Python Pizza Pizza GUI

Contents

Time required: 90 minutes

We are going to work with radio buttons and check buttons. This will be our result. We will start our masterpiece by adding radio buttons.
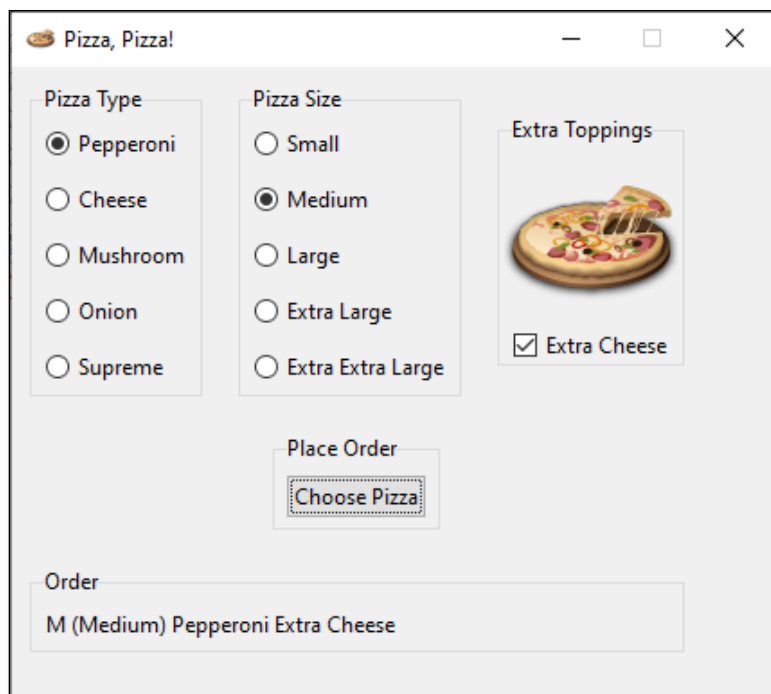
Tkinter Radio Buttons

Radio buttons allow you to select between one of several mutually exclusive choices.

Typically, you use radio buttons together in a set. They're a good option if you have a few choices that you want users to select.

To create radio buttons, you use the **Radiobutton** widget. The following shows how to create radio buttons using the **ttk.Radiobutton** constructor:

```
selected = tk.StringVar()
rad_1 = ttk.Radiobutton(container, text='Option 1', value='Value 1',
variable=selected)
rad_2 = ttk.Radiobutton(container, text='Option 2', value='Value 2',
variable=selected)
rad_3 = ttk.Radiobutton(container, text='Option 3', value='value 3',
variable=selected)
```

Each radio button has a different value. Tkinter supplies a **StringVar()** string and an **IntVar()** variable type that can be shared across multiple widgets. These radio buttons are the same group and share the same variable.

- **container:** the parent widget for the radio buttons.

- **text:** specifies the text that appears on the radio button.

- **value:** specifies the value that the radio button will hold.

- **variable:** must be a **tk.StringVar()**

Tutorial 1: Radio Buttons

The following code shows an implementation of Radiobuttons.

Find a pizza ico file that you like. There are lots of free ico files. There are links to suggested web sites in Programming Resources.

Here is one you can use.

https://www.iconfinder.com/icons/3558094/download/ico/4096

```python
"""
    Name: pizza1.py
    Author:
    Created:
    Purpose: Pizza ordering program with radio buttons and check boxes
"""
from tkinter import *
from tkinter.ttk import *


class Pizza:
    def __init__(self):
        self.root = Tk()
        self.root.title("Pizza, Pizza!")
        # Add icon to window corner
        self.root.iconbitmap("pizza.ico")
        # Prevent window from resizing
        self.root.resizable(False, False)
        # Set windows size and location on screen
        self.root.geometry("300x300+400+300")

        # Toppings is a tuple of tuples for the radio buttons
        # A tuple of tuples to build the radio buttons
        # first: display, second: value
        self.PIZZA_SIZES = (
            ("Small", "S (Small)"),
            ("Medium", "M (Medium)"),
            ("Large", "L (Large)"),
            ("Extra Large", "XL (Extra Large)"),
            ("Extra Extra Large", "XXL (Extra Extra Large)")
        )

        # Tkinter dynamic variable object
        # String: StringVar()
        # Int: IntVar()
        self.pizza_size = StringVar()
        # Sets the Medium radio button as chosen when the program loads
        self.pizza_size.set("M (Medium)")

        # Create the GUI widgets in a separate method
        self.create_widgets()
        # Call the mainloop method to start program
        mainloop()
```
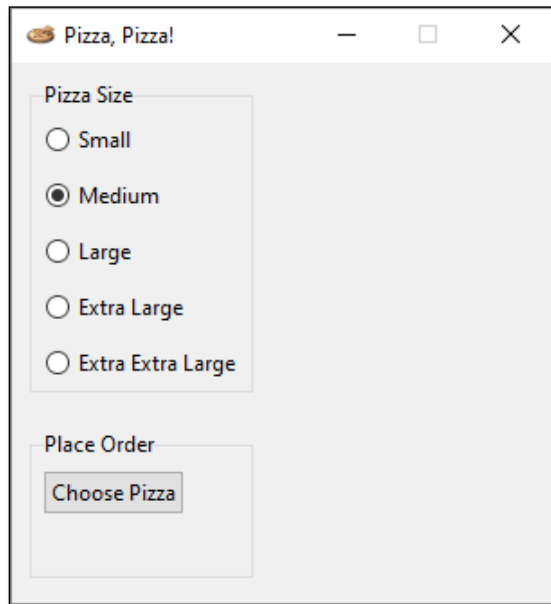
```
45  #---------------------- GET ORDER ----------------------------------#
46      def get_order(self):
47          # Get variable value from radio buttons
48          size = self.pizza_size.get()
49          # Display values
50          self.lbl_pizza_order.configure(text=f"{size}")
51
52  #---------------------- CREATE WIDGETS ----------------------------------#
53      def create_widgets(self):
54          self.create_frames()
55
56          # Build all the radio buttons in a for loop from a tuple
57          # Pack them into the frame
58          # text: Radio button text display
59          # value: value returned by choosing the radio button
60          for text, topping in self.PIZZA_SIZES:
61              Radiobutton(
62                  self.pizza_size_frame,      # Container for widget
63                  text=text,                  # Text displayed by radio button
64                  variable=self.pizza_size,   # Variable the value is stored in
65                  value=topping               # Value returned
66              ).pack(anchor=W, padx=5, pady=5)
67
68          self.btn_get_pizza = Button(
69              self.order_frame,
70              text="Choose Pizza",
71              command=self.get_order
72          )
73
74          self.lbl_pizza_order = Label(self.order_frame)
75
76          self.btn_get_pizza.grid(row=0, column=0, sticky=W)
77          self.lbl_pizza_order.grid(row=1, column=0)
78
79          # Set padding for all widgets inside the frames
80          for widget in self.order_frame.winfo_children():
81              widget.grid_configure(padx=5, pady=5)
82
83  #---------------------- CREATE FRAMES ----------------------------------#
84      def create_frames(self):
85          self.pizza_size_frame = LabelFrame(
86              self.root,
87              text="Pizza Size",
88              relief=GROOVE
89          )
90
91          self.order_frame = LabelFrame(
92              self.root,
93              text="Place Order",
94              relief=GROOVE
95          )
96
97          # Grid frames
98          self.pizza_size_frame.grid(row=0, column=0, padx=10, pady=10)
99          self.order_frame.grid(row=1, column=0, columnspan=2,
100                                padx=10, pady=10, sticky=EW)
101
102 # Create object to start program
103 pizza = Pizza()
```

Example run:

Tutorial 2: Check Buttons

Check Buttons stand alone. They are either on, or off.

```python
"""
    Name: pizza2.py
    Author:
    Created:
    Purpose: Pizza ordering program with radio buttons and check boxes
"""
from tkinter import *
from tkinter.ttk import *


class Pizza:
    def __init__(self):
        self.root = Tk()
        self.root.title('Pizza, Pizza!')
        # Add icon to window corner
        self.root.iconbitmap('pizza.ico')
        # Prevent window from resizing
        self.root.resizable(False, False)
        # Set windows size and location on screen
        self.root.geometry("300x300+400+300")

        # Toppings is a tuple of tuples for the radio buttons
        # Toppings is a tuple of tuples for the radio buttons
        # A tuple of tuples to build the radio buttons
        # first: display, second: value
        self.PIZZA_SIZES = (
            ("Small", "S (Small)"),
            ("Medium", "M (Medium)"),
            ("Large", "L (Large)"),
            ("Extra Large", "XL (Extra Large)"),
            ("Extra Extra Large", "XXL (Extra Extra Large)")
        )

        # Tkinter dynamic variable object
        # String: StringVar()
        # Int: IntVar()
        self.pizza_size = StringVar()
        # Sets the Medium radio button as chosen when the program loads
        self.pizza_size.set("M (Medium)")
        # StringVar for extra cheese check button
        self.extra_cheese = StringVar()

        # Create the GUI widgets in a separate method
        self.create_widgets()
        # Call the mainloop method to start program
        mainloop()
```

```python
48  #---------------------- GET ORDER -----------------------------------#
49      def get_order(self):
50          # Get variable values from radio button and checkbox
51          pizza_size = self.pizza_size.get()
52          extra_cheese = self.extra_cheese.get()
53          # Display values
54          self.lbl_pizza_order.configure(text=f"{pizza_size} {extra_cheese}")
55
56  #---------------------- CREATE WIDGETS ------------------------------#
57      def create_widgets(self):
58          self.create_frames()
59
60          # Build all the radio buttons in a for loop from a tuple
61          # Pack them into the frame in a group, they shared the same variable
62          # text: Radio button text display
63          # value: value returned by choosing the radio button
64          for text, topping in self.PIZZA_SIZES:
65              Radiobutton(
66                  self.pizza_size_frame,
67                  text=text,                      # Text displayed by radio button
68                  variable=self.pizza_size,       # Variable the value is stored in
69                  value=topping                   # Value returned
70              ).pack(anchor=W, padx=5, pady=5)
71
72          # Build checkbutton one at a time
73          self.chk_extra_cheese = Checkbutton(
74              self.extra_toppings_frame,
75              text="Extra Cheese",            # Text display by check button
76              onvalue="Extra Cheese",         # Value returned if checkbox is checked
77              offvalue="",                    # Value returned if checkbox is not checked
78              variable=self.extra_cheese      # Variable the value is stored in
79          )
80
81          self.btn_get_pizza = Button(
82              self.order_frame,
83              text="Choose Pizza",
84              command=self.get_order
85          )
86
87          self.lbl_pizza_order = Label(self.order_frame)
88
89          self.chk_extra_cheese.grid(row=0, column=0)
90          self.btn_get_pizza.grid(row=0, column=0, sticky=W)
91          self.lbl_pizza_order.grid(row=1, column=0)
92
93          # Set padding for all widgets inside the frames
94          for widget in self.order_frame.winfo_children():
95              widget.grid_configure(padx=5, pady=5)
96          for widget in self.extra_toppings_frame.winfo_children():
97              widget.grid_configure(padx=5, pady=5)
```

```
 99 #--------------------- CREATE FRAMES ---------------------------------#
100     def create_frames(self):
101         self.pizza_size_frame = LabelFrame(
102             self.root,
103             text="Pizza Size",
104             relief=GROOVE
105         )
106
107         self.extra_toppings_frame = LabelFrame(
108             self.root,
109             text="Extra Toppings",
110             relief=GROOVE
111         )
112
113         self.order_frame = LabelFrame(
114             self.root,
115             text="Place Order",
116             relief=GROOVE
117         )
118
119         # Grid frames
120         self.pizza_size_frame.grid(row=0, column=0, padx=10, pady=10)
121         self.extra_toppings_frame.grid(row=0, column=1, padx=10, pady=10)
122         self.order_frame.grid(row=1, column=0, columnspan=2,
123                             padx=10, pady=10, sticky=EW)
124
125
126 pizza = Pizza()
```

Example run:
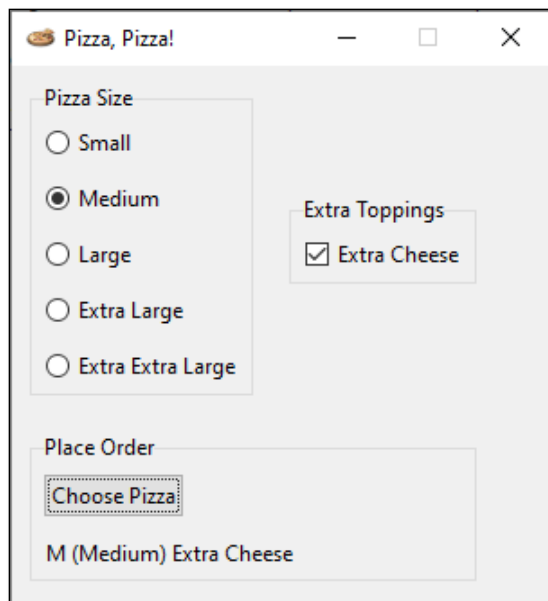
Tutorial 3: All Together Now

We are going to add one more frame of Radio Buttons to finish our Pizza Pizza Application.

```python
1  """
2      Name: pizza3.py
3      Author:
4      Created:
5      Purpose: Pizza program with radio buttons and check boxes
6  """
7  from tkinter import *
8  from tkinter.ttk import *
9
10
11 class Pizza:
12     def __init__(self):
13         self.root = Tk()
14         self.root.title('Pizza, Pizza!')
15         # Add icon to window corner
16         self.root.iconbitmap('pizza.ico')
17         # Prevent window from resizing
18         self.root.resizable(False, False)
19         self.root.geometry("425x350+400+300")
20
21         # Toppings is a tuple of tuples for the radio buttons
22         self.TOPPINGS = (
23             ("Pepperoni", "Pepperoni"),
24             ("Cheese", "Cheese"),
25             ("Mushroom", "Mushroom"),
26             ("Onion", "Onion"),
27             ("Supreme", "Supreme"),
28         )
29         # Toppings is a tuple of tuples for the radio buttons
30         # A tuple of tuples to build the radio buttons
31         # first: display, second: value
32         self.PIZZA_SIZES = (
33             ("Small", "S (Small)"),
34             ("Medium", "M (Medium)"),
35             ("Large", "L (Large)"),
36             ("Extra Large", "XL (Extra Large)"),
37             ("Extra Extra Large", "XXL (Extra Extra Large)")
38         )
39
40         self.pizza = StringVar()
41         self.pizza.set("Pepperoni")
42         self.selected_size = StringVar()
43         self.selected_size.set("M (Medium)")
44         self.extra_cheese = StringVar()
45
46         # Create the GUI widgets in a separate method
47         self.create_frames()
48         self.create_widgets()
49         # Call the mainloop method to start program
50         mainloop()
```

```python
52  #----------------------- GET ORDER ----------------------------------#
53      def get_order(self):
54          topping = self.pizza.get()
55          size = self.selected_size.get()
56          extra_cheese = self.extra_cheese.get()
57          self.lbl_pizza_order.configure(text=f"{size} {topping} {extra_cheese}")
58
59  #----------------------- CREATE WIDGETS -----------------------------------#
60      def create_widgets(self):
61
62          for text, topping in self.TOPPINGS:
63              Radiobutton(
64                  self.pizza_type_frame,
65                  text=text,
66                  variable=self.pizza,
67                  value=topping
68              ).pack(anchor=W, padx=5, pady=5)
69
70          for text, size in self.PIZZA_SIZES:
71              Radiobutton(
72                  self.pizza_size_frame,
73                  text=text,
74                  variable=self.selected_size,
75                  value=size
76              ).pack(anchor=W, padx=5, pady=5)
77
78          self.chk_extra_cheese = Checkbutton(
79              self.extra_toppings_frame,
80              text="Extra Cheese",
81              onvalue="Extra Cheese",
82              offvalue="",
83              variable=self.extra_cheese
84          )
85
86          btn_get_pizza = Button(
87              self.order_frame,
88              text="Choose Pizza",
89              command=self.get_order
90          )
91
92          self.lbl_pizza_order = Label(self.display_frame)
93
94          self.lbl_pizza_order.grid(row=0, column=0)
95          btn_get_pizza.grid(row=0, column=0)
96          self.chk_extra_cheese.grid(row=0, column=0)
97
98          # Set padding for all widgets inside the frame
99          for widget in self.order_frame.winfo_children():
100             widget.grid_configure(padx=5, pady=5)
101         for widget in self.display_frame.winfo_children():
102             widget.grid_configure(padx=5, pady=5)
```

```
104 #--------------------- CREATE FRAMES ---------------------------------#
105     def create_frames(self):
106         self.pizza_type_frame = LabelFrame(
107             self.root,
108             text="Pizza Type",
109             relief=GROOVE
110         )
111
112         self.pizza_size_frame = LabelFrame(
113             self.root,
114             text="Pizza Size",
115             relief=GROOVE
116         )
117
118         self.extra_toppings_frame = LabelFrame(
119             self.root,
120             text="Extra Toppings",
121             relief=GROOVE
122         )
123
124         self.order_frame = LabelFrame(
125             self.root,
126             text="Place Order",
127             relief=GROOVE
128         )
129         self.display_frame = LabelFrame(
130             self.root,
131             text="Order",
132             relief=GROOVE
133         )
134         # Grid frames
135         self.pizza_type_frame.grid(row=0, column=0, padx=10, pady=10)
136         self.pizza_size_frame.grid(row=0, column=1, padx=10, pady=10)
137         self.extra_toppings_frame.grid(row=0, column=2, padx=10, pady=10)
138         self.order_frame.grid(row=1, column=0, columnspan=3, padx=10, pady=10)
139         self.display_frame.grid(
140             row=2, column=0, columnspan=3, padx=10, pady=10, stick="ew")
141
142
143 pizza = Pizza()
```

Example run:

Assignment

- Add one more: Radio button choice

- Add two more: Check buttons

- Display the results of the additional control results in the order display

Assignment Submission

1. Attach the pseudocode.

2. Attach the program files.

3. Attach screenshots showing the successful operation of the program.

4. Submit in Blackboard.