

Python Programming

Unit 04 – Lecture 03: Event Handling and Input Validation

Tofik Ali

School of Computer Science, UPES Dehradun

February 14, 2026

Repository: <https://github.com/tali7c/Python-Programming>

Quick Links

Core Concepts

Demo

Interactive

Summary

Agenda

1 Core Concepts

2 Demo

3 Interactive

4 Summary

Learning Outcomes

- Handle button clicks using callbacks

Learning Outcomes

- Handle button clicks using callbacks
- Bind keyboard/mouse events using `bind`

Learning Outcomes

- Handle button clicks using callbacks
- Bind keyboard/mouse events using `bind`
- Validate user input (required fields, numeric, email pattern)

Learning Outcomes

- Handle button clicks using callbacks
- Bind keyboard/mouse events using `bind`
- Validate user input (required fields, numeric, email pattern)
- Provide user-friendly feedback (labels, message boxes)

What is an Event?

- An event is a user action: click, key press, mouse move, window close

What is an Event?

- An event is a user action: click, key press, mouse move, window close
- Tkinter runs an event loop and calls your functions when events occur

Button Callback (command=...)

```
def submit():  
    print("Submitted")  
  
tk.Button(root, text="Submit", command=submit).pack()
```

Binding Events (`widget.bind`)

- `bind` provides an event object with details

```
def on_key(event):  
    print("Key:", event.keysym)  
  
entry.bind("<KeyRelease>", on_key)
```

Binding Events (`widget.bind`)

- `bind` provides an event object with details
- Useful for live validation and shortcuts

```
def on_key(event):  
    print("Key:", event.keysym)  
  
entry.bind("<KeyRelease>", on_key)
```

Validation Strategies

- Validate on submit (simplest)

Validation Strategies

- Validate on submit (simplest)
- Validate while typing (better UX)

Validation Strategies

- Validate on submit (simplest)
- Validate while typing (better UX)
- Validation types:

Validation Strategies

- Validate on submit (simplest)
- Validate while typing (better UX)
- Validation types:
 - required field (non-empty)

Validation Strategies

- Validate on submit (simplest)
- Validate while typing (better UX)
- Validation types:
 - required field (non-empty)
 - numeric range (age)

Validation Strategies

- Validate on submit (simplest)
- Validate while typing (better UX)
- Validation types:
 - required field (non-empty)
 - numeric range (age)
 - format (email using regex)

Good Validation Feedback

- Show clear messages: what is wrong and how to fix it

Good Validation Feedback

- Show clear messages: what is wrong and how to fix it
- Avoid technical tracebacks for end users

Good Validation Feedback

- Show clear messages: what is wrong and how to fix it
- Avoid technical tracebacks for end users
- Highlight the field or show a status label

Demo: Registration Form with Validation

- File: `demo/event_validation_demo.py`

Demo: Registration Form with Validation

- File: `demo/event_validation_demo.py`
- Demonstrates:

Demo: Registration Form with Validation

- File: `demo/event_validation_demo.py`
- Demonstrates:
 - button callback for submit

Demo: Registration Form with Validation

- File: `demo/event_validation_demo.py`
- Demonstrates:
 - button callback for submit
 - live validation using `bind`

Demo: Registration Form with Validation

- File: `demo/event_validation_demo.py`
- Demonstrates:
 - button callback for submit
 - live validation using `bind`
 - email validation using `regex`

Checkpoint 1

Question: When would you use `command=` vs `bind()`?

Checkpoint 2

Question: Why is input validation important in GUI applications?

Think-Pair-Share

Discuss:

- What is a good error message for an invalid email?
- What is a bad error message?

Key Takeaways

- Events drive GUI behavior

Key Takeaways

- Events drive GUI behavior
- `command=` is great for buttons; `bind` is great for key/mouse events

Key Takeaways

- Events drive GUI behavior
- `command=` is great for buttons; `bind` is great for key/mouse events
- Validation improves correctness and user experience

Key Takeaways

- Events drive GUI behavior
- `command=` is great for buttons; `bind` is great for key/mouse events
- Validation improves correctness and user experience
- Always provide clear feedback to the user

Exit Question

Write one validation rule for Age input in a registration form.