

File - yarusoto.pyw

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

File - yarusoto.pyw

```
58         todos_frame,
59         text=todo.name,
60         variable=todo.checked,
61         onvalue=True,
62         offvalue=False,
63         command=lambda: update_todos(todos_frame, show_checked_var.get())
64     ).pack(anchor="w")
65
66
67 # global vars
68 todo_list: List[TodoItem] = []
69 show_checked_var: Optional[tk.BooleanVar] = None
70
71
72 def main() -> None:
73     """
74     Manages tk initialization and adds the non-dynamic widgets.
75     """
76     global show_checked_var
77
78     # ctypes.windll library is only found on Windows systems.
79     try:
80         # noinspection PyUnresolvedReferences
81         from ctypes import windll
82     except ImportError:
83         pass
84     else:
85         windll.shcore.SetProcessDpiAwareness(1)
86
87     root = tk.Tk()
88     root.title("Yarusoto")
89     root.geometry("400x600")
90     dir_path = path.abspath(path.dirname(__file__))
91     icon = tk.PhotoImage(file=path.join(dir_path, "icon.png"))
92     root.iconphoto(False, icon)
93
94     # vars
95     show_checked_var = tk.BooleanVar()
96
97     # frames
98     todos_frame: ttk.Frame = ttk.Frame(root)
99     add_todo_frame: ttk.Frame = ttk.Frame(root)
100
101     # widgets
102     title_label: ttk.Label = ttk.Label(root, text="Welcome to Yarusoto!")
103     show_checked: ttk.Checkbutton = ttk.Checkbutton(
104         root,
105         text="Show completed to-dos",
106         variable=show_checked_var,
107         onvalue=True,
108         offvalue=False,
109         command=lambda: update_todos(todos_frame, show_checked_var.get())
110     )
111     seperator: ttk.Separator = ttk.Separator(root, orient="horizontal")
112     todos_label = ttk.Label(text="To-dos:")
113
114     # add_todos_frame widgets
115     add_todo_text: ttk.Entry = ttk.Entry(add_todo_frame, width=30)
```

File - yarusoto.pyw

```
116     add_todo_button: ttk.Button = ttk.Button(
117         add_todo_frame,
118         text="Add",
119         command=lambda: add_todo(add_todo_text, todos_frame)
120     )
121     root.bind("<Return>", lambda _: add_todo(add_todo_text, todos_frame))
122
123     # pack
124     title_label.pack()
125     show_checked.pack()
126     seperator.pack(fill="x", padx=5)
127     todos_label.pack(anchor="w")
128
129     todos_frame.pack(anchor="w")
130
131     add_todo_text.pack(side="left")
132     add_todo_button.pack(side="left")
133     add_todo_frame.pack(side="bottom")
134
135     # loop
136     root.mainloop()
137
138
139 if __name__ == "__main__":
140     main()
141
```