## Exercise 2 - Layout Using Grid

* Goal - layout controls in a Grid
* Stuff to Know
  + Grid contains child controls
  + Grid contains RowDefinitions, with RowDefinition elements
  + Grid contains ColumnDefinitions, with ColumnDefinition elements
  + Default Height or Width is \*
  + Height/Width choices for grid rows/columns: \*, Auto, Fixed
  + Can using VerticalAlignment, HorizontalAlignment on contained controls
  + Use ComboBox control for dropdown list
  + Use ListBox control for normal list
  + Set ListBox items by including <ListBoxItem> elements, setting Content property for text
* Best Practices
  + Avoid using Fixed row and column sizes
  + One property per line
  + Attached properties (e.g. Grid.Row) come first
  + Set ShowGridLines to True only for debugging purposes

### The Exercise

* Create new project or use existing (after deleting older content)
* Create controls in a Grid
  + Create a grid with 5 rows and 3 columns
  + The first 2 columns should contain (one item in each row): A label, a dropdown list, a regular listbox, and a button. Items should have some space around them and be centered, left to right, in their columns
  + Dropdown list can remain empty
  + The 5th row should contain a label that stretches across the first 2 columns
  + When main window resizes
    - The third (empty) column should get bigger
    - The 3rd row (with listbox) should get bigger
  + Show grid lines while creating things, then turn them off
* Extra Credit
  + Add items to the two listbox controls
  + The third column should have a purple background and some text, rendered sideways, centered across all rows. The text should read from bottom to top.
  + Center the labels in the first row and make the text bold
  + Notice how things behave as you resize window. What happens when things get very small?