

Here's how you can create an interactive dashboard in Python using tkinter for a GUI interface and matplotlib to generate charts. This includes an input box for table feet and a switch button to display a dashboard with 10 different charts.

This title reflects the core functionality of the program: creating an interactive user interface with a table to display insights and a dashboard to visualize data with multiple charts.

```
In [1]: import tkinter as tk
        from tkinter import ttk
        import matplotlib.pyplot as plt
        from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
        import random

        # Function to create a dashboard with charts
        def open_dashboard():
            dashboard = tk.Toplevel()
            dashboard.title("Dashboard of Charts")

            fig, axes = plt.subplots(2, 5, figsize=(15, 6))
            axes = axes.ravel() # Flattening for easy iteration

            # Generating 10 random charts
            for i in range(10):
                x = range(10)
                y = [random.randint(0, 100) for _ in x]
                axes[i].plot(x, y)
                axes[i].set_title(f"Chart {i + 1}")

            plt.tight_layout()

            # Embedding matplotlib figure into tkinter
            canvas = FigureCanvasTkAgg(fig, master=dashboard)
            canvas.get_tk_widget().pack(fill=tk.BOTH, expand=True)
            canvas.draw()

        # Main application window
        def main():
            root = tk.Tk()
            root.title("Interactive Input Box and Dashboard")
            root.geometry("400x200")

            # Input Label and Box
            tk.Label(root, text="Enter Table Feet:").pack(pady=10)
            input_box = ttk.Entry(root)
            input_box.pack(pady=5)
```

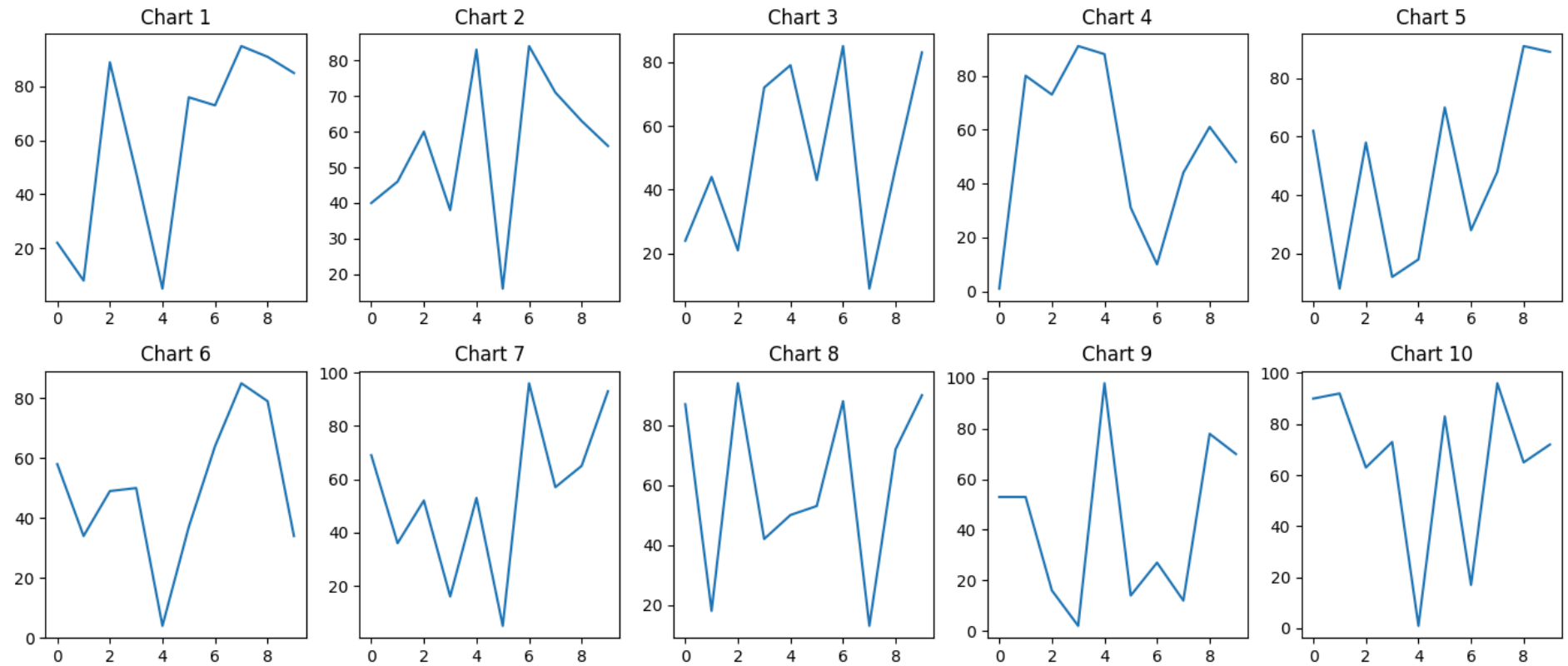
```

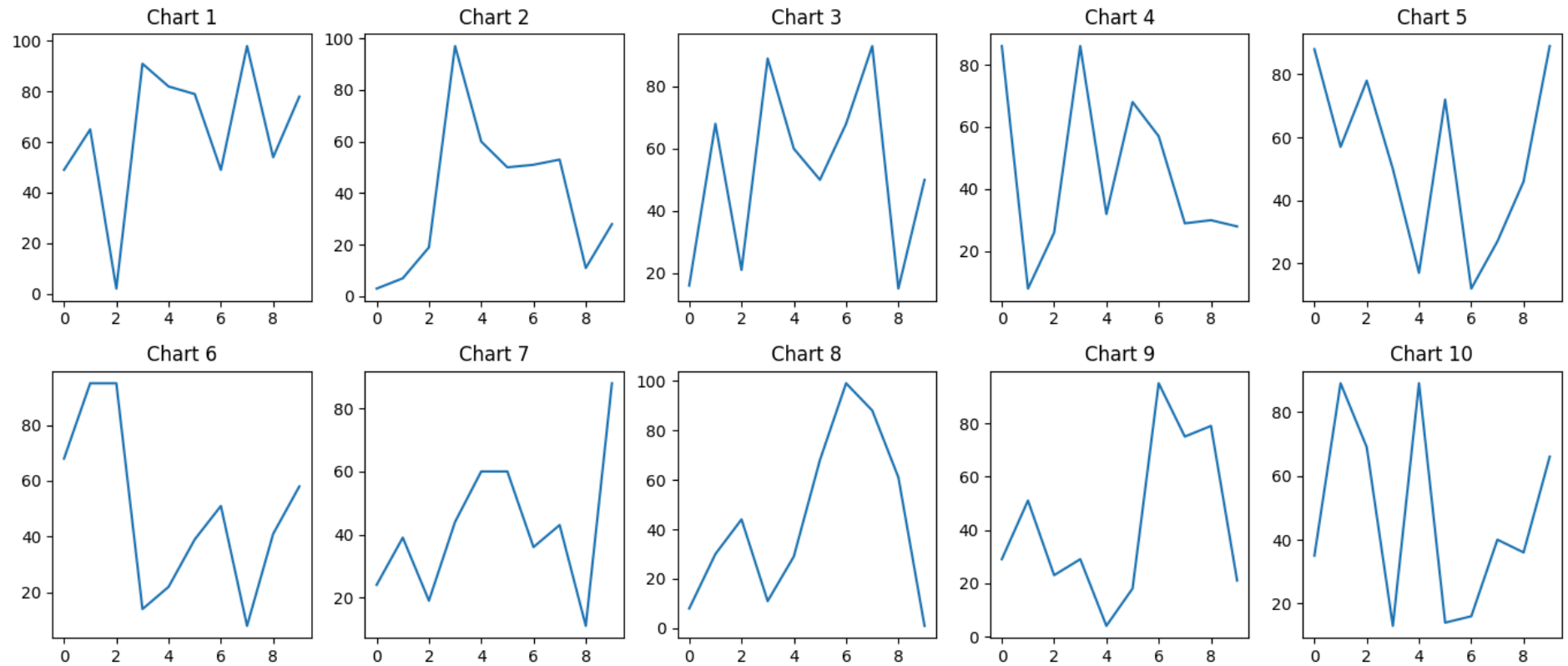
# Switch Button to Open Dashboard
switch_button = ttk.Button(root, text="Switch to Dashboard", command=open_dashboard)
switch_button.pack(pady=20)

root.mainloop()

if __name__ == "__main__":
    main()

```





Features: Input Box: Allows the user to enter text (e.g., table feet). Switch Button: Opens a new window with a dashboard containing 10 randomly generated charts. Matplotlib Integration: Generates charts dynamically using matplotlib. Interactive GUI: Utilizes tkinter for a user-friendly interface. How It Works: Run the script to launch the main window. Enter any input in the text box and click "Switch to Dashboard". A new window opens, displaying 10 charts. This script ensures no API-related errors (e.g., APY) as everything runs locally.

In []: