* Course Overview
* Introduction
  + Autofac is one of the more modern DI containers for .NET
  + MVC and Web API
* Course Agenda
  + DI Primer/App Demo
  + Autofac Integration
    - NuGet packages
  + Registering Components
* Course Agenda, Continued
  + Injection & Resolve
  + Advanced patterns
    - Decorators
    - Asp.net filters
  + Additional scenarios
    - Owin
    - Web forms
    - Client-side libraries
* Begin Autofac Primer
* Overview of Primer App in Ra..
* Introduce Abstractions
* Library Class Modification
  + Have the class implement the interfaces
  + Use constructor to inject services
* Hard-to-Maintain Instantiations
* Install Autofac and Registration
  + Install autofac nuget package
  + Autofac is a two step container building process
  + Create container builder
    - Ex) ContainerBuilder builder = new ContainerBuilder();
  + Then register services that need to resolve DI
    - Ex) builder.RegisterType<AvengerRepository>().As<IAvengerRepository>();
    - Everytime container request IAvengerRepository it will return an instance of Avenger Repository class
  + Non abstracted type must also be registered
  + Then build container
    - Ex) IContainer container = builder.Build();
* Resolving the Service