平台功能使用说明

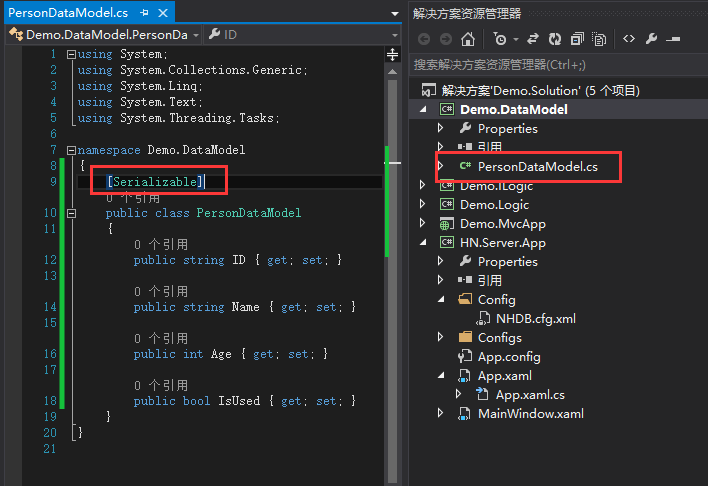
**开发环境：VS2013+WIN10**

**文档版本：v1.0**

# 开发流程及要点

## 添加模型

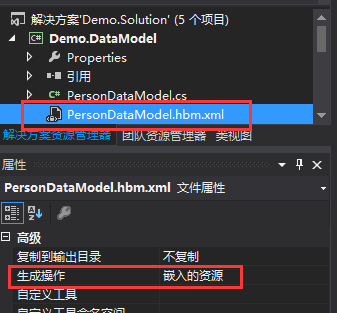
**【要点】模型添加可序列化标记**



## 添加数据库映射文件

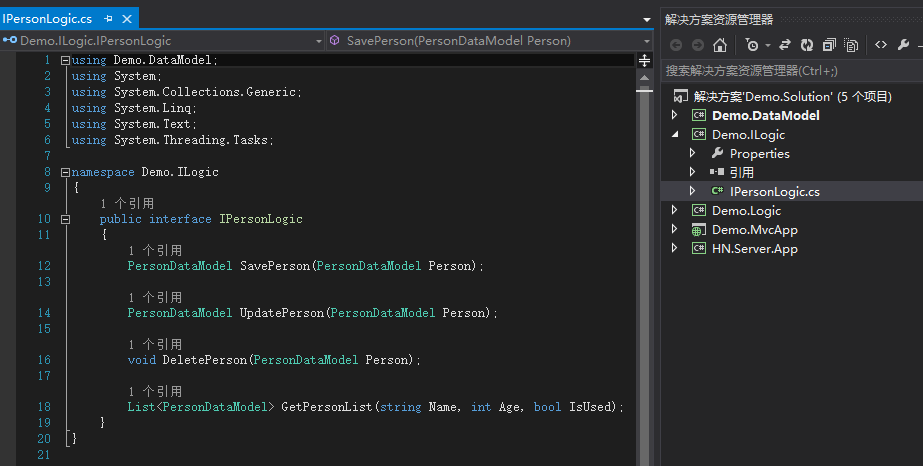
**【要点】1、文件名字为 模型名字+.hbm.xml；**

**2、文件生成操作为 嵌入的资源。**



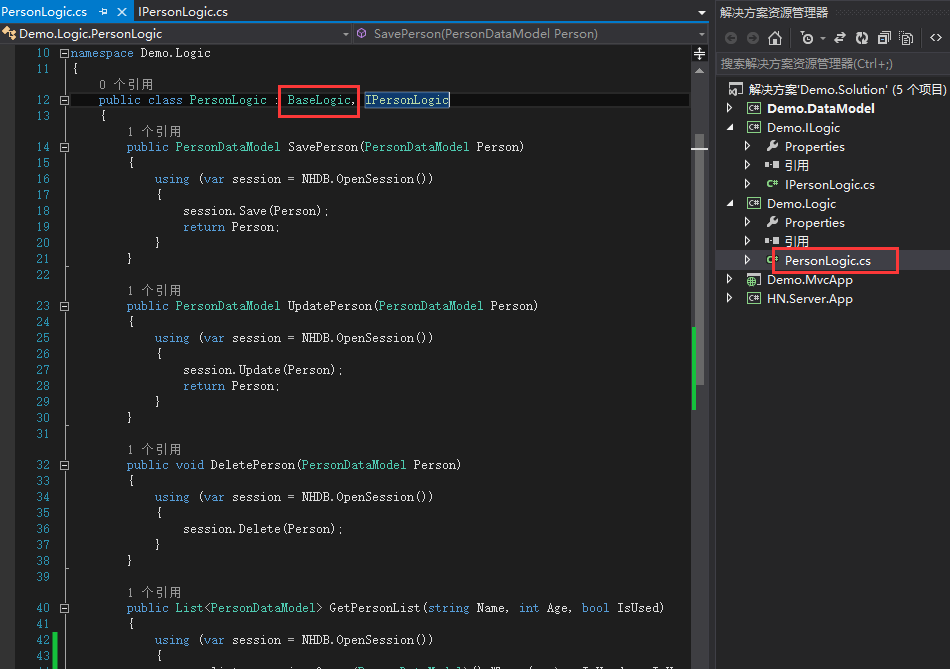
## 添加接口及定义方法

**【要点】接口名称尽量保持统一**



## 实现接口

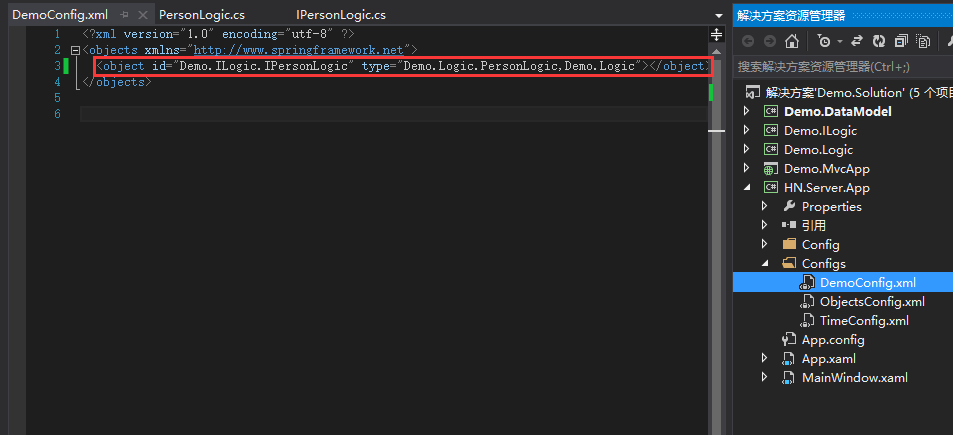
**【要点】继承BaseLogic类，该类提供了Nhibernate方法**



## 在服务端添加映射配置

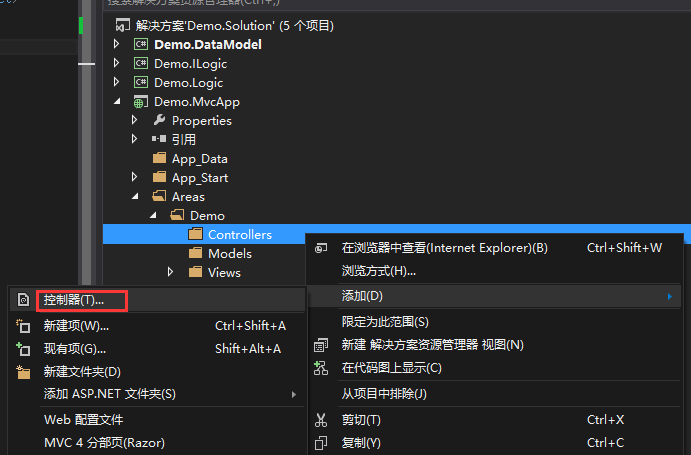
**【要点】id规范：接口的命名空间+接口名称；**

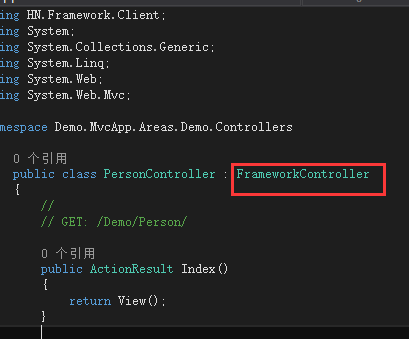
**type规范：类的命名空间+类名称+，+程序集名称。**



## 右键添加控制器

**【要点】控制器继承FrameworkController**

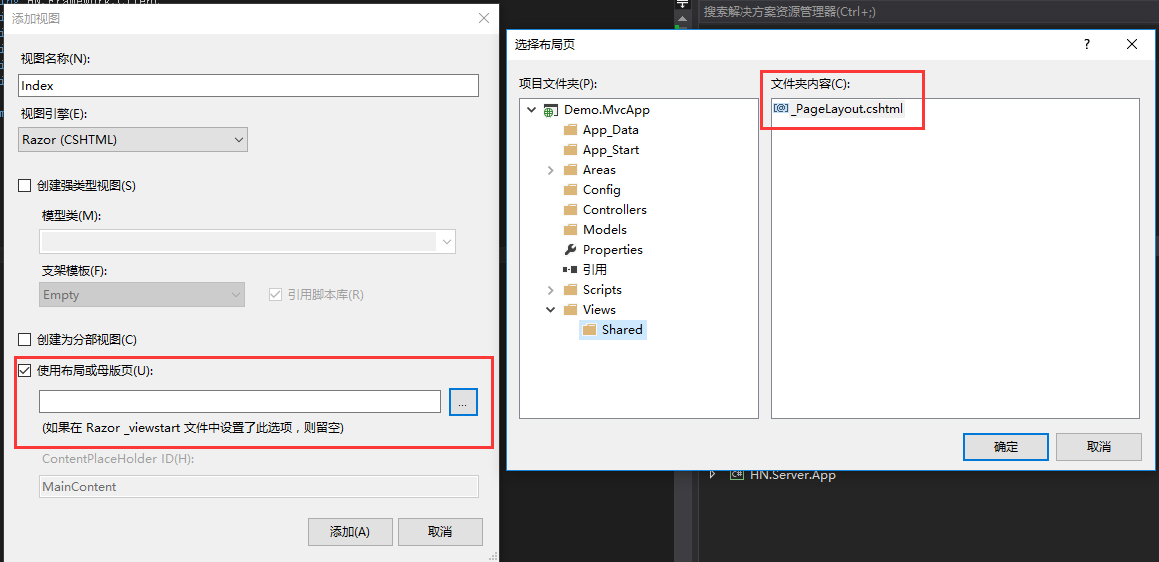




## 在Index上点右键添加视图，并选择母板页

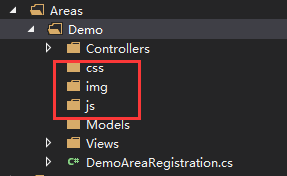
**【要点】选择正确的母版页**

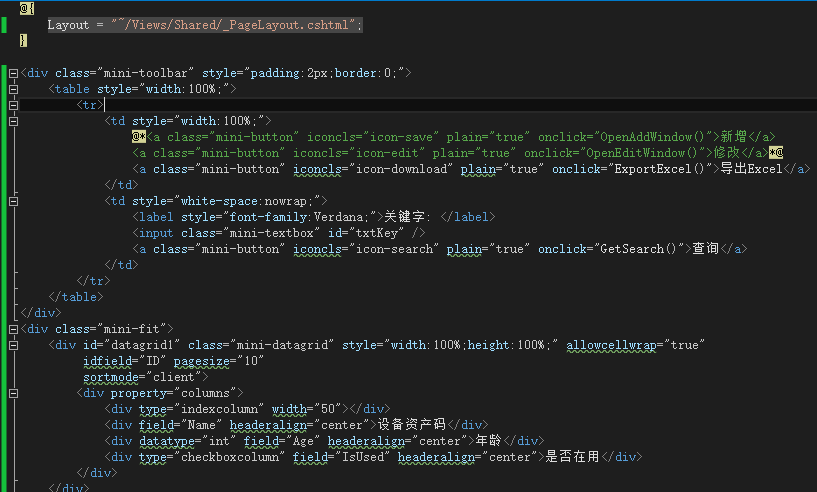


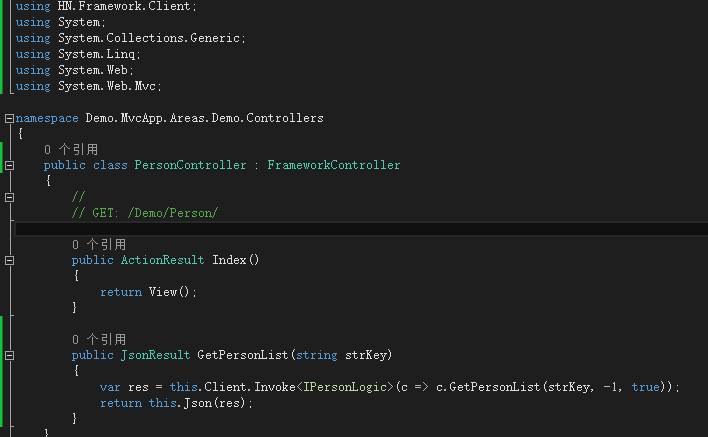


## 在视图上写相应代码，在控制器里添加相应方法

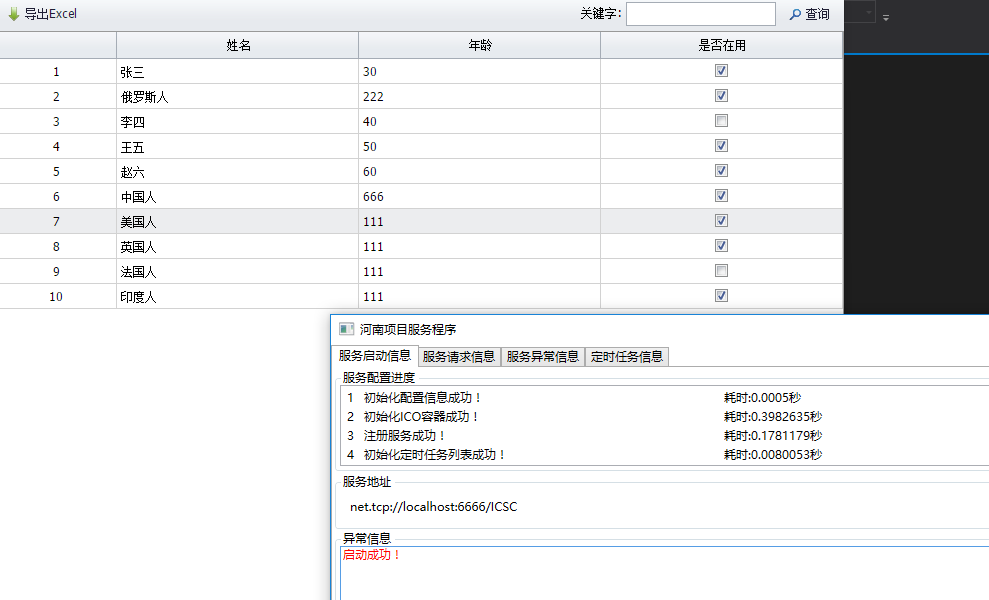
【注意】视图上所引用的非公共的JS文件、CSS样式、图片等，放到自己的模块区域里







## 启动服务和MVC应用，即可看到效果



# 前台框架功能使用说明

## 2.1 framwork.js功能说明

### app. DataRequest

【方法描述】

通过ajax调用Controller里的方法

【参数说明】

url：方法的地址，若为当前视图所属的控制器，只需要写方法名即可

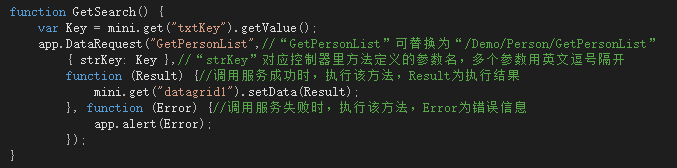
data：调用方法时传递的参数

succeed：成功时调用的方法

fail：失败时调用的方法

async：是否锁死浏览器，一般不传该参数，默认为false

【举例】



### app.QueryPara

【方法描述】

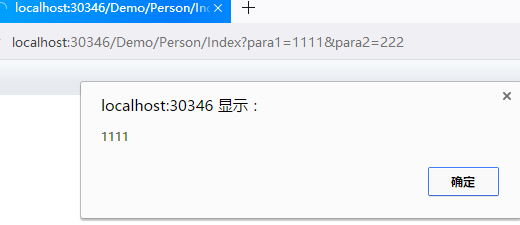
获取url参数

【参数说明】

name：参数名

【举例】





### app. ChineseABC

【方法描述】

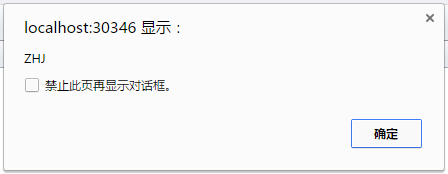
获取汉字首字母拼音

【参数说明】

str：字符串

【举例】





### app. alert

【方法描述】

弹出提示框

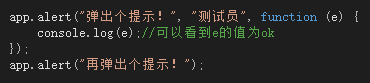
【参数说明】

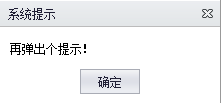
message：消息内容

title：提示框标题，可为空，默认值为：系统提示

callback：点击按钮后执行的方法

【举例】



### app. confirm

【方法描述】

弹出确认框

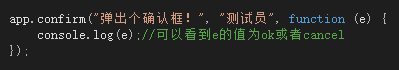
【参数说明】

message：消息内容

title：提示框标题，可为空，默认值为：系统提示

callback：点击按钮后执行的方法

【举例】





### app. GetJsonValue

【方法描述】

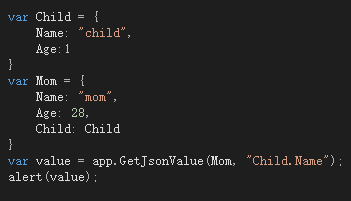
根据路径获取Json值

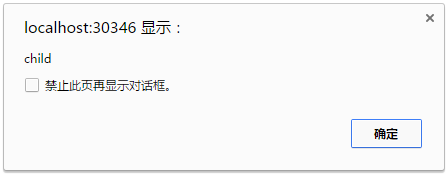
【参数说明】

JsonObject：Json对象

Path：路径

【举例】





### app. ExportExcel

【方法描述】

将miniui的grid数据导出到Excel【仅支持miniui，若用其它UI框架，虚重写该方法】

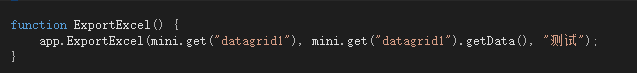
【参数说明】

grid：miiui的grid对象

jsonData：要导出的json数据

fileName：导出的文件名

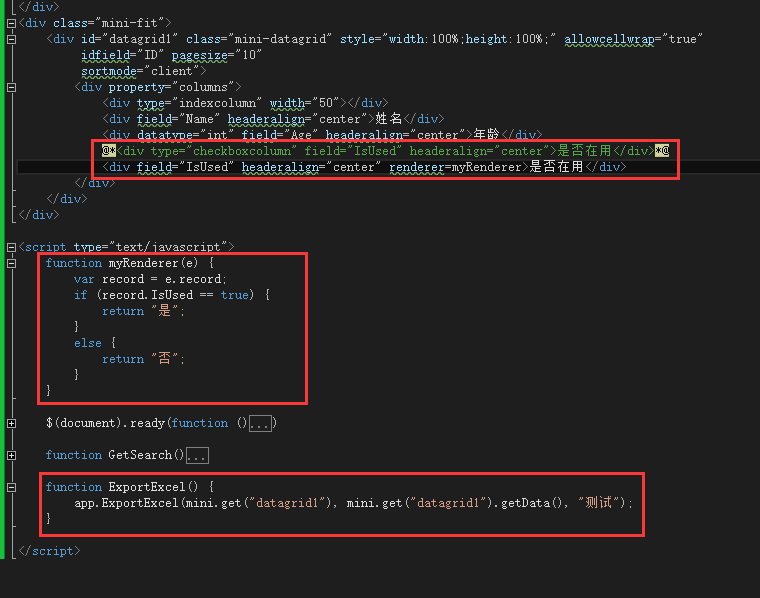
【举例】







**【注：是否在用列为特殊非文字列，暂不支持，改一下显示方式即可，如下图】**



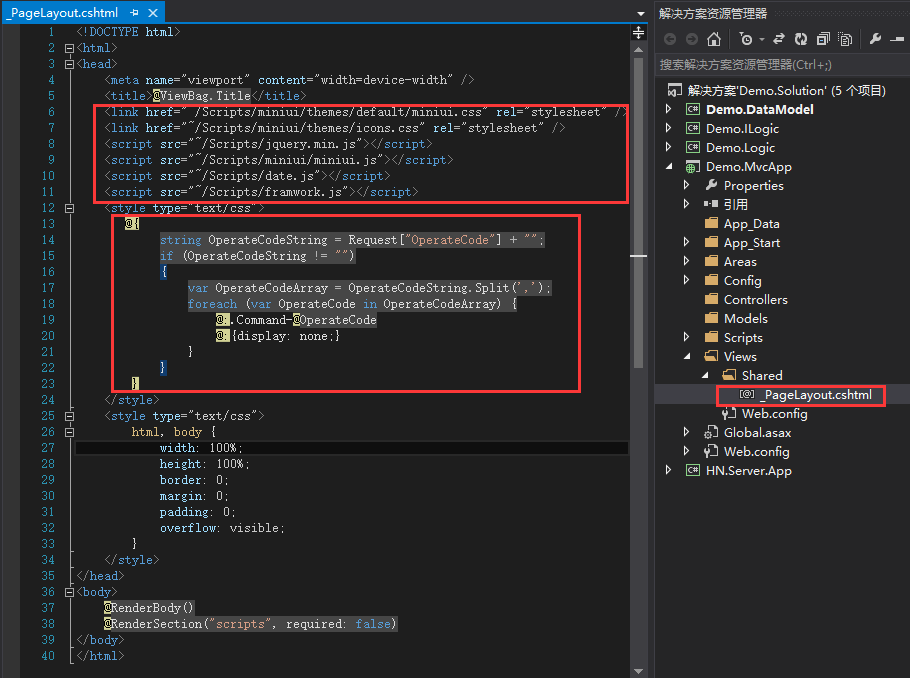


## 2.2 母版页功能说明

1、可根据项目需要修改母版页引用的样式或js文件

2、可以修改按钮权限控制代码

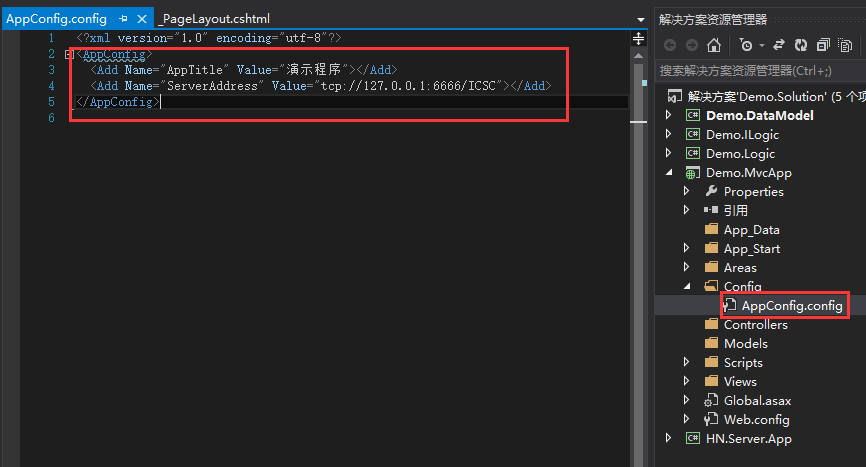
3、可根据项目需要增加多个母版页



## 2.3 MVC配置说明

AppTitle：标题【Portal框架所用，普通功能开发用不到，读取地址：GlobalManager. AppTitle】

ServerAddress：调用服务的地址



## 2.4 FrameworkController功能说明

### this.Client

【方法描述】

用来调用业务方法，并返回执行结果ServiceResponse。

ServiceResponse说明：

IsSuccess（bool）：执行是否成功

data（object）：返回值

total（int）：预留，miniui可能会用到

Error（string）：执行不成功时错误信息

Duration（double）：执行时间（秒）

【参数说明】

T：接口对象

methodExpression：执行方法及参数

【举例】



### this. GetUserData

【方法描述】

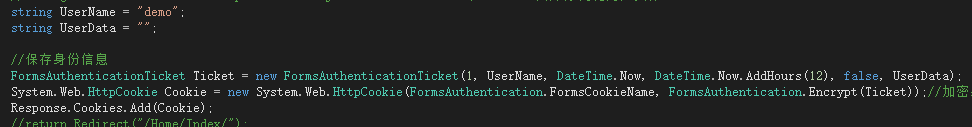
得到登录用户信息【注：前提是已登录，并保存了相关信息】

【参数说明】

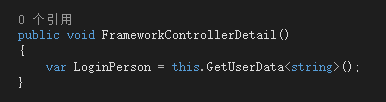
T：返回的数据类型（与保存的类型需一致）

【举例】

//保存用户信息



//得到用户信息



### this. LogDebug、this. LogInfo、this. LogWarm、this. LogError

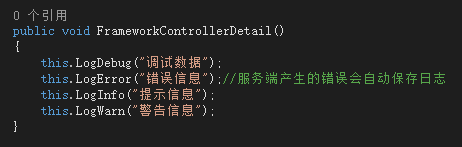
【方法描述】

保存不同类型的日志信息

【参数说明】

Log：日志内容

【举例】



## 2.5 IFrameworkLogic方法说明

### LogDebug、LogInfo、LogWarm、LogError

【方法描述】

保存不同类型的日志信息【注：已集成到FrameworkController里，js端可直接调用】

【参数说明】

Log：日志内容

### 简单查询GetObject

【方法描述】

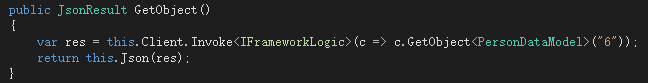
根据主键查询对象，返回唯一Object对象

【参数说明】

T：查询对象

ID：主键值

【举例】



### 简单查询GetObjectList

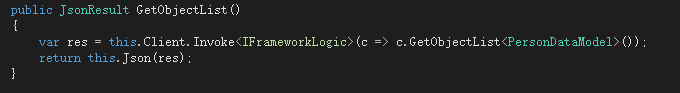
【方法描述】

根据主键查询对象，返回所有Object对象

【参数说明】

T：查询对象

【举例】



### 简单保存SaveObject

【方法描述】

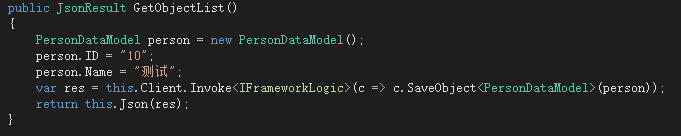
保存对象，返回保存后的对象

【参数说明】

T：保存对象

Entity：对象实例

【举例】



### 简单保存SaveObjectList

【方法描述】

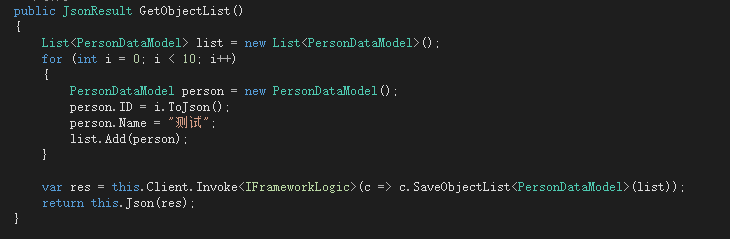
保存对象集合，无返回结果

【参数说明】

T：保存对象

EntityList：对象实例集合

【举例】



### 简单修改UpdateObject

【方法描述】

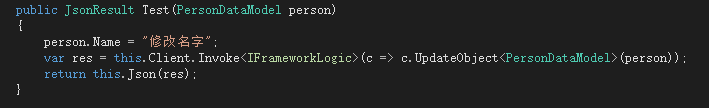
修改对象，返回保存后的对象

【参数说明】

T：修改对象

Entity：对象实例

【举例】



### 简单修改UpdateObjectList

【方法描述】

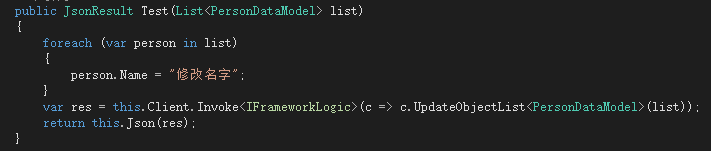
修改对象集合，无返回结果

【参数说明】

T：修改对象

EntityList：对象实例集合

【举例】



### 简单删除DeleteObject

【方法描述】

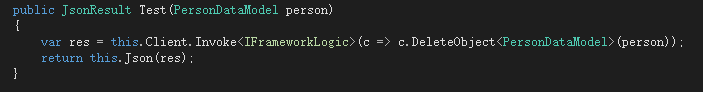
删除对象，无返回结果

【参数说明】

T：删除对象

Entity：对象实例

【举例】



### 简单删除DeleteObjectList

【方法描述】

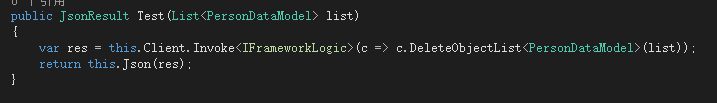
删除对象集合，无返回结果

【参数说明】

T：删除对象

EntityList：对象实例集合

【举例】



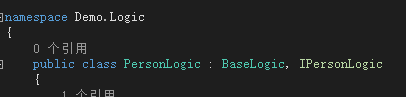
# 后台框架功能使用说明

## 3.1 BaseLogic功能说明

【描述】

业务逻辑基类

【举例】



### NHDB

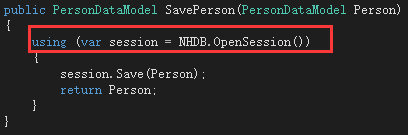
【属性描述】

根据默认路径（/Config/NHDB.cfg.xml）返回INHDataBase对象

【参数说明】

无

【举例】



### GetNHDB

【方法描述】

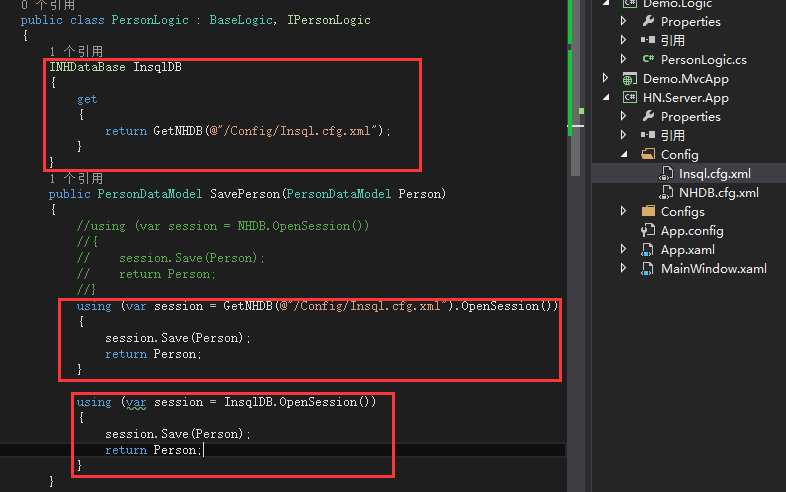
根据指定路径返回INHDataBase对象

【参数说明】

ConfigPath：路径地址【注：配置文件一般放在Config下】

【举例】

两种调用方式，一种是直接使用方法，一种是定义一个INHDataBase属性



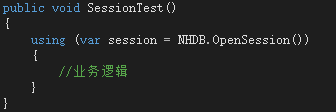
## 3.2 INHDbSession对象说明

【描述】

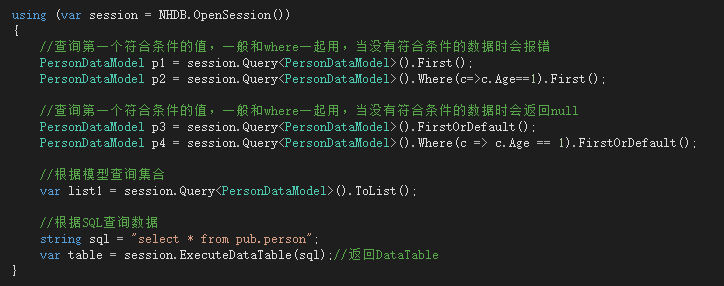
INHDbSession对象由INHDataBase对象OpenSession()而来。

一般和using一起使用，在使用后自动释放内存。

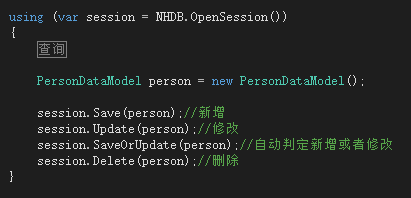
【举例】



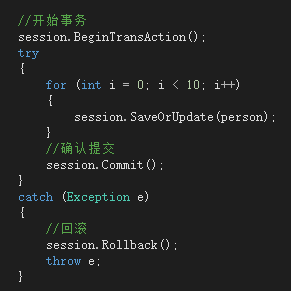
### 查询常用的方法



### 增删改



### 事务



## 3.3 CacheManager功能说明

### SetCache

【方法描述】

设置缓存，若缓存已存在，则会覆盖之前的缓存数据

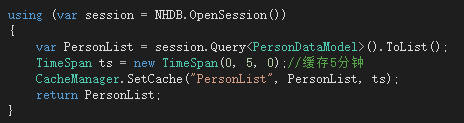
【参数说明】

CacheName：string类型，缓存名称标识

CacheData：object类型，缓存内容

ExpireTime：TimeSpan类型，缓存时间

【举例】



### GetCache

【方法描述】

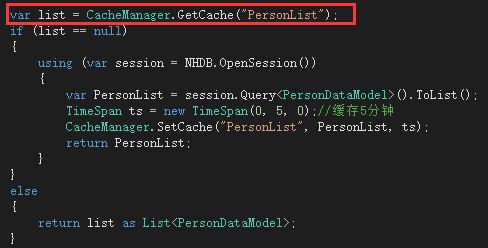
获得缓存内容，无缓存内容或缓存过期时，返回null

【参数说明】

CacheName：string类型，缓存名称标识

【举例】

当缓存为空时，查找数据并设置缓存，缓存有数据时直接返回结果



### DeleteCache

【方法描述】

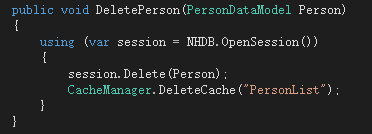
主动删除缓存内容

【参数说明】

CacheName：string类型，缓存名称标识

【举例】

当数据发生变化时，需要删除缓存，以免缓存数据与数据库不一致



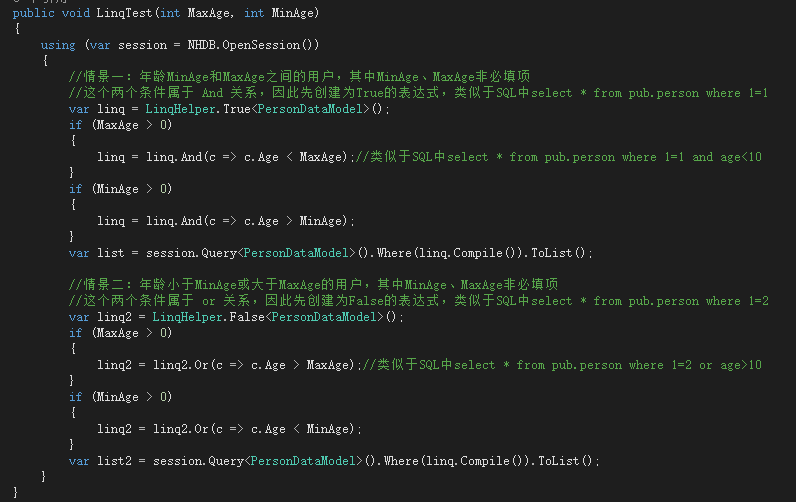
## 3.4 LinqHelper功能说明

【描述】

命名空间：HN.Common

拼接Linq表达式，当有多个查询条件时，以前的做法是，先将所有数据查询出来，再根据条件一个一个筛选，效率不高并消耗大量内存；现在可以通过LinqHelper将查询条件处理好，一次性将结果查出来。

【举例】

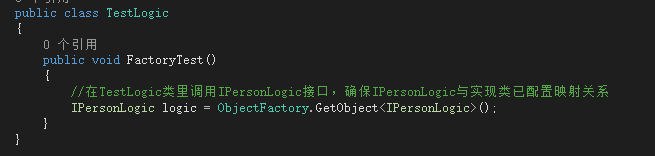


## 3.5 ObjectFactory功能说明

【描述】

在业务逻辑里调用其它接口时，使用该类获取接口对应实例

【举例】



## 3.6 Json功能说明

【描述】

命名空间：HN.Common

将对象转化为Json，或者将Json转化为对象

【举例】



# 相关配置介绍

## 4.1 服务端App.config

【参数说明】

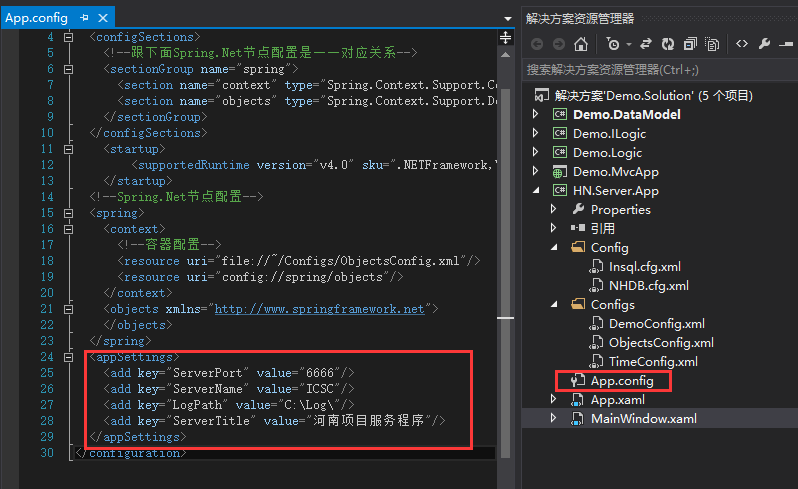
ServerPort：服务所用端口

ServerName：服务所用名称

LogPath：日志保存地址

ServerTitle：服务程序标题

【举例】



## 4.2 服务端TimeConfig.xml

【功能描述】

定义定时执行的方法

【参数说明】

Interface：接口路径

Method：方法名称（不能带参数）

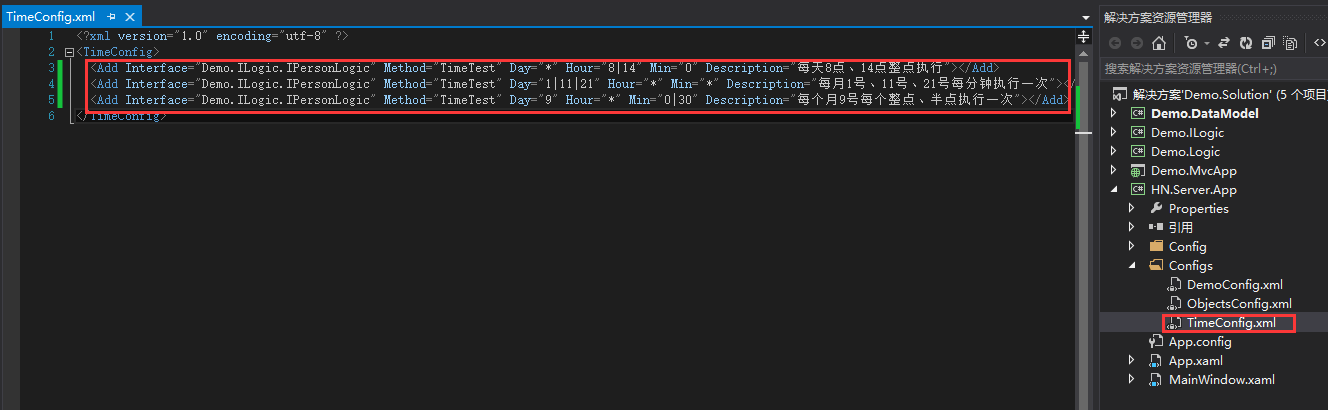
Day：【\*】代表每天，指定日期直接写多个用【|】隔开，一个不用隔开，从小到大写

Hour：【\*】代表每时，指定小时直接写多个用【|】隔开，一个不用隔开，从小到大写

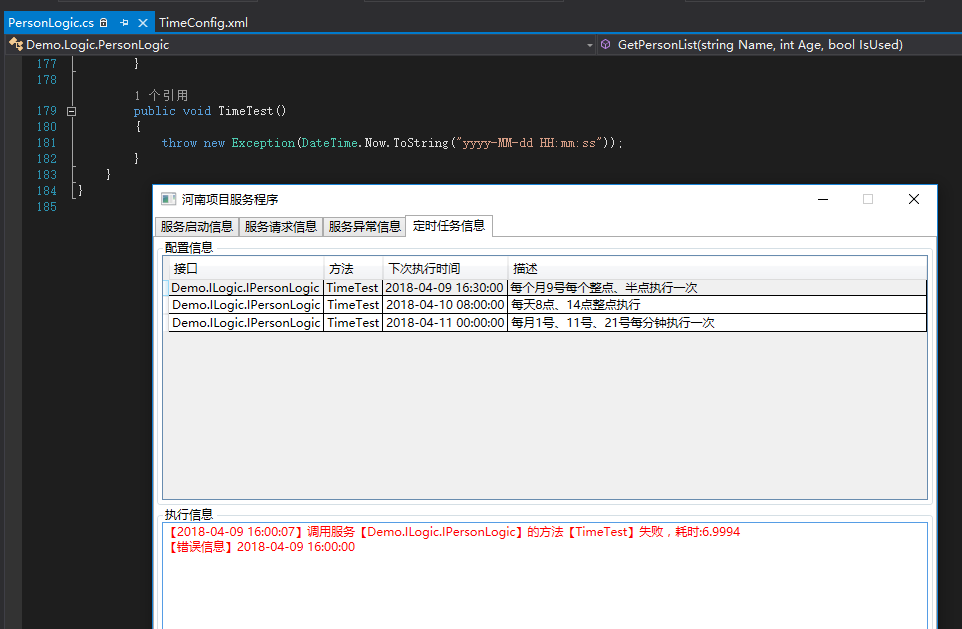
Min：【\*】代表每分，指定分钟直接写多个用【|】隔开，一个不用隔开，从小到大写

Description：方法描述

【举例】



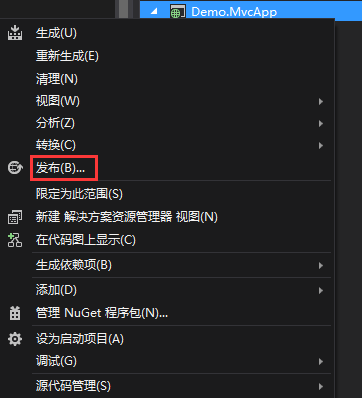
因调试模式在抛出异常时中断了，所以耗时高



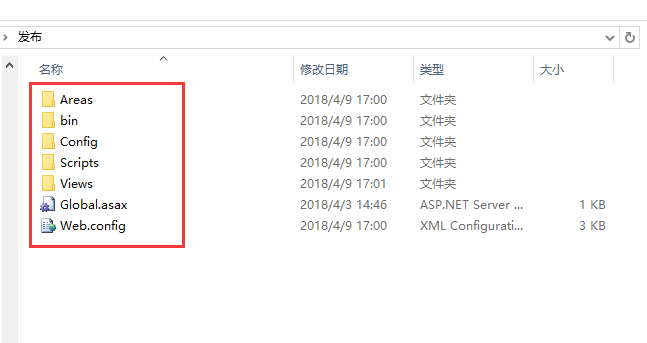
# 发布

## 5.1 前台发布（框架未发布）

1. 在MVC项目上点右键》发布，使用文件发布



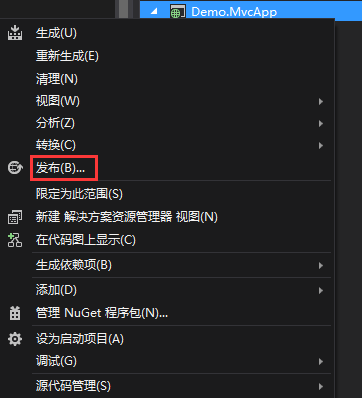
1. 将所有文件拷贝至IIS服务器上，记得改AppConfig.config里的服务请求地址



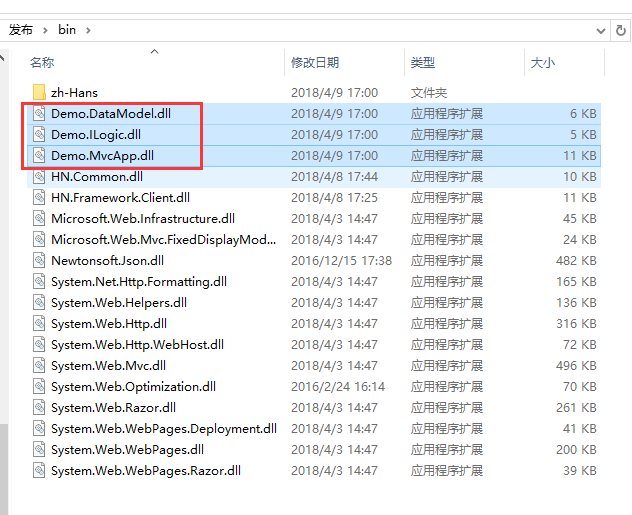
1. 配置IIS网站

## 5.2 前台发布（框架已发布）

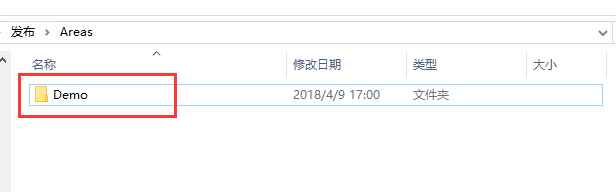
1. 在MVC项目上点右键》发布，使用文件发布



1. 将bin文件夹下对应的模型、接口、MVC应用程序的dll拷贝至IIS服务器上对应目录下



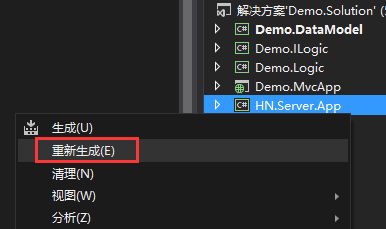
1. 将Areas目录下的内容复制到IIS服务器下对应目录下



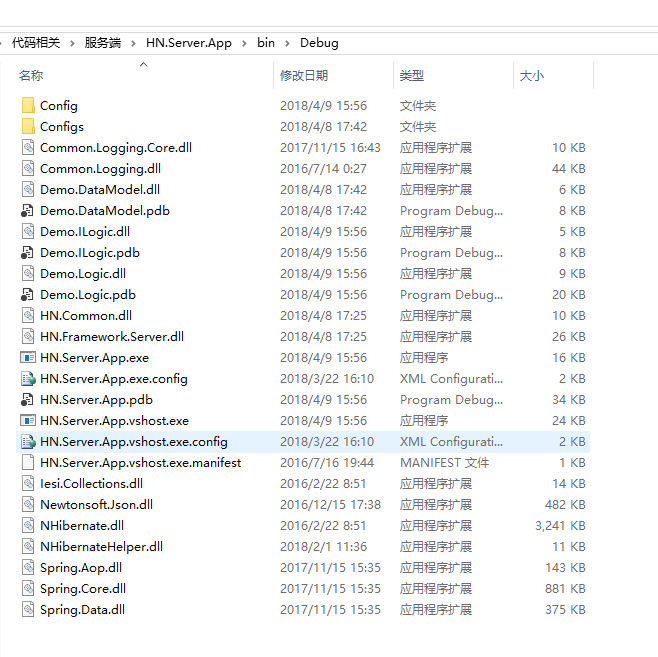
1. 重启IIS网站

## 5.3 后台发布（框架未发布）

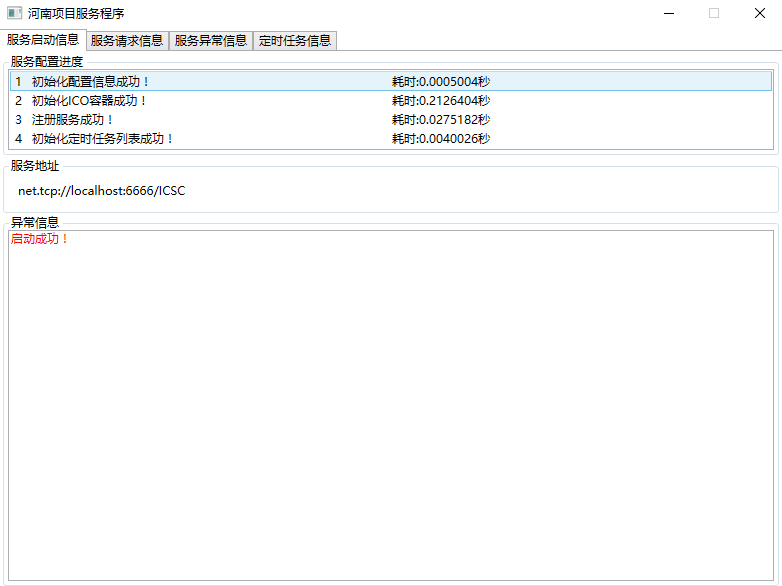
1. 在服务的项目上点右键》重新生成



1. 将生成的Debug下的文件，复制到业务处理服务器上

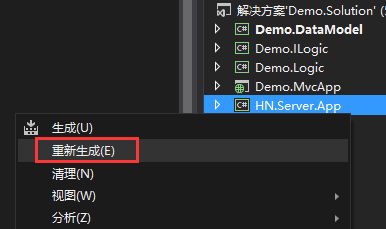


1. 运行HN.Server.App.exe，启动服务

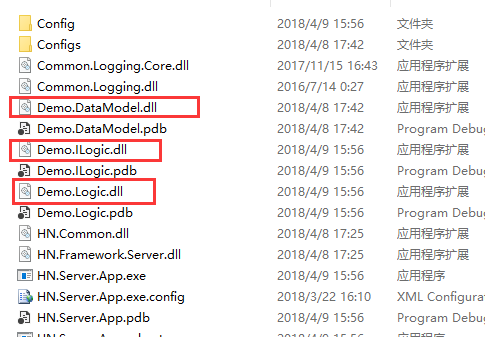


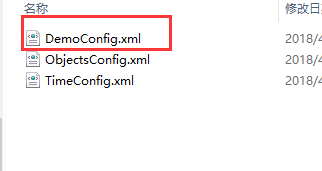
## 5.4 后台发布（框架已发布）

1. 在服务的项目上点右键》重新生成



1. 将生成的Debug下模块所用的dll及配置文件，复制到业务处理服务器上对应文件夹下





【注】若第一次发布本模块

1. 把服务器上的ObjectsConfig.xml中加入对DemoConfig.xml的引用
2. 数据库配置文件NHDB.cfg.xml中加入对数据模型的引用