CodeWithHarry

HTML CSS JS C C++ JAVA PYTHON PHP

Overview Q&A Downloads Announcements

Menus & Submenus In Tkinter Python | Python Tkinter GUI Tutorial In Hindi #17

The **Menu** widget is used to implement different types of menus (toplevel, pulldown, and popup menus). The goal of this widget is to allow us to create all kinds of menus that can be used by our applications. It is also possible to use other extended widgets to implement new types of menus, such as the *OptionMenu* widget, which implements a special type that generates a pop-up list of items within a selection.

Attributes:

- **bg:** The background color for choices not under the mouse.
- **bd:** The width of the border around all the choices. **Default is 1.**
- **fg:** The foreground color used for choices not under the mouse.
- **tearoff:** Normally, a menu can be torn off. The first position (position 0) in the list of choices is occupied by the tear-off element and the additional choices are added starting at position 1. If you set tearoff=0, the menu will not have a tear-off feature and choices will be added starting at position 0.
- **relief:** The default 3-D effect for menus is relief=**RAISED**.
- **title:** Normally, the title of a tear-off menu window will be the same as the text of the menubutton or cascade that lead to this menu. If you want to change the title of that window, set the title option to that string.

Code is described below:

```
from tkinter import *
root = Tk()
root.geometry("733x566")
root.title("Pycharm")

def myfunc():
    print("Mai ek bahut hi natkhat aur shaitaan function hoon")

# #Use these to create a non dropdown menu
# mymenu = Menu(root)
# mymenu.add_command(label="File", command=myfunc)
# mymenu.add_command(label="Exit", command=quit)
# root.config(menu=mymenu)


mainmenu = Menu(root)

m1 = Menu(mainmenu, tearoff=0)
m1.add_command(label="New project", command=myfunc)
m1.add_command(label="Save", command=myfunc)
m1.add_separator()
m1.add_command(label="Save As", command=myfunc)
m1.add_command(label="Print", command=myfunc)
root.config(menu=mainmenu)
mainmenu.add_cascade(label="File", menu=m1)

m2 = Menu(mainmenu, tearoff=0)
m2.add_command(label="Cut", command=myfunc)
m2.add_command(label="Copy", command=myfunc)
m2.add_separator()
m2.add_command(label="Paste", command=myfunc)
m2.add_command(label="Find", command=myfunc)
```

```
root.config(menu=mainmenu)
mainmenu.add_cascade(label="Edit", menu=m2)

root.mainloop()
```

- Importing *tkinter* is the same as importing any other module in the Python code. Note that, the name of the module in Python 2.x is '*Tkinter*' and in Python 3.x it is '*tkinter*'.

```
from tkinter import *
```

- To create the main window, Tkinter offers a method 'Tk'. To change the name of the window, you can change the className to the desired one.

```
root = Tk()
```

- To set the dimensions of the Tkinter window and to set the position of the main window on the user's desktop, geometry() function is used. As in example: the width is 733 pixels and height is 566 pixels so we can write the function as *geometry(733x566).*

```
root.geometry("733x566")
```

- To set a title of the GUI window title() method is used. Here it is taken as "Pycharm".

```
root.title("Pycharm")
```

- To define a function 'def' (i.e. here the function myfunc() is defined) is used. Whenever the function will be called the printing statement within this function will be executed.

```
def myfunc():
    print("Mai ek bahut hi natkhat aur shaitaan function hoon")
```

- **For creating a non-dropdown menu widget:**

1. Menu() widget is created in the root or parent window and this widget is taken as a variable named "mymenu".

2. For adding a menu item to the Menu widget we use add_command() method. We pass the attribute "label" which shows the name of the given menu item (i.e. "File", "Exit") and command which call the function and executes the statement within the function (i.e. myfunc, quit).
3. **config()** is used to access an object's attributes after its initialization and then the menu is set with the variable "mymenu" .

```
mymenu = Menu(root)
mymenu.add_command(label="File", command=myfunc)
mymenu.add_command(label="Exit", command=quit)
root.config(menu=mymenu)
```

- **For creating a dropdown menu widget:**

1. Menu() widget is created in the root or parent window and this widget is taken as a variable named "mainmenu".
2. Three dropdown menus are created (m1, m2 & m3). With each menu, four menu items are added and their names are set using the Label attribute (i.e. New Project, Save, Print, etc.) *Tearoff=0* is used so that the menu will not have a tear-off feature, and choices will be added starting at position 0 (the menu section will be attached with GUI window always).
3. **config()** is used to access an object's attributes after its initialization and then the menu is set with the variable "mainmenu" .
4. For adding the new hierarchical menu by associating a given menu m1 to a parent menu mainmenu add_cascade() is used and the label is passed to set the name of the label (i.e. "File").
5. For adding a separator line to the menu we use add_separator() method.

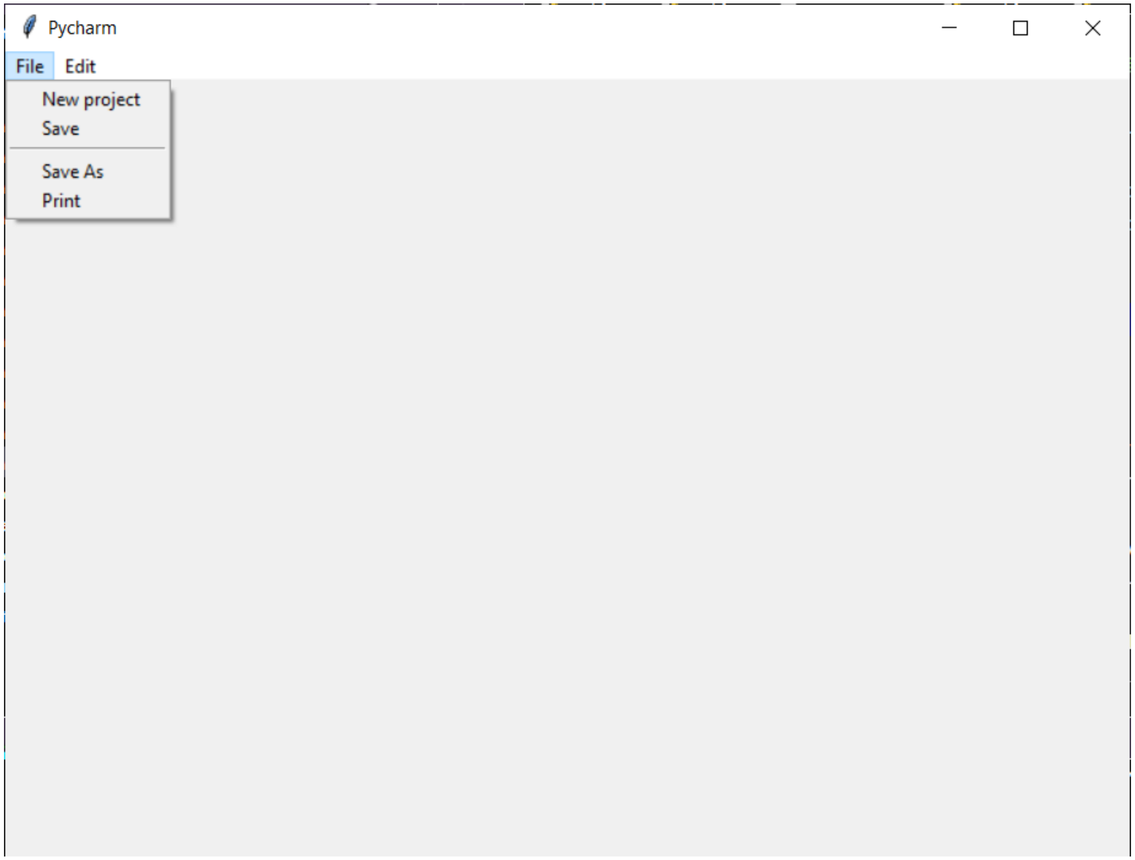
```
mainmenu = Menu(root)

m1 = Menu(mainmenu, tearoff=0)
m1.add_command(label="New project", command=myfunc)
m1.add_command(label="Save", command=myfunc)
m1.add_separator()
m1.add_command(label="Save As", command=myfunc)
m1.add_command(label="Print", command=myfunc)
root.config(menu=mainmenu)
mainmenu.add_cascade(label="File", menu=m1)

m2 = Menu(mainmenu, tearoff=0)
m2.add_command(label="Cut", command=myfunc)
```

```
m2.add_command(label="Copy", command=myfunc)
m2.add_separator()
m2.add_command(label="Paste", command=myfunc)
m2.add_command(label="Find", command=myfunc)
root.config(menu=mainmenu)
mainmenu.add_cascade(label="Edit", menu=m2)
```

- There is a method known by the name *mainloop()* which is used when your application is ready to run. This is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed.

```
root.mainloop()
```

Output: The output of the code (or the GUI window) is given below:



Code as described/written in the video

Copy

```
from tkinter import *
root = Tk()
root.geometry("733x566")
root.title("Pycharm")

def myfunc():
    print("Mai ek bahut hi natkhat aur shaitaan function hoon")

# #Use these to create a non dropdown menu
# mymenu = Menu(root)
# mymenu.add_command(label="File", command=myfunc)
# mymenu.add_command(label="Exit", command=quit)
# root.config(menu=mymenu)


mainmenu = Menu(root)

m1 = Menu(mainmenu, tearoff=0)
m1.add_command(label="New project", command=myfunc)
m1.add_command(label="Save", command=myfunc)
m1.add_separator()
m1.add_command(label="Save As", command=myfunc)
m1.add_command(label="Print", command=myfunc)
root.config(menu=mainmenu)
mainmenu.add_cascade(label="File", menu=m1)

m2 = Menu(mainmenu, tearoff=0)
m2.add_command(label="Cut", command=myfunc)
```

```
m2.add_command(label="Copy", command=myfunc)
m2.add_separator()
m2.add_command(label="Paste", command=myfunc)
m2.add_command(label="Find", command=myfunc)
root.config(menu=mainmenu)
mainmenu.add_cascade(label="Edit", menu=m2)

root.mainloop()
```

Previous

Next

CodeWithHarry