and Description
1	**deiconify()** Displays the window, after using either the iconify or the withdraw methods.
2	**frame()** Returns a system-specific window identifier.
3	**group(window)** Adds the window to the window group administered by the given window.
4	**iconify()** Turns the window into an icon, without destroying it.
5	**protocol(name, function)** Registers a function as a callback which will be called for the given protocol.
6	**iconify()** Turns the window into an icon, without destroying it.
7	**state()** Returns the current state of the window. Possible values are normal, iconic, withdrawn and icon.
8	**transient([master])** Turns the window into a temporary(transient) window for the given master or to the window's parent, when no argument is given.
9	**withdraw()** Removes the window from the screen, without destroying it.
10	**maxsize(width, height)** Defines the maximum size for this window.
11	**minsize(width, height)** Defines the minimum size for this window.
12	**positionfrom(who)** Defines the position controller.
13	**resizable(width, height)** Defines the resize flags, which control whether the window can be resized.
14	**sizefrom(who)** Defines the size controller.
15	**title(string)** Defines the window title.

Example

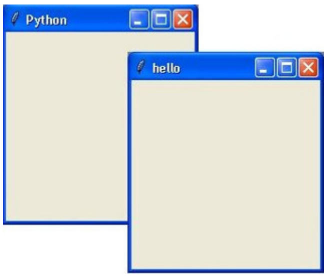
Try following example yourself −

```
# !/usr/bin/python3
from tkinter import *

root = Tk()
root.title("hello")
top = Toplevel()
top.title("Python")
top.mainloop()
```

Result

When the above code is executed, it produces the following result −


Python
hello