Visit Suncam.com for more courses

Continuing Education Course #364
Python Programming for Engineers - Part 3:
Graphical User Interfaces I

1. A graphical user interface (GUI) is

○ a. a computer program that enables a user to modify the performance and output of an electronic device via text commands typed in a command line.

○ b. an interface that enables a user to interact with a computer program or an electronic device by manipulating visual indications and graphical elements.

○ c. All of the above.

2. The advantages of GUIs include

○ a. simplicity and ease of use of a computer program by an untrained user.

○ b. a non-expert user needs to memorize fewer commands and special codes than otherwise.

○ c. All of the above.

3. Given the code,

```
import tkinter as tk
root = tk.Tk()
root.mainloop( )
```

Which of the following is correct?

○ a. The `tkinter` module is imported into the current session.

○ b. `Tk( )` opens a main window for developing the graphical user interface.

○ c. All of the above.

4. Given the code,

```
import tkinter as tk
root = tk.Tk()
root.mainloop( )
```

Which of the following is correct?

○ a. `tk` is a required variable name for the `tkinter` application.

○ b. `root` is the required name for the main application window.

○ c. `mainloop( )` keeps the main application window continuously visible.

5. The following are `tkinter` widgets.
○ a. Button, Entry, Checkbox
○ b. Label, Spinbox, Text
○ c. Textbox, Listbox, Menu

6. The following are Label widget options.
○ a. relief, textvariable, foreground
○ b. function, height, width
○ c. underline, side, justify

7. What are `tkinter` layout managers used for?
○ a. To control how the widgets are arranged on the main window.
○ b. To control what widgets on the main window are visible.
○ c. To control what widgets on the main window are visible and enabled.

8. Given the code,

```
import tkinter as tk
root = tk.Tk( )
label = tk.Label(root, text = 'Welcome to Python')
root.mainloop( )
```

Which of the following must be added for the Label to be arranged in such a manner that the available main window space is optimized?

○ a. `Label.pack( )`
○ b. `label.pack( )`
○ c. `label.pack`

9. Given the code,

```
import tkinter as tk
root = tk.Tk( )
label1 = tk.Label(root, text = 'Welcome to Python')
label2 = tk.Label(root, text = 'For Engineers')
root.mainloop( )
```

Which of the following must be added for the Labels to be arranged in a tabular format on the main window?

○ a.

```
label1.grid(row = 0, column = 1)
   label2.grid(row = 1, column = 2)
```

○ b.

```
label1.grid(0, 1)
  label2.grid(1, 2)
```

○ c.

```
label1.pack(row = 0, column = 1)
  label2.pack(row = 1, column = 2)
```

10. Given the code,

```
import tkinter as tk
root = tk.Tk( )
root.mainloop( )
```

Which of the following will add a multiline Text widget?

○ a.

```
text = tk.Text(root, text = 'Welcome to Python Programming\nFor Engineers')
  text.pack( )
```

○ b.

```
text = tk.Text(root)
  text.insert(tk.END, 'Welcome to Python Programming\nFor Engineers')
  text.pack( )
```

○ c. None of the above.

11. Given the code,

```
import tkinter as tk
root = tk.Tk( )
root = tk.Tk()
label = tk.Label(root, text = ' ')
entry = tk.Entry(root)
button = tk.Button(root, text= 'ENTER', command = func1)
entry.grid(row = 0, column = 0)
```

```
button.grid(row = 1, column = 0)
label.grid(row = 2, column = 0)
root.mainloop( )
```

Which of the following needs to be implemented to give functionality to the button?

- ○ a. The `command` attribute needs to be defined
- ○ b. The value assigned to the command option needs to be defined
- ○ c. All of the above

12. Given the function definition,

```
def func1( ):
    label.config(text = entry.get( ))
```

What does the `get( )` method do?

- ○ a. It transfers control of program execution from the function back to the main program.
- ○ b. It returns the current content of the Entry widget, as a string.
- ○ c. It write the value of the Text widget to the Entry widget.

13. Given the function definition,

```
def func1( ):
    label.config(text = entry.get( ))
```

What does the function `func1( )` do?

- ○ a. Modifies the `entry` option of the widget `label` using the value returned from the widget `text`.
- ○ b. Modifies the `text` option of the widget `entry` using the value returned from the widget `label`.
- ○ c. Modifies the `text` option of the widget `label` using the value returned from the widget `entry`.

14. Given the code,

```
def func1( ):
    label.config(text = entry.get( ))

import tkinter as tk
root = tk.Tk()
label = tk.Label(root, text = Python Programming ')
entry = tk.Entry(root)
button = tk.Button(root, text= 'ENTER', command = func1)
entry.grid(row = 0, column = 0)
button.grid(row = 1, column = 0)
label.grid(row = 2, column = 0)
root.mainloop()
```

The user types 'For Engineers and Scientists' in the Entry widget, and clicks on the ENTER Button. The output in the Label is:

○ a. `'Python Programming '`
○ b. `'Python Programming For Engineers and Scientists'`
○ c. `'For Engineers and Scientists'`

15. Which of the following code snippets will create a Checkbutton?

○ a. `check = tk.Checkbutton(root, text = 'I Agree to the Terms and Conditions')`
○ b. `Checkbutton = tk.check(root, text = 'I Agree to the Terms and Conditions')`
○ c. `check = tk.Checkbutton(text = 'I Agree to the Terms and Conditions')`

16. Which of the following code snippets will create a Radiobutton with three options?

○ a.

```
radio1 = tk.Radiobutton root, text = 'Asphalt')
   radio2 = tk.Radiobutton root, text = 'Concrete')
   radio3 = tk.Radiobutton root, text = 'Pavers')
```

○ b.

```
radio = tk.Radiobutton root, text = 'Asphalt')
   radio = tk.Radiobutton root, text = 'Concrete')
   radio = tk.Radiobutton root, text = 'Pavers')
```

○ c.

```
radio1 = tk.Radiobutton root1, text = 'Asphalt')
   radio2 = tk.Radiobutton root2, text = 'Concrete')
   radio3 = tk.Radiobutton root3, text = 'Pavers')
```

17. Given the code,

```
def func1( ):
    label.config(text = list1..curselection( ))

import tkinter as tk
root = tk.Tk( )
label = tk.Label(root, text = 'Python Engineering Inc. ')
list1 = tk.Listbox(root, height = 4)
list1.insert(1, ' ')
list1.insert(2, 'Asphalt')
list1.insert(3, 'Concrete')
list1.insert(4, 'Pavers')
```

```
button = tk.Button(root, text= 'ENTER', command = func1)
list1.grid(row = 0, column = 0)
button.grid(row = 1, column = 0)
label.grid(row = 2, column = 0)
root.mainloop()
```

The user selects `'Concrete'` in the Listbox and clicks the ENTER Button. The output in the Label is:

- a. `2`
- b. `'Concrete'`
- c. `(Concrete, 2)`

18. Given the code,

```
def func1( ):
    label.config(text = list1.get(list1.curselection( )))

import tkinter as tk
root = tk.Tk( )
label = tk.Label(root, text = 'Python Engineering Inc. ')
list1 = tk.Listbox(root, height = 4)
list1.insert(1, ' ')
list1.insert(2, 'Asphalt')
list1.insert(3, 'Concrete')
list1.insert(4, 'Pavers')

button = tk.Button(root, text= 'ENTER', command = func1)
list1.grid(row = 0, column = 0)
button.grid(row = 1, column = 0)
label.grid(row = 2, column = 0)
root.mainloop( )
```

The user selects `'Pavers'` in the Listbox and clicks the ENTER Button. The output in the Label is:

- a. `3`
- b. `'Pavers'`
- c. `(3, 'Pavers')`

19. Given the code,

```
def func1( ):
    if float(spin.get( )) == 1:
        x = 'Freeway'
    elif float(spin.get( )) == 2:
        x = 'Uninterrupted Flow Highway'
    elif float(spin.get( )) == 3:
        x = 'State Signalized Arterial'
    elif float(spin.get( )) == 4:
        x = 'Non-State Signalized Roadway'
```

```
    label.config(text = x)

import tkinter as tk
root = tk.Tk( )
root.geometry('200x200')
label = tk.Label(root, text = 'Python Engineering Inc. ')
label2 = tk.Label(root, text = '', height = 2)
label3 = tk.Label(root, text = ' ', height = 2)
spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.RIGHT)

button = tk.Button(root, text= 'ENTER', command = func1)
label2.grid(row = 0, column = 0)
spin.grid(row = 1, column = 0)
button.grid(row = 2, column = 0)
label3.grid(row = 3, column = 0)
label.grid(row = 4, column = 0)
root.mainloop( )
```

The user selects the value 2 in the Spinbox and clicks the ENTER Button. The output in the Label is:

- ○ a. `'Uninterrupted Flow Highway'`
- ○ b. `'State Signalized Arterial'`
- ○ c. None of the above

20. Given the code,

```
def func1( ):
    if float(spin.get( )) == 1:
        x = 'Freeway'
    elif float(spin.get( )) == 2:
        x = 'Uninterrupted Flow Highway'
    elif float(spin.get( )) == 3:
        x = 'State Signalized Arterial'
    elif float(spin.get( )) == 4:
        x = 'Non-State Signalized Roadway'

    label.config(text = x)

import tkinter as tk
root = tk.Tk( )
root.geometry('200x200')
label = tk.Label(root, text = 'Python Engineering Inc. ')
label2 = tk.Label(root, text = '', height = 2)
label3 = tk.Label(root, text = ' ', height = 2)
spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.RIGHT)

button = tk.Button(root, text= 'ENTER', command = func1)
label2.grid(row = 0, column = 0)
spin.grid(row = 1, column = 0)
button.grid(row = 2, column = 0)
label3.grid(row = 3, column = 0)
label.grid(row = 4, column = 0)
root.mainloop( )
```

Which of the following code updates will change the alignment of the values in the Spinbox?

- ○ a. `spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.MIDDLE)`
- ○ b. `spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.CENTER)`
- ○ c. `spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.ALIGN)`

21. Given the code,

```
def func1( ):
    if float(spin.get( )) == 1:
        x = 'Freeway'
    elif float(spin.get( )) == 2:
        x = 'Uninterrupted Flow Highway'
    elif float(spin.get( )) == 3:
        x = 'State Signalized Arterial'
    elif float(spin.get( )) == 4:
        x = 'Non-State Signalized Roadway'

    label.config(text = x)

import tkinter as tk
root = tk.Tk( )
root.geometry('200x200')
label = tk.Label(root, text = 'Python Engineering Inc. ')
label2 = tk.Label(root, text = '', height = 2)
label3 = tk.Label(root, text = ' ', height = 2)
spin = tk.Spinbox(root, from_ = 1, to = 5, justify = tk.RIGHT)

button = tk.Button(root, text= 'ENTER', command = func1)
label2.grid(row = 0, column = 0)
spin.grid(row = 1, column = 0)
button.grid(row = 2, column = 0)
label3.grid(row = 3, column = 0)
label.grid(row = 4, column = 0)
root.mainloop( )
```

Which of the following code updates will change output the Label font color?

- ○ a. `label.grid(row = 4, column = 0, fg = 'red')`
- ○ b. `label.config(text = x, bg = 'red')`
- ○ c. `label.config(text = x, fg = 'red')`

22. Given the code,

```
import tkinter as tk
root = tk.Tk( )
root.geometry('200x200')

menu1 = tk.Menu(root)
menu2 = tk.Menu(menu1)
root.config(menu = menu1)
```

```
menu1.add_cascade(label = 'File', menu = menu2)

root.mainloop( )
```

Which of the following code updates will add an item to the File menu?

○ a. `menu2.add_command(label = 'New', command = CreateNewFile)`
○ b. `menu2.add_command(label = 'New', command = CreateNewFile)`
in conjunction with a definition of the function CreateNewFile
○ c. `menu1 = menu2.add_command(label = 'New', command = CreateNewFile)`
in conjunction with a definition of the function CreateNewFile

23. Given the code,

```
def CreateNewFile( ):
    print('Create a new file')

import tkinter as tk
root = tk.Tk( )
root.geometry('200x200')

menu1 = tk.Menu(root)
menu2 = tk.Menu(menu1)
root.config(menu = menu1)
menu1.add_cascade(label = 'File', menu = menu2)

menu2.add_command(label = 'New', command = CreateNewFile)

root.mainloop( )
```

Which of the following code snippets will add a new menu?

○ a.

```
menu3 = tk.Menu(menu1)
  menu1.add_cascade(label = 'Edit', menu = menu3)
```

○ b.

```
menu3 = tk.Menu(menu1)
  menu2.add_cascade(label = 'Edit', menu = menu3)
```

○ c.

```
menu1.add_cascade(label = 'Edit', menu = menu3)
```

24. Which of the following will add a Frame or a LabelFrame to the main window?

- a. `x = tk.Frame(root, text = 'Login Information')`
- b. `x = tk.LabelFrame(root, text = 'Login Information')`
- c. All of the above

25. Which of the following is the preferred approach to developing a Python `tkinter` program?

- a. Writing and testing the code line-by-line and block-by-block.
- b. Writing the entire script and then testing and validating the results.
- c. Troubleshooting the script when system errors and exceptions are raised.

Purchase this course on Suncam.com