

File - yarusoto.pyw

```
1 import tkinter as tk
2 from dataclasses import dataclass
3 from tkinter import ttk
4 from typing import List, Optional
5 from os import path
6
7
8 @dataclass
9 class TodoItem:
10     name: str
11     checked: Optional[tk.BooleanVar] = None
12
13
14 def add_todo(add_todo_text: ttk.Entry, todos_frame: ttk.Frame) -> None:
15     """
16     Add a new TodoItem to todo_list from the add_todo_text widget and re-renders the
17     list.
18     :param add_todo_text: ttk.Entry widget that hold the text that should be the new
19     TodoItem
20     :param todos_frame: The ttk.Frame that needs to be passed to update_todos to re-
21     render the list
22     """
23     global todo_list
24
25     todo_to_add: str = add_todo_text.get().strip()
26     if len(todo_to_add) > 0:
27         todo_list.append(TodoItem(todo_to_add))
28         add_todo_text.delete(0, "end")
29         update_todos(todos_frame, show_checked_var.get())
30
31
32 def update_todos(todos_frame: ttk.Frame, show_checked: bool) -> None:
33     """
34     Updates the todos_frame with the TodoItems in todo_list. This allows for hidden
35     checkboxes to be removed
36     immediately after they are checked .
37     :param todos_frame: The ttk.Frame that should display the TodoItems
38     :param show_checked: Whether just the checked or all the todos should be rendered
39     """
40     for old_todo in todos_frame.winfo_children():
41         old_todo.destroy()
42     for todo in todo_list:
43         # check if [TodoItem].checked is None, so it can be initialized
44         if todo.checked is None:
45             # need to instantiate tk var here because if it is done outside the class
46             or
47             # before tk is initialized the script crashes
48             todo.checked = tk.BooleanVar()
49         if not show_checked and todo.checked.get():
50             continue
```

File - yarusoto.pyw

```
48         else:
49             ttk.Checkbutton(
50                 todos_frame,
51                 text=todo.name,
52                 variable=todo.checked,
53                 onvalue=True,
54                 offvalue=False,
55                 command=lambda: update_todos(todos_frame, show_checked_var.get())
56             ).pack(anchor="w")
57
58
59 # global vars
60 todo_list: List[TodoItem] = []
61 show_checked_var: Optional[tk.BooleanVar] = None
62
63
64 def main() -> None:
65     """
66     Manages tk initialization and adds the non-dynamic widgets.
67     """
68     global todo_list, show_checked_var
69
70     # ctypes.windll library is only found on Windows systems.
71     try:
72         from ctypes import windll
73     except AttributeError:
74         pass
75     else:
76         windll.shcore.SetProcessDpiAwareness(1)
77
78     root = tk.Tk()
79     root.title("Yarusoto")
80     root.geometry("400x600")
81     dir_path = path.abspath(path.dirname(__file__))
82     icon = tk.PhotoImage(file=path.join(dir_path, "icon.png"))
83     root.iconphoto(False, icon)
84
85     # vars
86     show_checked_var = tk.BooleanVar()
87
88     # frames
89     todos_frame: ttk.Frame = ttk.Frame(root)
90     add_todo_frame: ttk.Frame = ttk.Frame(root)
91
92     # widgets
93     title_label: ttk.Label = ttk.Label(root, text="Welcome to Yarusoto!")
94     show_checked: ttk.Checkbutton = ttk.Checkbutton(
95         root,
96         text="Show completed to-dos",
97         variable=show_checked_var,
98         onvalue=True,
99         offvalue=False,
```

File - yarusoto.pyw

```
100         command=lambda: update_todos(todos_frame, show_checked_var.get())
101     )
102     seperator: ttk.Separator = ttk.Separator(root, orient="horizontal")
103     todos_label = ttk.Label(text="To-dos:")
104
105     # add_todos_frame widgets
106     add_todo_text: ttk.Entry = ttk.Entry(add_todo_frame, width=30)
107     add_todo_button: ttk.Button = ttk.Button(
108         add_todo_frame,
109         text="Add",
110         command=lambda: add_todo(add_todo_text, todos_frame)
111     )
112     root.bind("<Return>", lambda _: add_todo(add_todo_text, todos_frame))
113
114     # pack
115     title_label.pack()
116     show_checked.pack()
117     seperator.pack(fill="x", padx=5)
118     todos_label.pack(anchor="w")
119
120     todos_frame.pack(anchor="w")
121
122     add_todo_text.pack(side="left")
123     add_todo_button.pack(side="left")
124     add_todo_frame.pack(side="bottom")
125
126     # loop
127     root.mainloop()
128
129
130 if __name__ == "__main__":
131     main()
132
```