

Chapter 10

Basic GUI Programming Using Tkinter



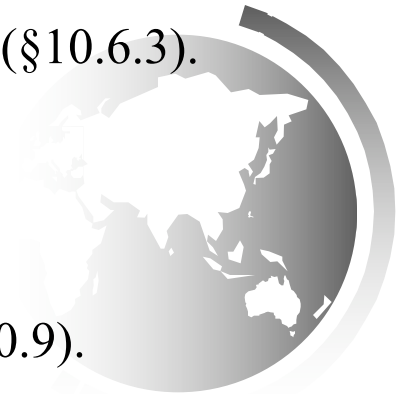
Motivations

Tkinter is not only a useful tool for developing GUI projects, but also a valuable pedagogical tool for learning object-oriented programming.



Objectives

- ◆ To create a simple GUI application with Tkinter (§10.2).
- ◆ To process events by using callback functions that are bound to a widget's command option (§10.3).
- ◆ To use labels, entries, buttons, check buttons, radio buttons, messages, and text to create graphical user interfaces (§10.4).
- ◆ To draw lines, rectangles, ovals, polygons, and arcs and display text strings in a canvas (§10.5).
- ◆ To use geometry managers to lay out widgets in a container (§10.6).
- ◆ To lay out widgets in a grid by using the grid manager (§10.6.1).
- ◆ To pack widgets side by side or on top of each other by using the pack manager (§10.6.2).
- ◆ To place widgets in absolute locations by using the place manager (§10.6.3).
- ◆ To write a GUI loan calculator (§10.7).
- ◆ To write a GUI program for checking a Sudoku solution (§10.8).
- ◆ To use images in widgets (§10.9).
- ◆ To write a GUI program that displays the images of four cards (§10.9).



Getting Started with Tkinter

Getting started with Tkinter with a simple example.



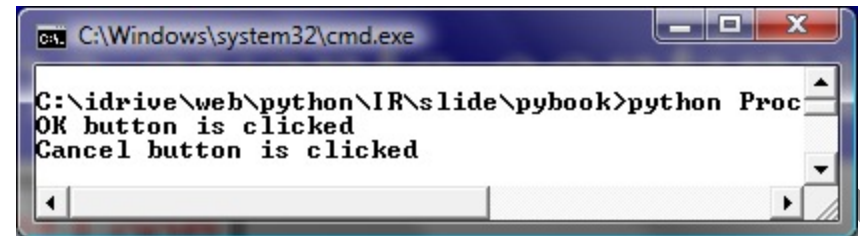
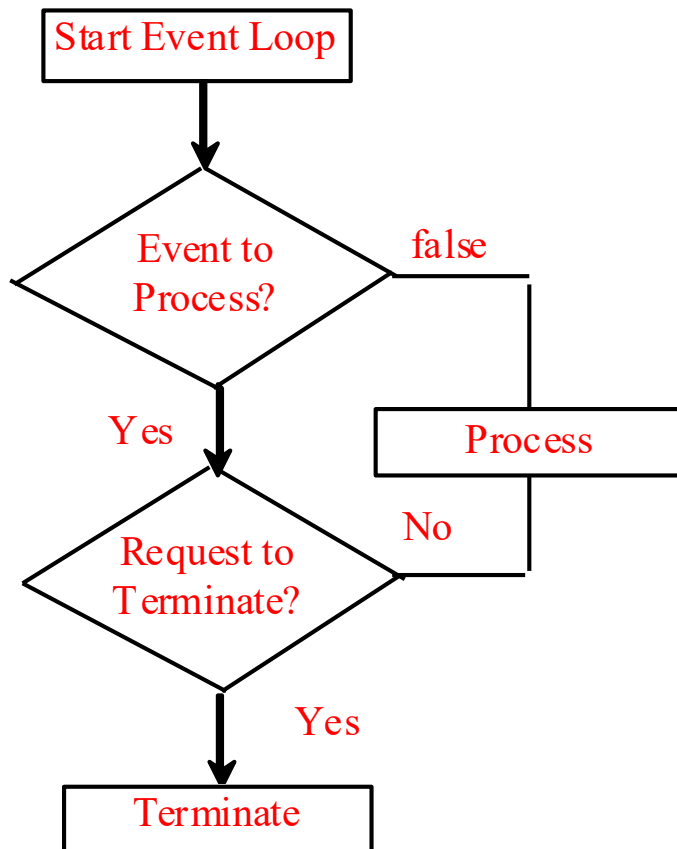
SimpleGUI



Processing Events

`window.mainloop()` # Create an event loop

The statement creates an event loop. The event loop processes the events continuously.



ProcessButtonEvent

The Widget Classes

Widget Class	Description
Button	A simple button, used to execute a command.
Checkbutton	Clicking a check button toggles between the values.
Radiobutton	Clicking a radio button sets the variable to that value, and clears all other radio buttons associated with the same variable.
Entry	A text entry field, a.k.a a text field or a text box.
Frame	A container widget for containing other widgets.
Menu	A menu pane, used to implement pull down and popup menus.
Menubutton	A menu button, used to implement pull down menus.
Label	Displays a text or an image.
Message	Displays a text. Similar to the label widget, but can automatically wrap text to a given width or aspect ratio.
Text	Formatted text display. Allows you to display and edit text with various styles and attributes. Also supports embedded images and windows.
Scale	Allows you to set a numerical value by dragging a "slider".
Canvas	Structured graphics, used to draw graphs and plots, create graphics editors, and to implement custom widgets.
Toplevel	A container widget displayed as a separate, top-level window.



Color and Font

To specify a color, you can either use a color name such as red, yellow, green, blue, white, black, purple, etc, or explicitly specify the red, green, and blue (RGB) color components using a string #RRGGBB, where RR, GG, BB are hexadecimal representations of the red, green and blue values, respectively.

"Times 10 bold"

"Helvetica 10 bold italic"

"Courier New 20 bold italic"

"Courier New 20 bold italic over strike underline"



Text Formatting

The text in a label and a button is centered by default. You can change it by using the justify option with values LEFT, CENTER, or RIGHT. You can also display the text in multiple lines by inserting the newline character `\n` to separate texts.



Mouse Cursor

You can set a mouse cursor by using the cursor option with values such as "arrow" (default), "circle", "cross" "plus", etc.



Change Properties

```
widgetName["propertyName"] = newPropertyValue
```

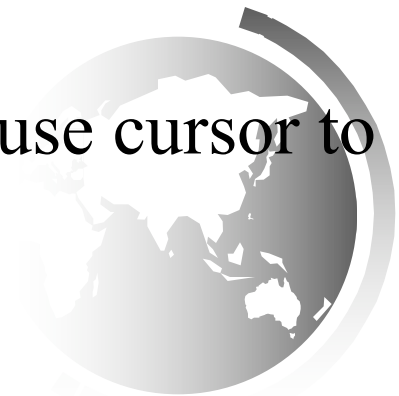
```
btShowOrHide = Button(window, text = "Show", bg =  
"white")
```

```
btShowOrHide["text"] = "Hide"
```

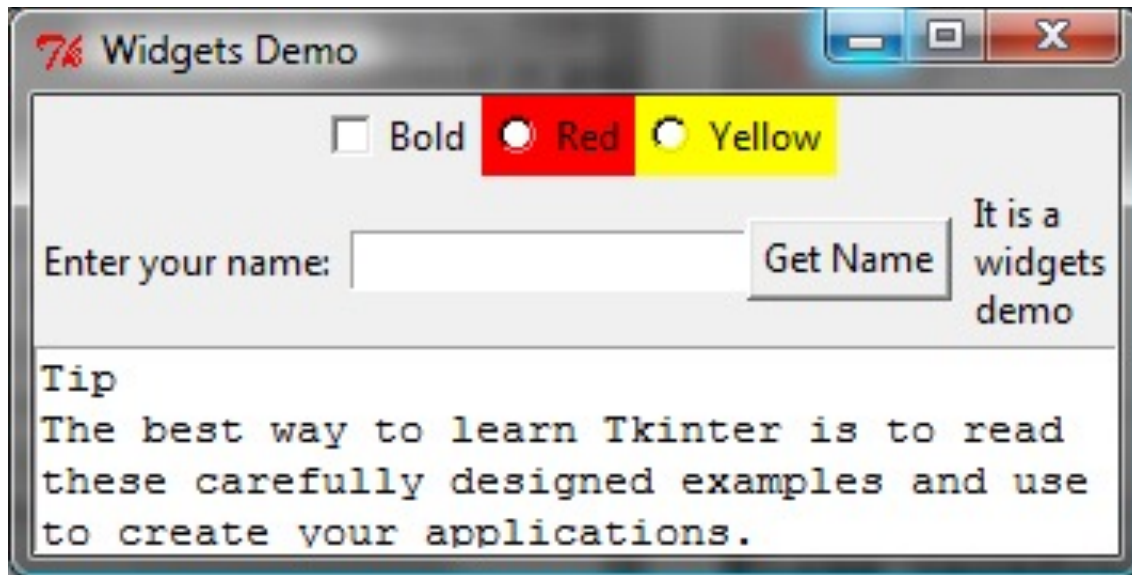
```
btShowOrHide["bg"] = "red"
```

```
btShowOrHide["fg"] = "#AB84F9" # Change fg color to  
#AB84F9
```

```
btShowOrHide["cursor"] = "plus" # Change mouse cursor to  
plus
```



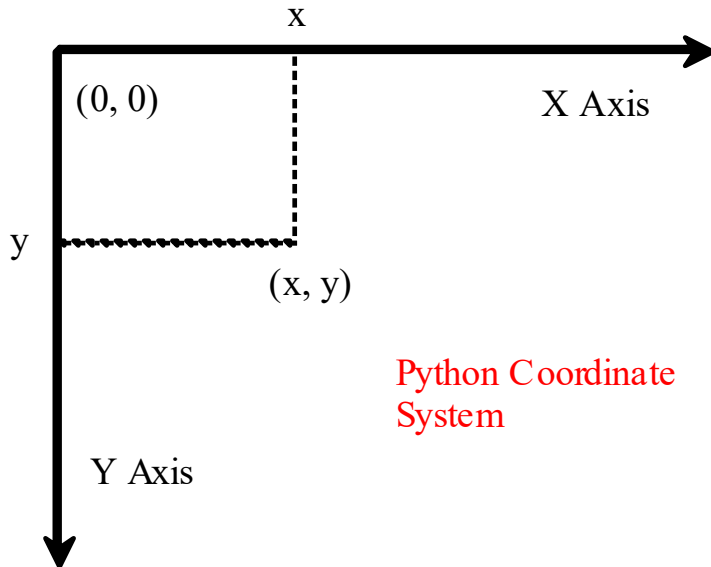
Widget Demo



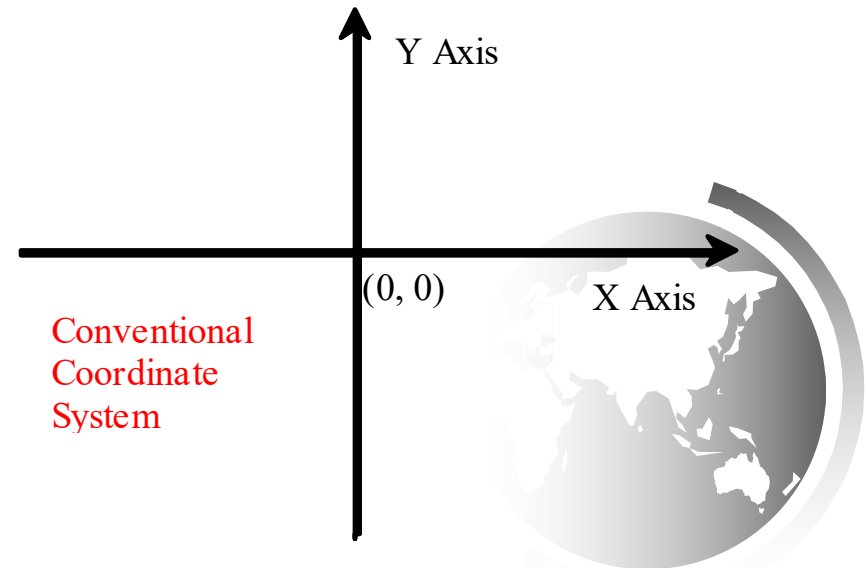
WidgetsDemo

Canvas

Canvas can be used to display shapes. You can use the method such as `create_rectangle`, `create_oval`, `create_arc`, `create_polygon`, and `create_line` to draw a rectangle, oval, arc, polygon, and line on a canvas.

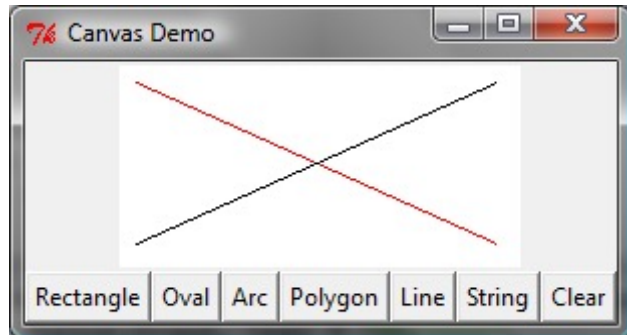
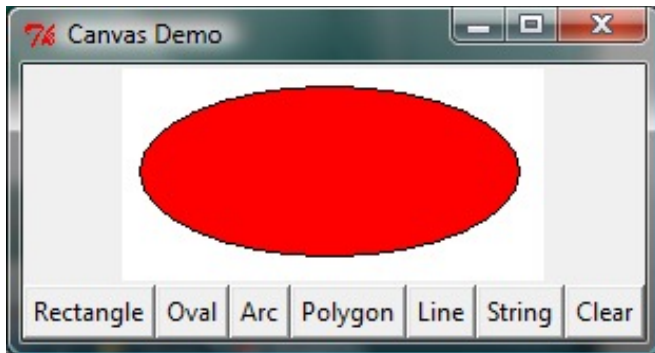
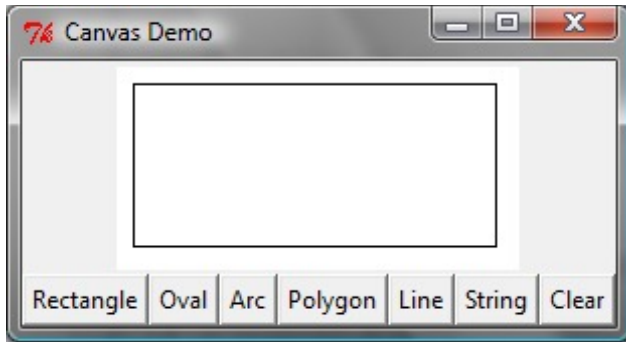


Python Coordinate System



Conventional Coordinate System

Canvas Demo



CanvasDemo



Drawing Methods

(x1, y1)



(x2, y2)

`canvas.create_rectangle(x1, y1,
x2, y2)`

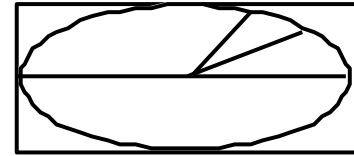
(x1, y1)



(x2, y2)

`canvas.create_oval(x1, y1, x2, y2)`

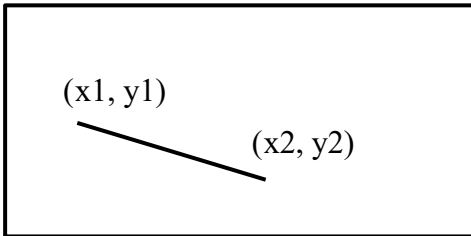
(x1, y1)



(x2, y2)

`canvas.create_arc(x1, y1, x2, y2, start, extent)`

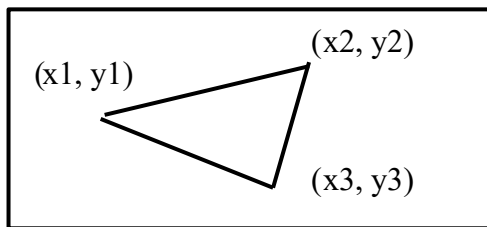
(x1, y1)



(x2, y2)

`canvas.create_line(x1, y1, x2, y2)`

(x1, y1)



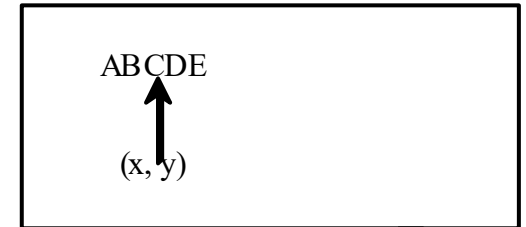
(x2, y2)

(x3, y3)

`canvas.create_polygon(x1, y1, x2, y2, x3, y3)`

ABCDE

(x, y)



`canvas.create_text(x, y, text = "ABCDE")`

Geometry Managers

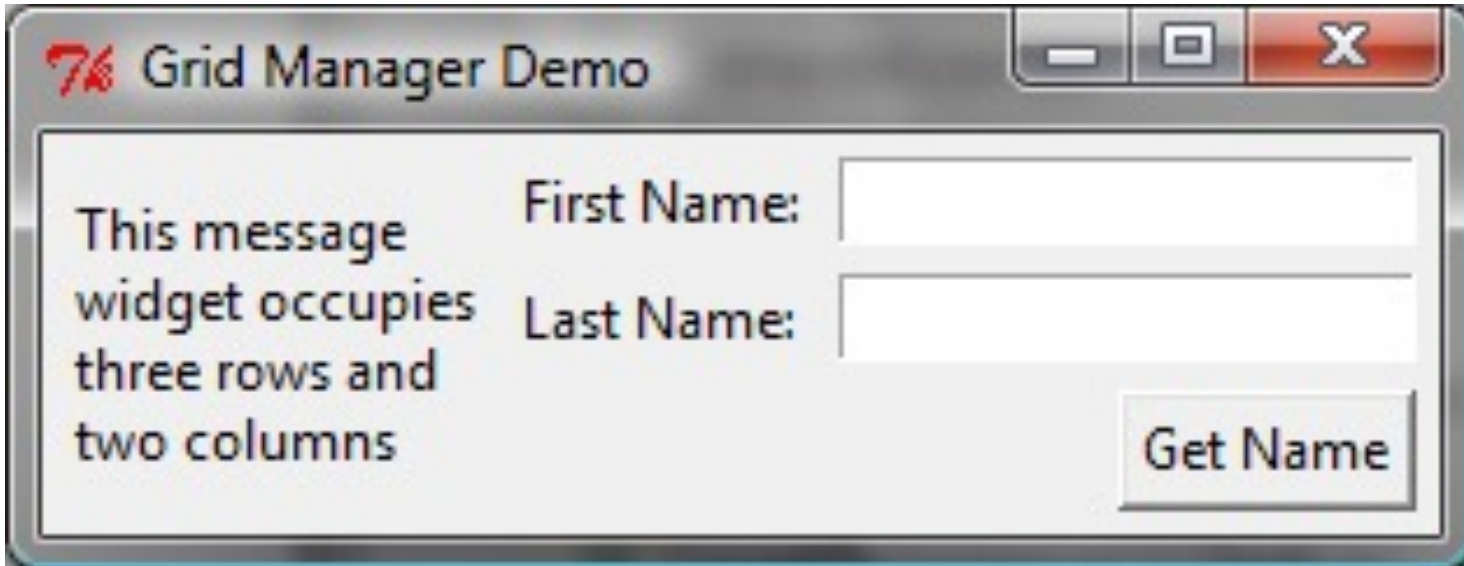
Grid Manager

Pack Manager

Place Manager

Since each manager has its own style of placing the widget, it is not a good practice to mix the managers for the widgets in the same container. You can use a frame as a subcontainer to achieve desired layout.

Grid Managers

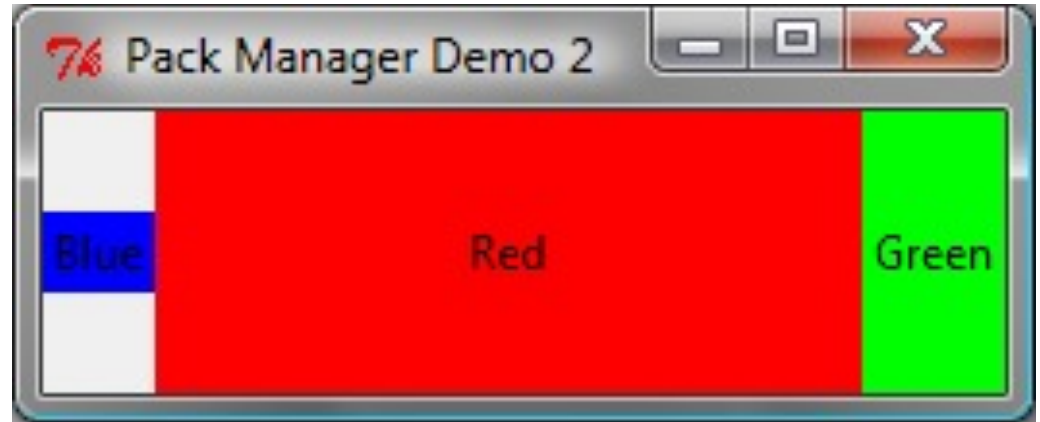
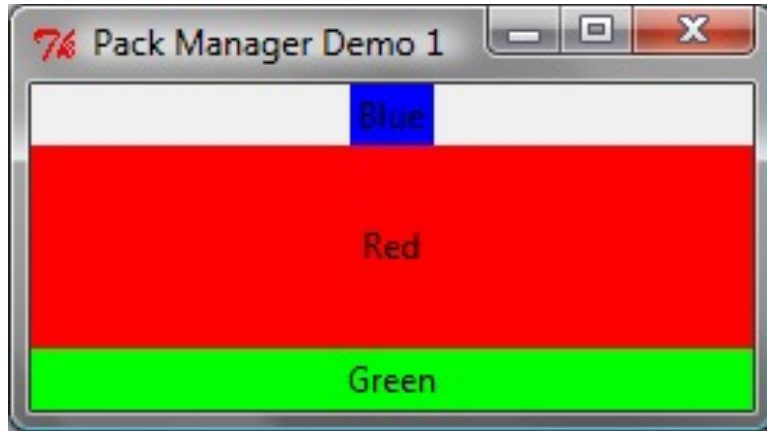


The screenshot shows a Java Swing window titled "Grid Manager Demo". The window contains a grid layout with the following components:

- A text label: "This message widget occupies three rows and two columns".
- A label: "First Name:" followed by a text input field.
- A label: "Last Name:" followed by a text input field.
- A button: "Get Name".

GridManagerDemo

Pack Managers

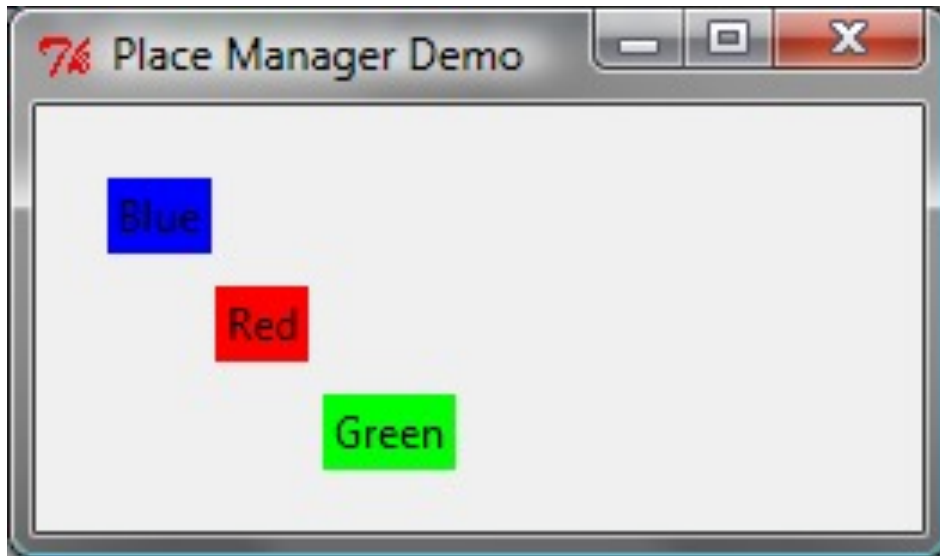


PackManagerDemo1

PackManagerDemo2



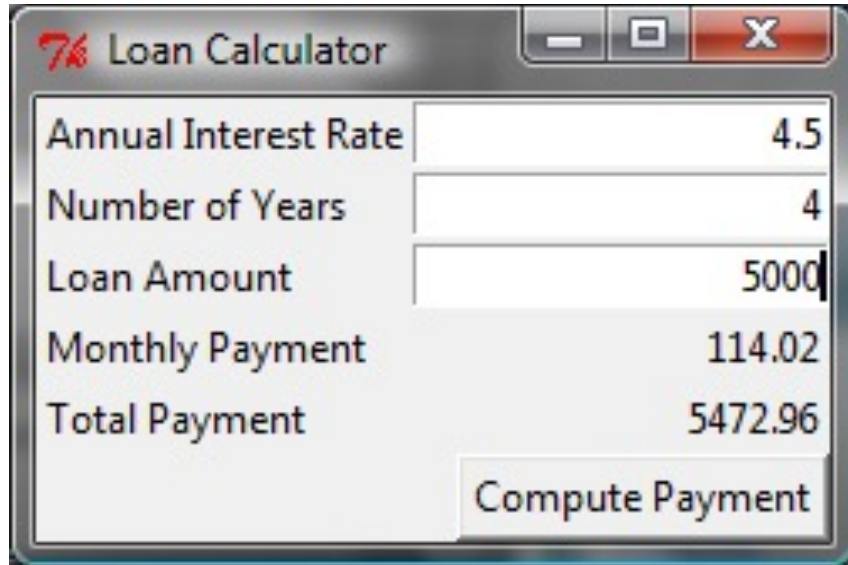
Place Managers



PlaceManagerDemo

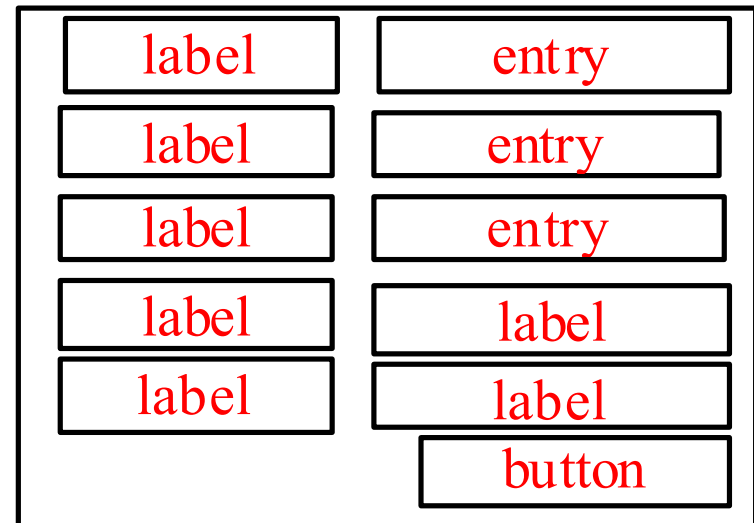


Case Study: Loan Calculator



Annual Interest Rate	4.5
Number of Years	4
Loan Amount	5000
Monthly Payment	114.02
Total Payment	5472.96

Compute Payment

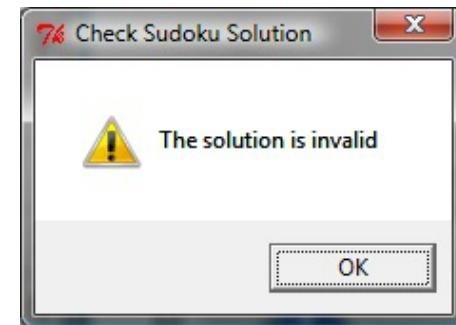
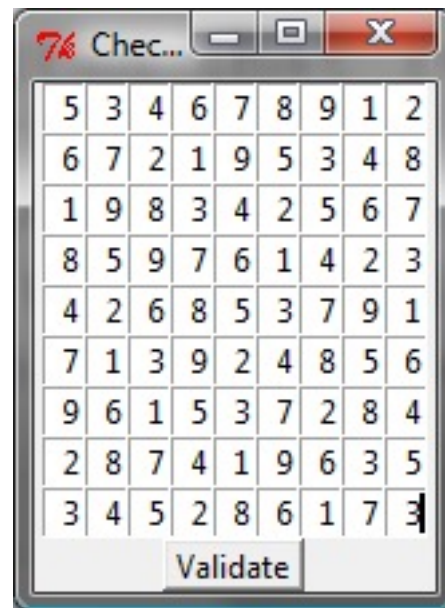
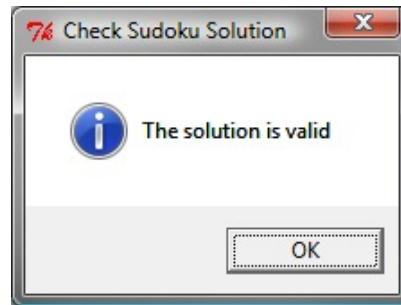
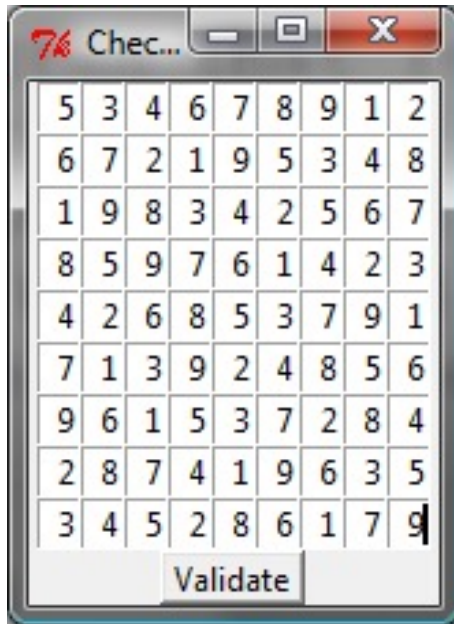


label	entry
label	entry
label	entry
label	label
label	label
	button

LoanCalculator



Sudoku GUI



SudokuGUI

Display Images

You can add an image in a label, button, check button, and radio button. To create an image, use the PhotoImage class as follows:

```
photo = PhotoImage(file = imagefilename)
```

The image file must be GIF. You can use a conversion utility to convert image files in other format into GIF.



Image Example



ImageDemo

Deck of Cards GUI



DeckOfCardsGUI