Create A Responsive Flutter

1. Media Query

Media Query is useful if you need to decide based on the complete context rather than just the size of a particular widget

2. Layout Builder

simplified version of Media Query. The main difference between Media Query and Layout Builder is that Media Query is based on the full context of the screen rather than just the size of a particular widget; on the other hand, Layout Builder determines the maximum width and height of a specific widget only. Layout Builder class provides the Box Constraints object that can be used for determining the maxWidth and maxHeight of the widget.

3. Orientation Builder

You can get the Orientation object using the builder property of OrientationBuilder class. For example, you can use the OrientationBuilder class to change the number of columns of GridView.

4. Expanded and Flexible Widgets

Expanded and Flexible Widgets are two widgets that can be used inside Row, Column, or Flex to give their children the flexibility in order to expand to fill the available space. The only difference is that the Expanded widget requires the child to fill all the available space, whereas Flexible does not. Also, Expanded and Flexible widgets can be used to get a responsive Flutter UI that works with percentages instead of hardcoded values.

5. Aspect Ratio Widget

The Aspect Ratio widget can be used to size the child to a specific aspect ratio. When developing an app, one can neglect its size but must consider the aspect ratio. The AspectRatio widget thus assists you at this point by sizing the child value to a specific aspect ratio and ensures responsive design in Flutter.

7. Custom MutiChild Layout Class

widget that utilizes a delegate to specify the size and position of multiple children within it. This delegate is responsible for defining the layout constraints of each child and determining their positioning within the parent widget. While the delegate can define the size of the parent, it cannot be dependent on the size of its children.