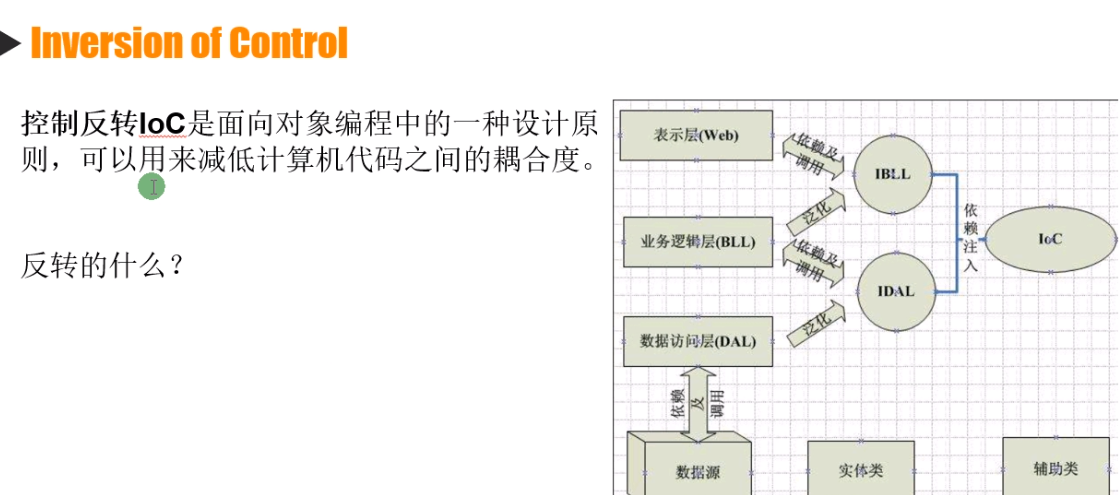
 例如更换数据库的dal,bll等等，改一处配置文件就行了，因为用的是反射获取的dal,bll



//抽象不能直接实例化，但是又不要细节，，所以使用第三方 反射。

具体方法

1. 在项目的appsettings.json中添加 程序集名称和具体实现类

{

"IUserDAL": "Zhaoxi.IocDI.DAL.UserDAL,Zhaoxi.IocDI.DAL",

"IUserBLL": "Zhaoxi.IocDI.BLL.UserBLL,Zhaoxi.IocDI.BLL"

}

创建第三方类库

1. 获取配置文件公共方法

public class ConfigurationManager

{

private static IConfigurationRoot \_configurationRoot;

static ConfigurationManager()

{

var builder = new ConfigurationBuilder().SetBasePath(Directory.GetCurrentDirectory()).AddJsonFile("appsettings.json");

\_configurationRoot = builder.Build();

}

public static string GetNode(string nodeName)

{

return \_configurationRoot[nodeName];

}

}

1. 创建反射，获取appsetting中的程序集

public static IUserDAL CreateUserDAL()

{

IUserDAL userDAL = null;

//反射创建DAL实例

string config = ConfigurationManager.GetNode("IUserDAL");

Assembly assembly = Assembly.Load(config.Split(',')[1]);

Type type = assembly.GetType(config.Split(',')[0]);

userDAL = (IUserDAL)Activator.CreateInstance(type);

return userDAL;

}

public static IUserBLL CreateUserBLL(IUserDAL userDAL)

{

IUserBLL userBLL = null;

//反射创建DAL实例

string config = ConfigurationManager.GetNode("IUserBLL");

Assembly assembly = Assembly.Load(config.Split(',')[1]);

Type type = assembly.GetType(config.Split(',')[0]);

userBLL = (IUserBLL)Activator.CreateInstance(type,new object[] { userDAL });

return userBLL;

}

}

2 调用 (添加UserDAL,UserBLL类库)

IUserDAL userDAL = ObjectFactory.CreateUserDAL();

IUserBLL userBLL = ObjectFactory.CreateUserBLL(userDAL);

UserModel userModel = userBLL.Login("123");