В лабе нужно было переделать 3 лабу «Телефонный справочник» на Ninject

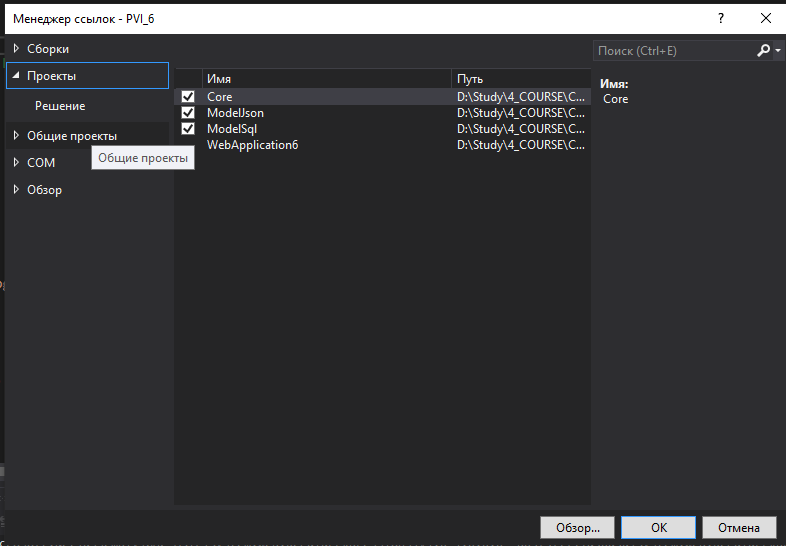
Проще будет вам внимательно прочитать лекцию Смелова 03\_06\_MVC\_DI\_Ninject.

Так же читаем вот это: https://metanit.com/sharp/mvc5/21.2.php

Для этого нужно было немного поменять архитектуру проекта и разбить на разные DLL-ки.

1. Core – ядро (сама модель).
2. ModelJson - обеспечивает хранения данных в json-файле (аналогично приложению, разработанному в лабораторной работе № 3). Репозиторий, является реализацией интерфейса **IPhoneDictionary**.
3. ModelSql - обеспечивает хранения данных в SQL-базе банных (аналогично приложению, разработанному в лабораторной работе 5. Репозиторий должен быть реализацией интерфейса **IPhoneDictionary (это ContactsHolderSql).**

Чтобы все это работало, нам нужно настроить ссылки в каждой DLL-ке и PVI\_6

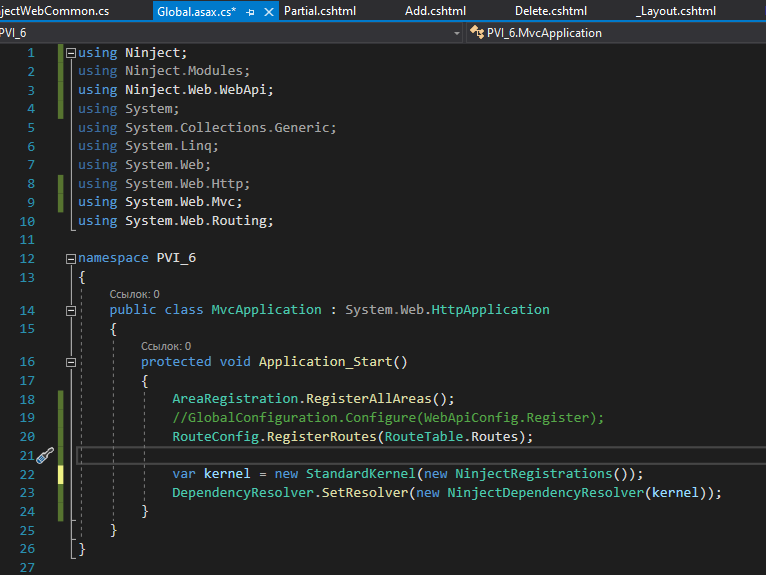
К примеру к решении PVI\_6 

NINJECT

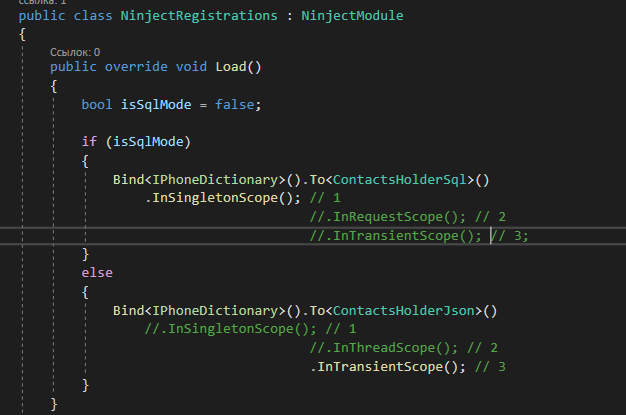
1. **Dependency Injection** –внедрение зависимости.
2. **DI:** программный механизм, позволяющий в автоматическом режиме создавать программный объект, с заданными жизненным циклом (задаются события инстансирования и разрушения объекта), способом применения (в качестве параметра метода или конструктора, свойства или поля объекта) и областью действия.
3. **DI:** позволяет создавать слабосвязанные компоненты.
4. **DI:** повторноеприменение кода, упрощает внесение изменений, упрощает тестирование.
5. **DI:**  чаще всего внедряется contextDB или репозиторий модели данных.

**DI:** IoC - Inversion of Control, принцип программирования, позволяющий снизить зависимость между компонентами программ; DI – один из способов реализации IoC.

глобальная регистрации зависимостей, **Global.asax**



Далее сам файл NinjectRegistrations



Здесь :

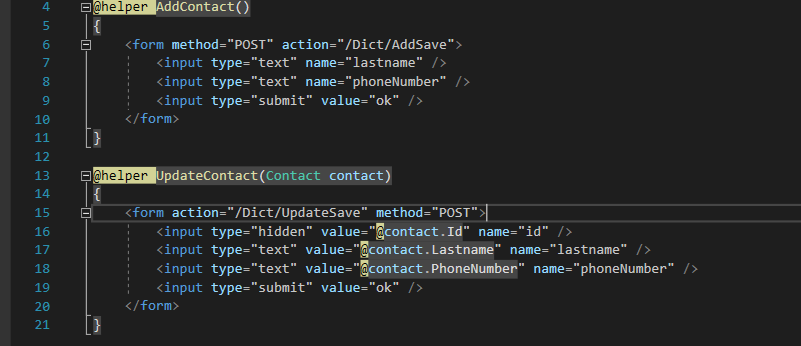
InTrasientScope – новый экземпляр на каждый поток

InRequestScope – на каждый запрос

InSingleton – один экземпляр на все вызовы.

Механизм dependency injection, позволяющий предать ссылку на объект репозитория (п.3), как параметр контролера. При этом, можно выбрать такой тип внедрения (InTrasientScope, InRequestScope или InSingleton)который создает свой объект репозитория для каждого внедрения.

html-helper:

файл MyHelpers.cshtml - реализует формы для ввода и корректировки строки справочника. Своего рода помощник для index страницы.

partial - представление, позволяющее отобразить одну строку справочника, используйте это представление в цикле вывода списка телефонов (файл Partial.cshtml).

**Задание 6** Ответьте на следующие вопросы

1. Поясните понятие «внедрение зависимости» (DI).

* программный механизм, позволяющий в автоматическом режиме создавать программный объект, с заданными жизненным циклом (задаются события инстансирования и разрушения объекта), способом применения (в качестве параметра метода или конструктора, свойства или поля объекта) и областью действия.

1. Для чего применяется DI?

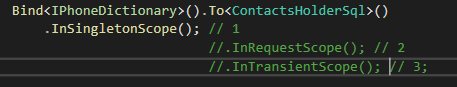
Чтобы в автоматическом режиме создавать программный объект, с заданными жизненным циклом

1. Что такое Ninject?

- IoC-контейнер, позволяющий внедрять зависимости

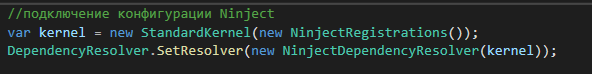
IoC - Inversion of Control, принцип программирования, позволяющий снизить зависимость между компонентами программ; DI – один из способов реализации IoC

1. Поясните принцип подключения и конфигурации Ninject в ASP.NET-приложении.

Сначала нужно написать класс, расширяющий класс NinjectModule. Далее создать объект этого класса и установить отношения между интерфейсами и их реализациями. 

Данное выражение указывает, что объекты IPhoneDictionary должны будут рассматриваться как ContactsHolderSql.

Подключение конфигурации происходит в Global.asax



1. Каким образом может быть выполнено DI в ASP.NET-приложении.

Не совсем понял вопроса, но я думаю, что они хотят услышать «Глобальная регистрация зависимостей».

1. Назовите четыре вида DI, которые позволяет сделать Ninject в ASP.NET-приложении и поясните их особенности.

InTrasientScope – новый экземпляр на каждый поток

InRequestScope – на каждый запрос

InSingleton – один экземпляр на все вызовы.

InThreadScope - Один экземпляр типа будет создан для каждого потока. (то же самое, что и InTrasientScope, как я понял)