

# Practical LINQ

Data Binding

Deborah Kurata

<http://msmvps.com/blogs/deborahk/>

deborahk@insteptech.com



**pluralsight**   
hardcore developer training

# Introduction

- **Windows Forms**
- **Windows Presentation Foundation (WPF)**
  - Silverlight
  - Windows 8
  - Windows Phone 8

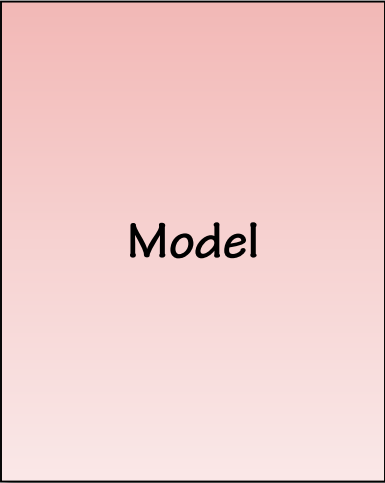
# Windows Forms

- **DataSource Property**
  - Binds data directly to a control on the user interface
- **BindingSource Control**
  - Separate control on the form
  - Bind each control to the BindingSource
  - Bind the data to the BindingSource
- **Goal**
  - Using the results of a LINQ query for binding

# Windows Presentation Foundation (WPF)

- **User Interface Framework**
- **Extensible Application Markup Language (XAML)**
  - Silverlight
  - Windows 8
  - Windows Phone 8
- **Model-View-View/Model (MVVM)**
- **Goal**
  - Using the results of a LINQ query for binding

# MVVM

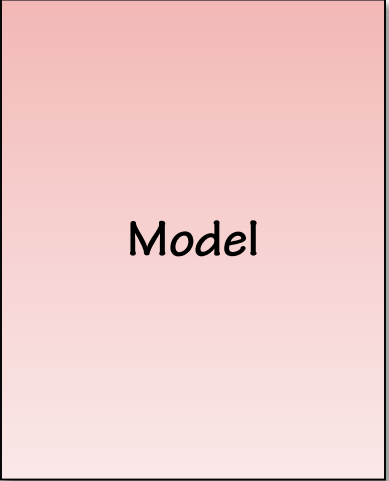


Model

# MVVM



View



Model

# MVVM



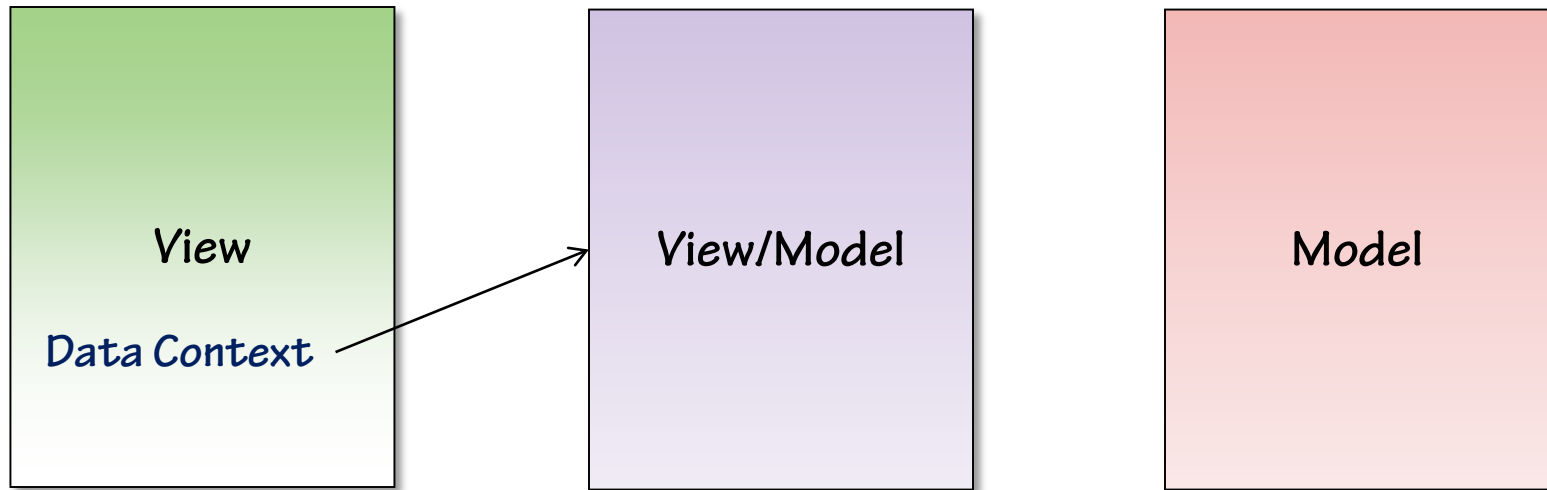
The diagram illustrates the MVVM (Model-View-ViewModel) pattern. It consists of three main components arranged horizontally: a green box on the left labeled 'View', a purple box in the center labeled 'View/Model', and a red box on the right labeled 'Model'. Each box has a black border and a subtle drop shadow. The 'View' box has a green-to-white gradient. The 'View/Model' box is a solid light purple. The 'Model' box is a solid light red.

View

View/Model

Model

# MVVM





# Summary

- **Windows Forms**

- DataSource
- ToList
- dynamic

- **Windows Presentation Foundation (WPF) / XAML-based UI**

- MVVM
- DataContext
- ObservableCollection<Customer>
- ObservableCollection<dynamic>
- ObservableCollection<CustomerModel>