**Is Windows Forms dead?**

2002 .Net 1.0 Windows Forms Launched

2006 .Net 3.0 WPF Launched. In many ways the successor to Windows Forms.

2008 .Net 3.5 Last time significant changes were added to WinForms.

2014 .Net 4.5.2 WinForms bug fixes included. Related to high res screens.

Winforms is in maintenance only mode. The advantage is the framework is mature and stable. The model is not changing.

**Why use Windows Forms?**

ROI- return on investment. Companies have already invested time and money in learning WinForms and developing with it.

Existing apps: lots of code, lots of customers, too long to migrate.

Existing experience: Developers have invested time in becoming experts, avoid learning new UI paradigm

Existing libraries: bought or made that give a custom or distinct look and feel

1st best practice – give all controls and event handlers meaningful names.

**ListBox – Data binding**

Adding strings to represent an object in a separate list means that an update in one means the other has to be updated which could result in synching issues.

You can add objects to list boxes, not just strings, and use the ‘DisplayMember’ property of the listBox to use the string from any member of the object. E.g.

listBox.DisplayMember = “Title”;

This means a separate list is no longer needed, we can cast the object as we retrieve it from the listBox.

ListBox also has a DataSource property that enables a listBox to be populated without needing to iterate through and add each element. E.G.

listBox.DataSource = pod.Episodes;