Laboratory Activity No. 10	
The Selection Widgets using Pycharm	
Course Code: CPE103	**Program:** BSCPE
Course Title: Object-Oriented Programming	**Date Performed:** 3 / 22 / 2025
Section: 1-A	**Date Submitted:** 3 / 22 / 2025
Name: Asugas, Kenneth R.	**Instructor:** Engr. Maria Rizette Sayo

1. Objective(s):

This activity aims to familiarize students with the Pycharm framework and selection widget

2. Intended Learning Outcomes (ILOs):

The students should be able to:
2.1 To create a Python program that use selection widget like Combobox
2.2 To use ttk function as part of Tk ( ) in the Tkinter module

3. Discussion:

A Graphical User Interface (GUI) application is a program that the user can interact with through graphics (windows, buttons, text fields, checkboxes, images, icons, etc..) such as the Desktop GUI of Windows OS by using a mouse and keyboard unlike with a Command-line program or Terminal program that support keyboard inputs only.

Pycharm is an integrated development environment used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems, and supports web development with Django.

4. Materials and Equipment:

Desktop Computer with Anaconda Python or Pycharm
Windows Operating System

5. Procedure:

```
# Creating tkinter window and set dimensions
window = tk.Tk()
window.title('Combobox')
window.geometry('500x250')


def choice(event):
    month = event.widget.get()
    print("Your birth month", month)


# label text for title
ttk.Label(window, text="Choose your birth month",
          background='light yellow', foreground="black",
          font=("Times New Roman", 15)).grid(row=0, column=1)
```

1.

```
month.grid(column=1, row=5)
month.current()

def choice(event):
    showinfo(
            title = "Selection",
            message = f'You selected{n.get()}')

month.bind("<<ComboboxSelected>>", choice)
window.mainloop()
```

2. Run the program and observe the output.

Adding an icon

3. Download any .ico picture from https://icon-icons.com/ or any similar sites.
4. Place the icon in your folder (ex. Oopfa1<lastname>_lab10)

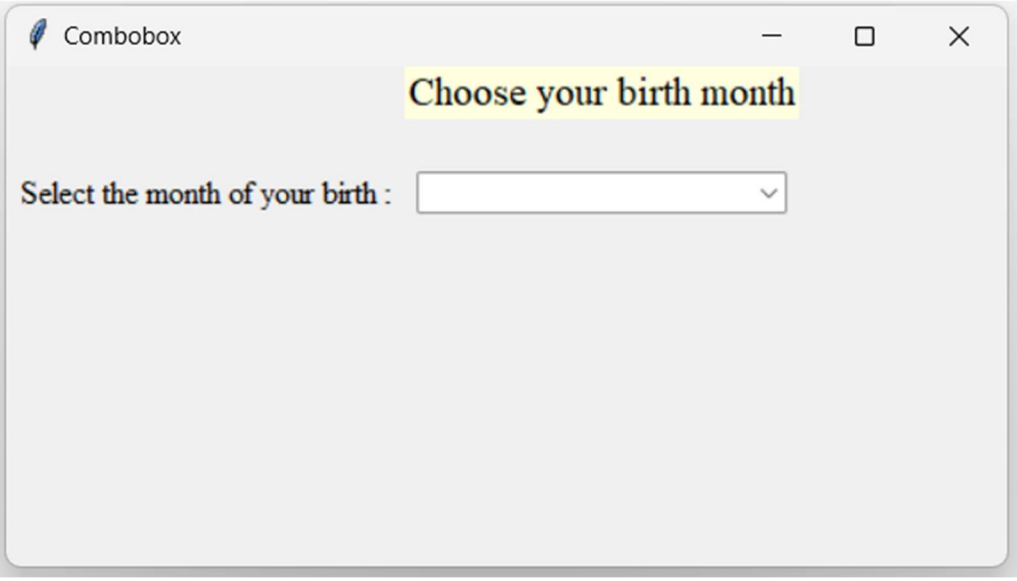
```
# Set label
ttk.Label(window, text="Select the month of your birth :",
          font=("Times New Roman", 12)).grid(column=0,
                                             row=5, padx=5, pady=25)

# Create Combobox
n = tk.StringVar()
month = ttk.Combobox(window, width=27, textvariable=n)

# Adding combobox drop down list
month['values'] = (' January',
                   ' February',
                   ' March',
                   ' April',
                   ' May',
                   ' June',
                   ' July',
                   ' August',
                   'September',
                   'October',
                   'November',
                   'December')
```

5. Run the program again, the program should now have an icon similar to the program below.

For the program please refer to this link: https://github.com/Kenneth-Asugas/CPE-103-OOP-1-A/blob/main/Laboratory%2010%20Procedure.py



6. Supplementary Activity:

Task

1. Create label widgets below to label your birth date <dd>, birth year <yyyy>
2. Create combobox to drop down your birth date <dd>, birth year <yyyy>
3. Create another method to show info about your birth date <dd>, birth year <yyyy>

Note: You may also use additional selection(listbox, radio button, check button) or common widgets to improve the design of your GUI.

For the program please refer to this link: https://github.com/Kenneth-Asugas/CPE-103-OOP-1-A/blob/main/Laboratory%2010%20Supplementary.py

Questions

1. What are selection widgets?

   Selection widgets are elements of a GUI which allow users to select from a set of options (eg. dropdowns, checkboxes or radio buttons). They allow for more natural and intuitive interaction with applications.

2. Which Python libraries provide selection widgets?

   Selection widgets exist in libraries like Tkinter, PyQt and Kivy, which offer different features for building user friendly interfaces.

3. How do selection widgets enhance user interaction in GUI applications?

   They make input simpler by providing predefined choices, eliminating human error, and making apps more efficient and visually appealing.

7. Conclusion:

Selection widgets are a critical part of the GUI design process: they make data entry easier, improve usability, and improve the overall user experience. With libraries like Tkinter and PyQt, developers can build efficient, interactive and customized interfaces for specific applications. This lab activity emphasizes the importance of understanding and using selection widgets to create robust and slick user interfaces. Pycharm provides an integrated environment to develop, debug, and test the GUI design process.

8. Assessment Rubric: