**Chapter 6**

In Chapter 6, I learned how to use buttons and text widgets to create interactive and user-friendly Flutter applications. The chapter began by introducing the **Text** widget, which is used to display static or dynamic text on the screen. I learned how to style text by customizing its properties, such as fontSize, fontWeight, color, and textAlign. The chapter also covered the use of Text.rich for combining multiple text styles within a single widget. Understanding these concepts helped me create visually appealing text elements that enhance the app's interface.

Next, I learned about Flutter's various button widgets, such as **ElevatedButton**, **TextButton**, and **OutlinedButton**, and their respective use cases. The chapter explained how to add functionality to buttons using the onPressed property, which triggers an action when the button is clicked. I practiced creating buttons with customized styles using the style property, where I adjusted colors, shapes, and padding. The chapter also introduced icons on buttons using the Icon widget, making them more interactive and visually distinct. These examples helped me understand how to combine buttons and text effectively in a user interface.

Finally, I learned how to handle user interactions by dynamically updating text when a button is pressed. The chapter demonstrated how to manage states in a stateful widget using the setState() method. For example, I created an app where clicking a button changed the displayed text, making the UI responsive. I also explored accessibility features, such as adding tooltips to buttons for better usability. By the end of the chapter, I felt confident in designing interactive apps with buttons and text while ensuring a seamless user experience.