



**CHANDIGARH
UNIVERSITY**

Discover. Learn. Empower.



CASE STUDY

PYTHON PROGRAMMING

CODE – 24CAH-606



UNIVERSITY INSTITUTE *of*
COMPUTING
Asia's Fastest Growing University

MASTERS OF COMPUTER APPLICATION

Submitted By :

Name: Badal Kumar

UID: 24MCA20381

Class: 24MCA-6(A)

Submitted To:

Ms. Palwinder Kaur Mangat

Introduction

Objective:

The primary objective of this project is to create a graphical user interface (GUI) data entry form using Python's Tkinter library. This form collects and validates user information and course details, presenting a structured, user-friendly way to input and display data.

Purpose and Use Cases:

- The form serves as a prototype for collecting structured data.
- It could be used in applications requiring personal and course data collection, like educational enrollment or simple administrative software.

Key Features:

- Separate sections for user and course information, with validation checks to ensure data accuracy and completeness.
- Error handling for required fields and terms acceptance.

Application Design and Layout

GUI Structure:

- Main Window:** A simple Tkinter window titled "Data Entry Form."
 - **Frames and Sections:** Organized into three main frames:
 - **User Information Frame:** Collects personal details (first name, last name, title, age, nationality).
 - **Course Information Frame:** Includes course registration status, number of courses completed, and number of semesters.
 - **Terms and Conditions Frame:** Contains a checkbox to accept terms, required before submission.
- **Widgets Used:**
 - **Labels:** For field descriptions.
 - **Entry Fields:** For free text input (e.g., first and last names).
 - **Comboboxes and Spinboxes:** For drop-down selection of titles, nationality, and numeric input.
 - **Checkbuttons and Buttons:** For registration status, terms acceptance, and data submission.

Layout Management:

- Grid Layout:** The code uses grid for arranging widgets, which provides a structured and readable layout.
- Padding and Alignment:** Configured with padding to create visual separation and improve readability.

Code Walkthrough and Key Components

Imports and Initialization:

•The code imports necessary Tkinter modules, initializes the main Tkinter window, and sets up frames.

enter_data Function:

•**Purpose:** This function handles data retrieval, validation, and display.

• **Functionality:**

- Retrieves values from each input field and stores them in variables.
- Validates that required fields (first and last names) are filled and that terms are accepted.
- Displays data in the console if validation passes, or triggers an error popup if it fails.

User Information Frame:

•**Widgets:** Labels, Entries, Comboboxes, and Spinbox for user details.

•**Configuration:** Each field is labeled and spaced with padding for clear organization.

Course Information Frame:

•**Widgets:** Includes Checkbuttons for registration status, Spinboxes for numeric inputs (number of courses and semesters).

•**Logic for Checkbutton:** Uses StringVar for registration status, toggling between "Registered" and "Not Registered."

Terms and Conditions Frame:

•**Terms Acceptance:** Requires users to accept the terms before submitting.

•**Validation:** Uses a StringVar linked to the terms Checkbutton to track whether terms are accepted.

Submission Button:

•**Trigger Action:** Calls enter_data when clicked, initiating validation and data display.

Data Flow and User Interaction

User Interaction Flow:

- 1.Step 1:** User fills out personal information and course details.
- 2.Step 2:** User checks "Currently Registered" if applicable.
- 3.Step 3:** User must check the terms and conditions box.
- 4.Step 4:** Click "Enter Data" button to submit.

Data Validation:

- Checks that essential fields (first and last names) are completed and terms are accepted.
- If validation fails, a popup warning appears, guiding the user to fix the issue.
- On successful submission, the form prints the data to the console.

Error Handling:

- Missing Information Warning:** If names are missing or terms are not accepted, it displays an error using `messagebox.showwarning`.
- General Validations:** Only minimal checks are performed. No validation for numeric fields or non-required fields.

Strengths of the Tkinter Data Entry Form

Organized Layout:

- Clear sectioning (User Information, Course Info, Terms) improves readability and usability.
- Padding and grid configuration provide a structured interface.

User-Friendly Controls:

- Comboboxes and Spinboxes:** Simplify input by restricting user choices.
- Error Feedback:** Users get immediate feedback if required fields are missing, improving form usability.

Code Readability:

- Simple structure with clearly labeled widgets and logical organization.
- Easy for developers to understand and extend if needed.

Limitations and Drawbacks

Limited Validation:

- Only basic validation for names and terms acceptance. Other fields lack checks.
- Numeric fields (e.g., age, number of courses) allow any input within the set range, which could lead to errors.

Static, Non-Responsive Design:

- The form doesn't adjust to different screen sizes, which may limit usability on smaller or larger displays.



Data Output:

- Currently, the form only prints data to the console, with no option for saving or exporting data for later use.

Limited Error Handling:

- Error handling is limited to warnings for missing names or terms.
- More specific validation (e.g., age or number of courses within a reasonable range) would improve data accuracy.

Suggested Improvements

1. Enhanced Validation:

- Validate other fields like age, nationality, and course numbers.
- Ensure realistic values are entered (e.g., age within a reasonable range, courses and semesters within reasonable limits).

2. Data Storage and Export Options:

- Save data to a file (e.g., CSV, JSON) or store in a database to retain information.
- Add options to retrieve or edit saved data, enhancing functionality.

3. Code Modularization:

- Separate different sections (User Info, Course Info) into functions or classes for improved readability and modularity.
- Makes the codebase easier to extend or maintain.

4. Responsive Design and Improved UX:

- Use Tkinter's layout management options to create a responsive design.
- Enhance feedback with success confirmation messages upon successful data entry.

5. Additional Features:

- Add a reset button to clear the form for new entries.
- Include more detailed tooltips or instructions for fields.



**CHANDIGARH
UNIVERSITY**

Discover. Learn. Empower.



Execution:

Data Entry Form

User Information

First Name	Last Name	Title
<input type="text"/>	<input type="text"/>	<input type="text"/>
Age	Nationality	
<input type="text"/>	<input type="text"/>	

Registration Status # Completed Courses # Semesters

☐ Currently Registered 0 0

Terms & Conditions

☐ I accept the terms and conditions.

Enter data

Data Entry Form

User Information

First Name	Last Name	Title
<input type="text" value="Badal"/>	<input type="text" value="kumar"/>	<input type="text" value="Mr."/>
Age	Nationality	
<input type="text" value="21"/>	<input type="text" value="Indian"/>	

Registration Status # Completed Courses # Semesters

☒ Currently Registered

Terms & Conditions

☒ I accept the terms and conditions.

Enter data

```
PS D:\MCA PROJECTS\Python\tkinter-data-entry-  
"  
First name: Badal Last name: kumar  
Title: Mr. Age: 21 Nationality: Indian  
# Courses: 1 # Semesters: 1  
Registration status Registered  
-----
```



**CHANDIGARH
UNIVERSITY**

Discover. Learn. Empower.



Conclusion

The Tkinter data entry form is a functional and organized solution for simple data collection tasks. It effectively demonstrates core Tkinter capabilities, including layout management, widget usage, and basic validation. However, to make it production-ready, additional validation, modular design, and data persistence features are necessary.

Future Scope: Enhancing the form with features like responsive design, data storage, and comprehensive validation would make it a robust tool for real-world applications.



**CHANDIGARH
UNIVERSITY**

Discover. Learn. Empower.

