**Ninject**

**Constructor Injection** (if ONLY 1 ctor, no attribute is needed. If more than 1 ctor, then ONLY 1 [Inject] attribute is allowed to put on 1 ctor)

[Inject]

    public Sale3(ITaxCalculator taxCalculator)  {

        this.taxCalculator = taxCalculator;

    }

**Property Injection**

[Inject]

    public ITaxCalculator TaxCalculator { get; set; }

**Method Injection** (the method will ONLY be called by the kernel once)

 [Inject]

    public void SetTaxCalculator(ITaxCalculator taxCalculator)   {

        this.taxCalculator = taxCalculator;

    }

**Init Binding /w Module**

    public class MainModule : NinjectModule  {

        private readonly bool designTime = false;

        public MainModule(bool designTime)         {

            this.designTime = designTime;

        }

**// Binding**

        public override void Load()       {

            if (designTime)  {

                Bind<IRepositoryService>().To<DesignRepositoryService>();

                Bind<EmployeeVM>().ToSelf().WithConstructorArgument("model", new EmployeeModel { BirthDate = DateTime.Now, …  });

            }

            else  {

                Bind<IRepositoryService>().To<RepositoryService>();

                Bind<EmployeeVM>().ToSelf();

            }

            Bind<IDialogService>().To<DialogService>().InSingletonScope();

            Bind<EmployeeMasterVM>().ToSelf();

        }

**// The Factory**

        public static IKernel Kernel { get; private set; }

// static ctor

       static Locator()  {

            if(Kernel ==  null)

                Kernel = new StandardKernel(new MainModule(ViewModelBase.IsInDesignModeStatic));

     }

// instance default ctor

        public Locator()   {

            EmployeeMasterVM = Kernel.Get<EmployeeMasterVM>();

        }

<https://github.com/ninject/ninject/wiki>

<http://stefanoricciardi.com/2011/01/21/ninject-mini-tutorial-part-1/>