**What is Widget?**

In Flutter, everything is a widget like button, a piece of text, a layout, or even the entire app — they’re all widgets. Think of widgets as building blocks. You assemble these blocks to construct your app’s user interface.

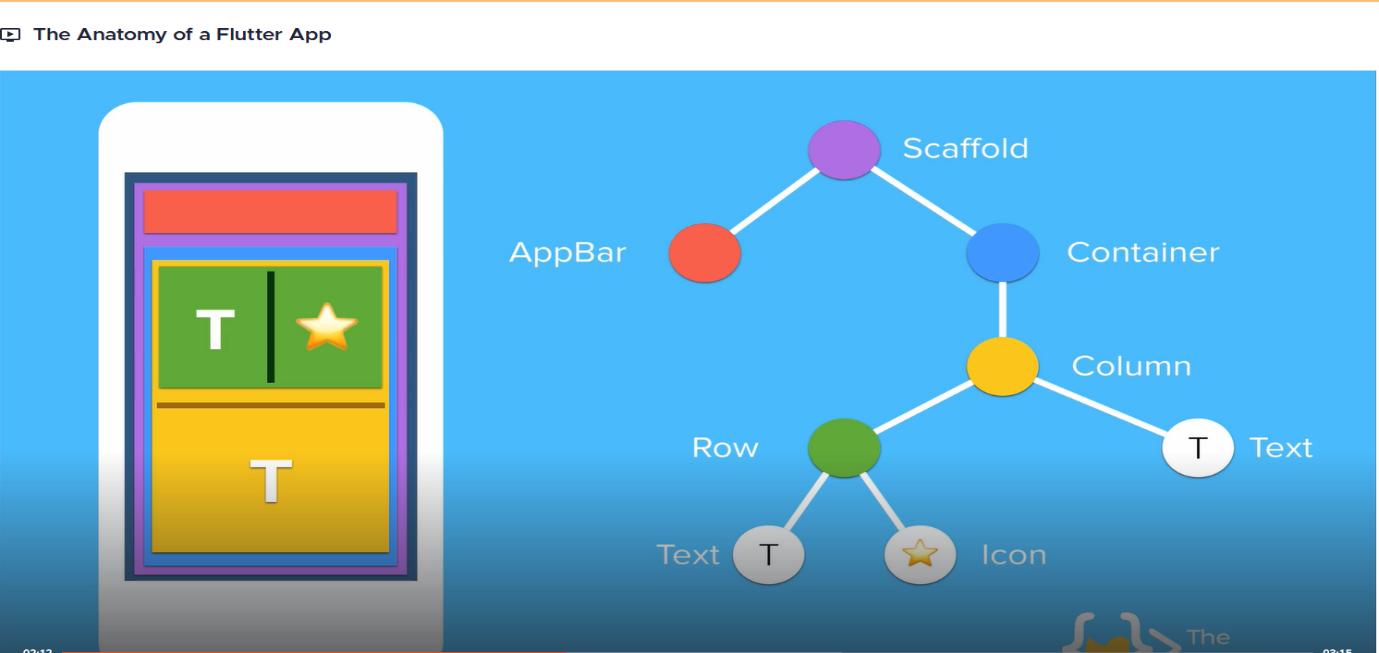
What is Widget Tree?

The widget tree is precisely what it sounds like — a tree structure where each node is a widget. This structure determines how your app’s UI is organized and displayed. Widgets are arranged hierarchically, forming a parent-child relationship.

The **widget tree** is the hierarchy or structure that organizes these widgets. It starts with a root widget and branches out to child widgets, each of which might have more child widgets.

Features of Widget Tree in flutter:

1. Widgets Rebuild: When something triggers a change, the affected widgets rebuild.
2. Diffing Process: Flutter’s engine performs a ‘diffing’ process, comparing the new widget tree with the previous one.
3. Efficient Updates: Flutter updates only the parts of the widget tree that have changed, making it incredibly efficient.

**Anatomy of the flutter App**

Widget Tree

Differences between the widgets hierarchy and the element and render
trees

**Widgets Tree ()**

In Flutter, the Widget Tree is a hierarchical structure of immutable objects that describes the layout and appearance of the user interface. Each widget in the tree represents a part of the UI, like a button or text, and the tree defines how these parts are organized and nested.

**Elements Tree**

A tree that represents the current state of the widgets and manages their lifecycle. Elements act as intermediaries between the widgets and the render objects.

**Render Tree**

In Flutter, the Render Tree is a hierarchy of `Render Object` instances responsible for the actual layout and painting of the user interface. It takes the layout constraints from its parent, calculates positions and sizes, and draws the UI elements on the screen.