**ALEXFW框架使用说明**

[1 配置 2](#_Toc3544127)

[1.1 获取框架 2](#_Toc3544128)

[1.2 打开解决方案 2](#_Toc3544129)

[1.3 启用种子数据 3](#_Toc3544130)

[1.4 编译运行，测试前台member登录功能 4](#_Toc3544131)

[2 自动化生产开发环境 5](#_Toc3544132)

[2.1 在领域模型的Entity项目中添加演示实体Category 5](#_Toc3544134)

[2.2 参考注释说明，利用特性定义实体增删查改中的样式 6](#_Toc3544135)

[2.3 打开数据上下文类，添加EF的映射 6](#_Toc3544136)

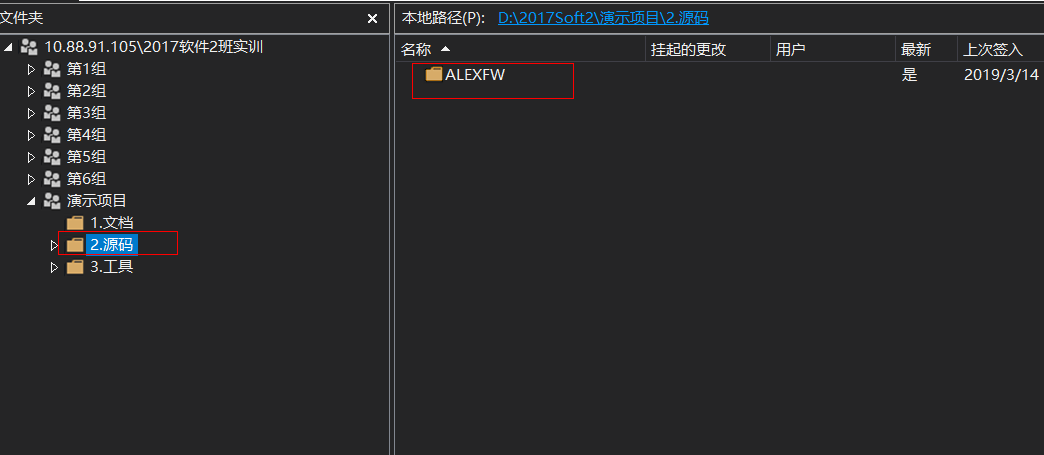
[2.4 自动生成控制器 7](#_Toc3544137)

[2.5 实体的权限控制 9](#_Toc3544138)

[3 例子 11](#_Toc3544139)

# 配置

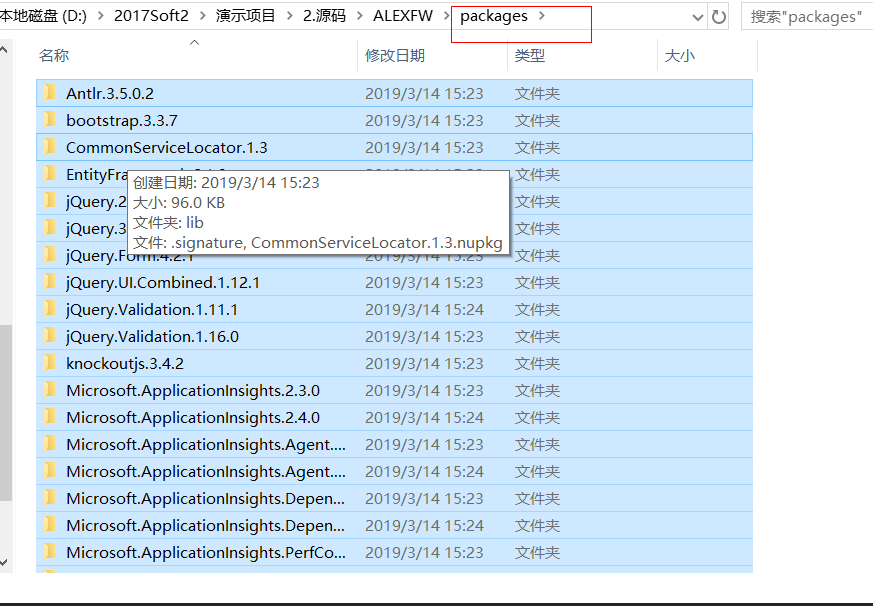
## 获取框架



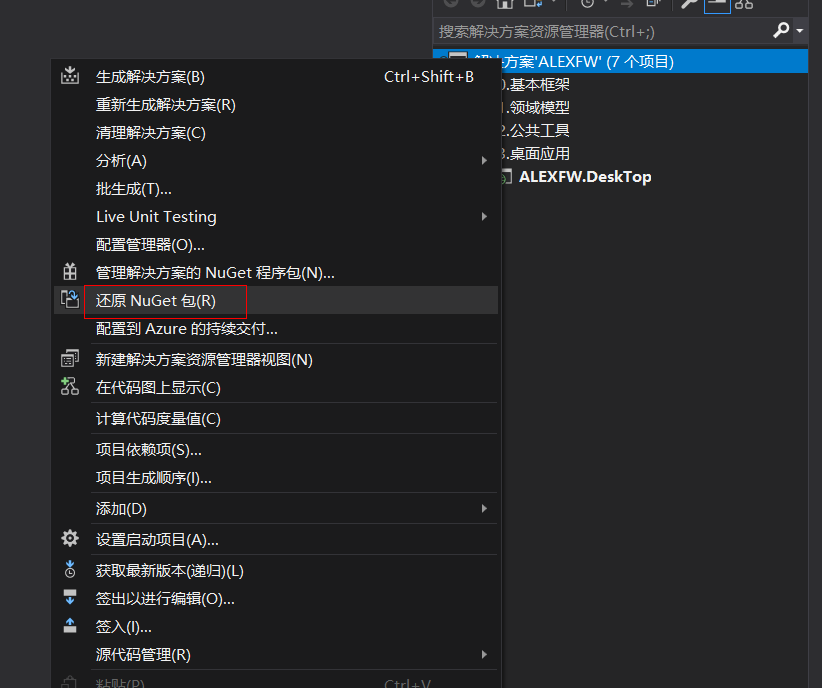
## 打开解决方案



删除Package中的内容

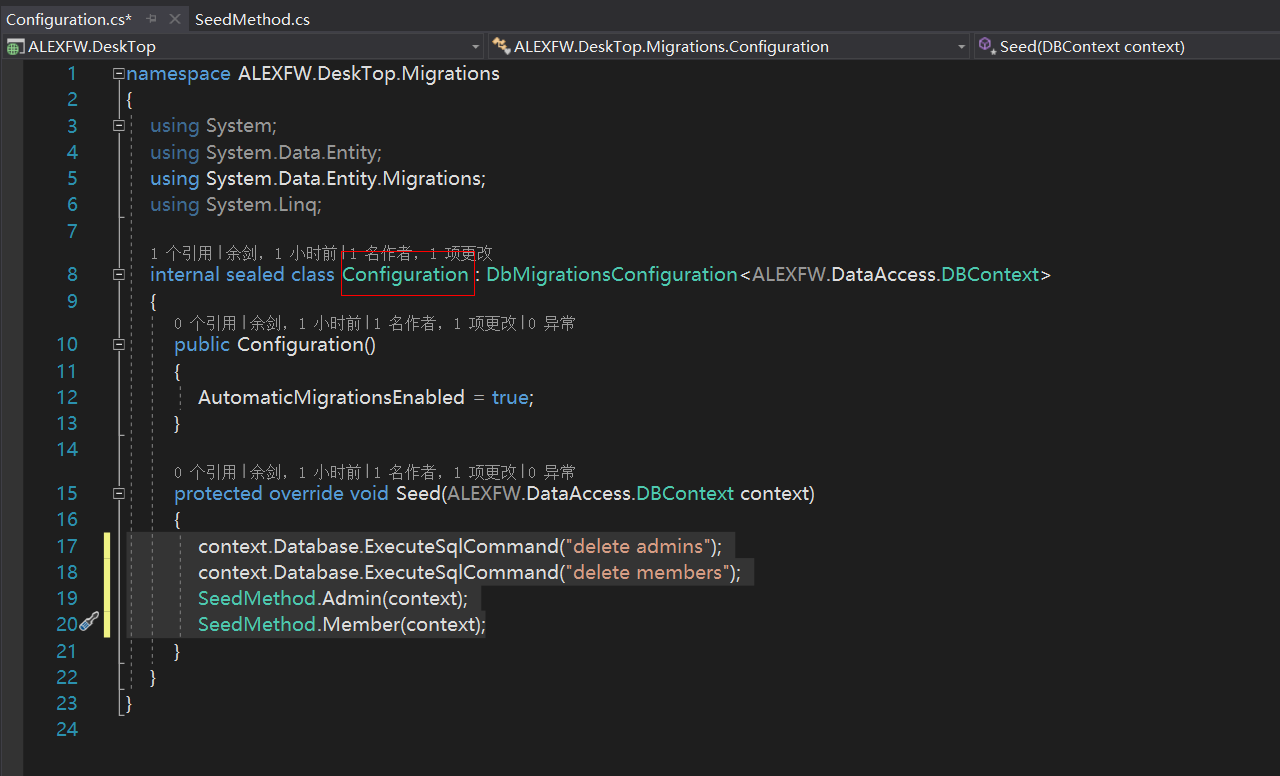


还原nuget包

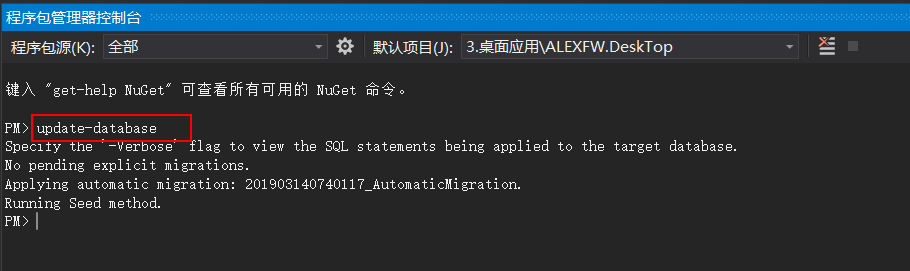


重新生成框架

## 启用种子数据



更新到SQL Server



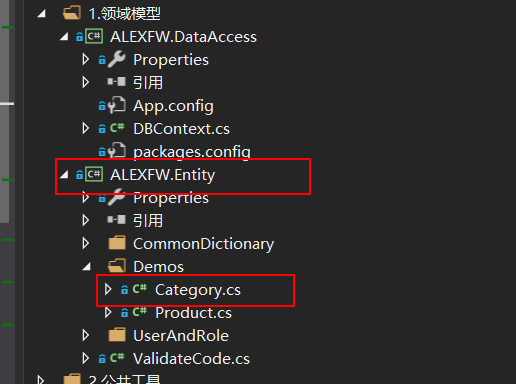
## 编译运行，测试前台member登录功能



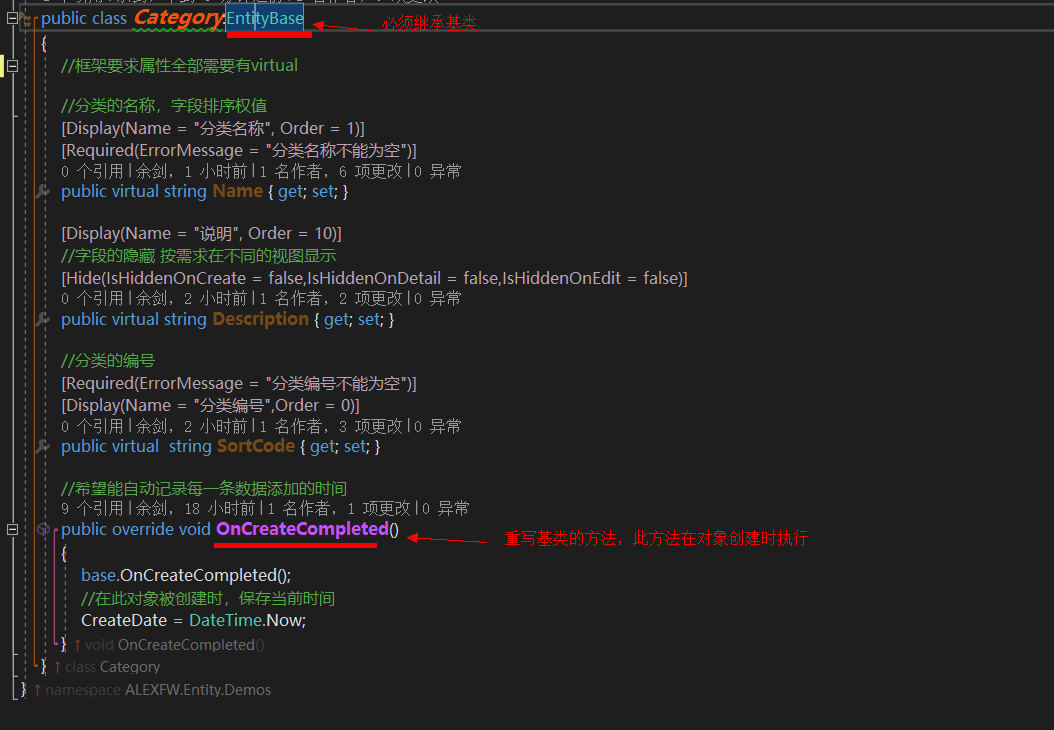
# 自动化生产开发环境



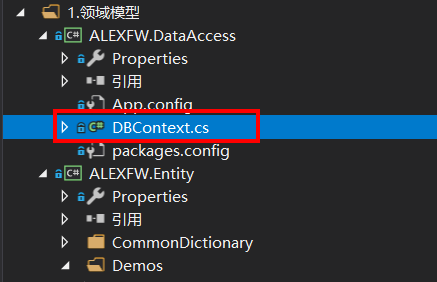
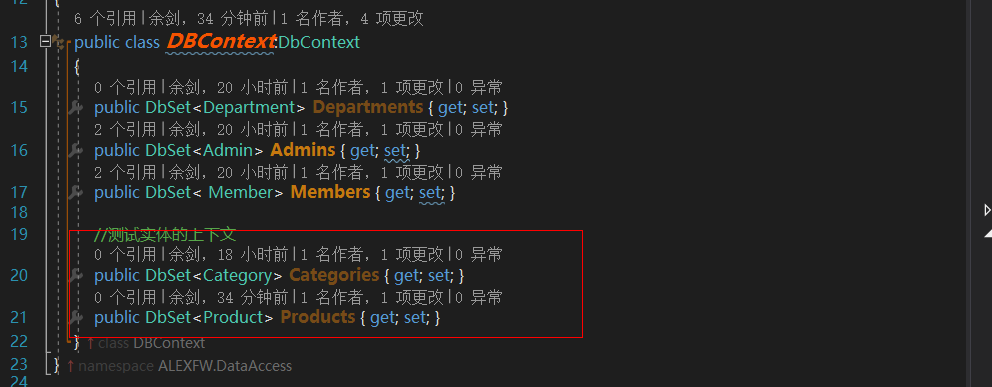
## 在领域模型的Entity项目中添加演示实体Category



## 参考注释说明，利用特性定义实体增删查改中的样式



## 打开数据上下文类，添加EF的映射

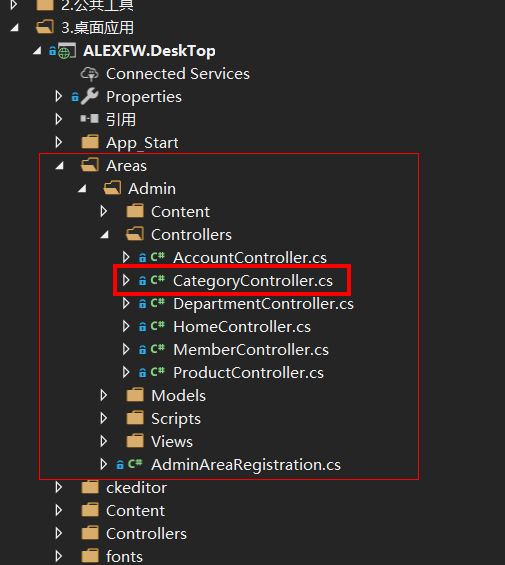
 

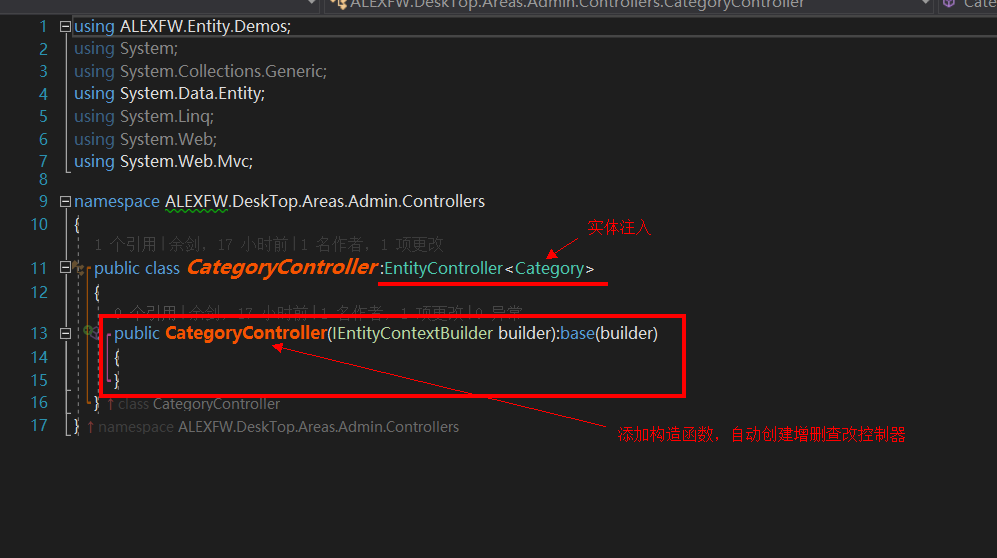
执行数据迁移，映射到SQL Server

## 自动生成控制器

框架为admin管理用户组单独使用管理域/admin做后台管理功能，并且在生产环境下自动生成CRUD控制器。

在对应位置创建与实体名称必须一致的空controller，如实体为Category，则控制器为CategoryController

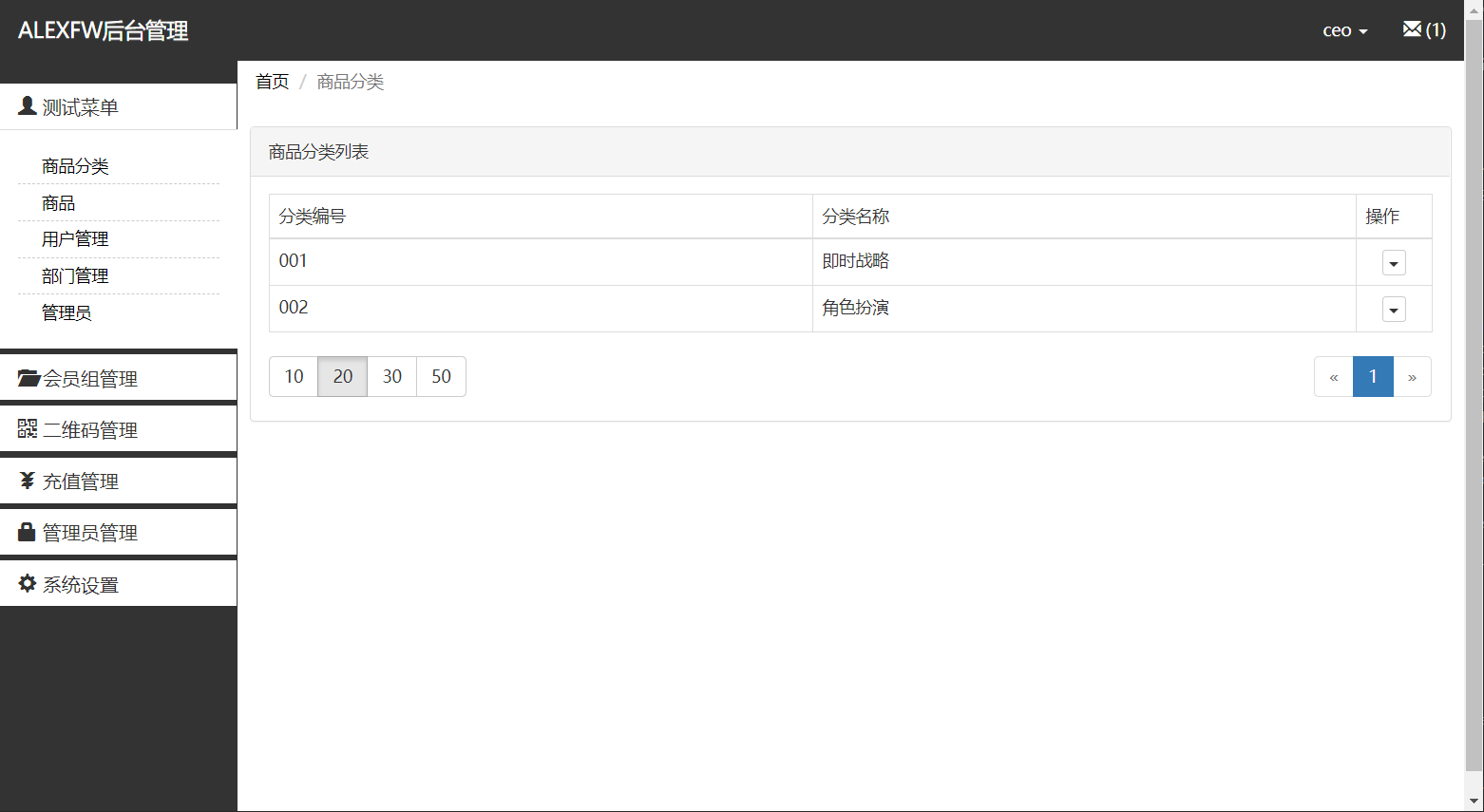




编译运行得到Category的实体维护功能



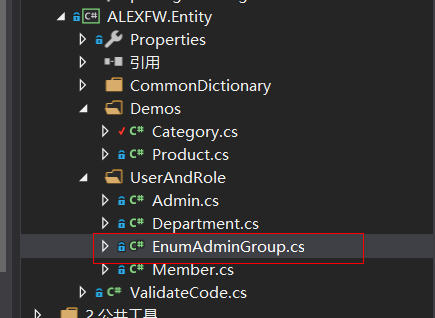
利用默认管理员登录



自动创建访问的控制器地址是http://localhost:XXXX/admin/category

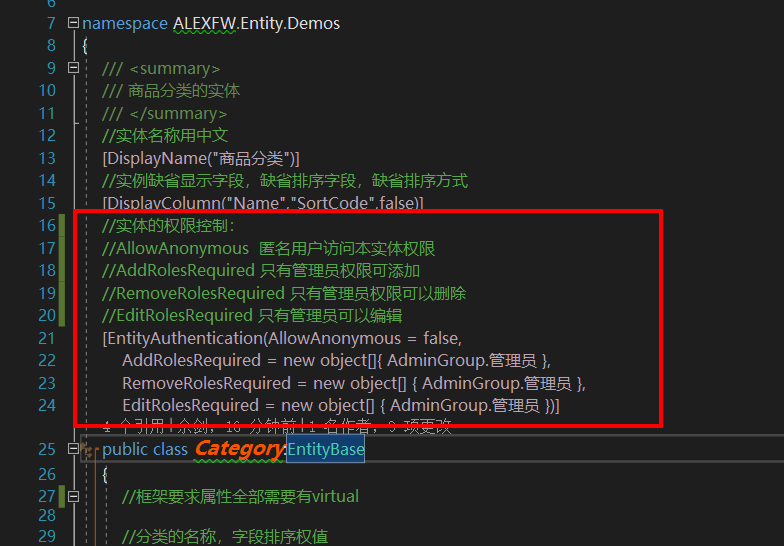
## 实体的权限控制

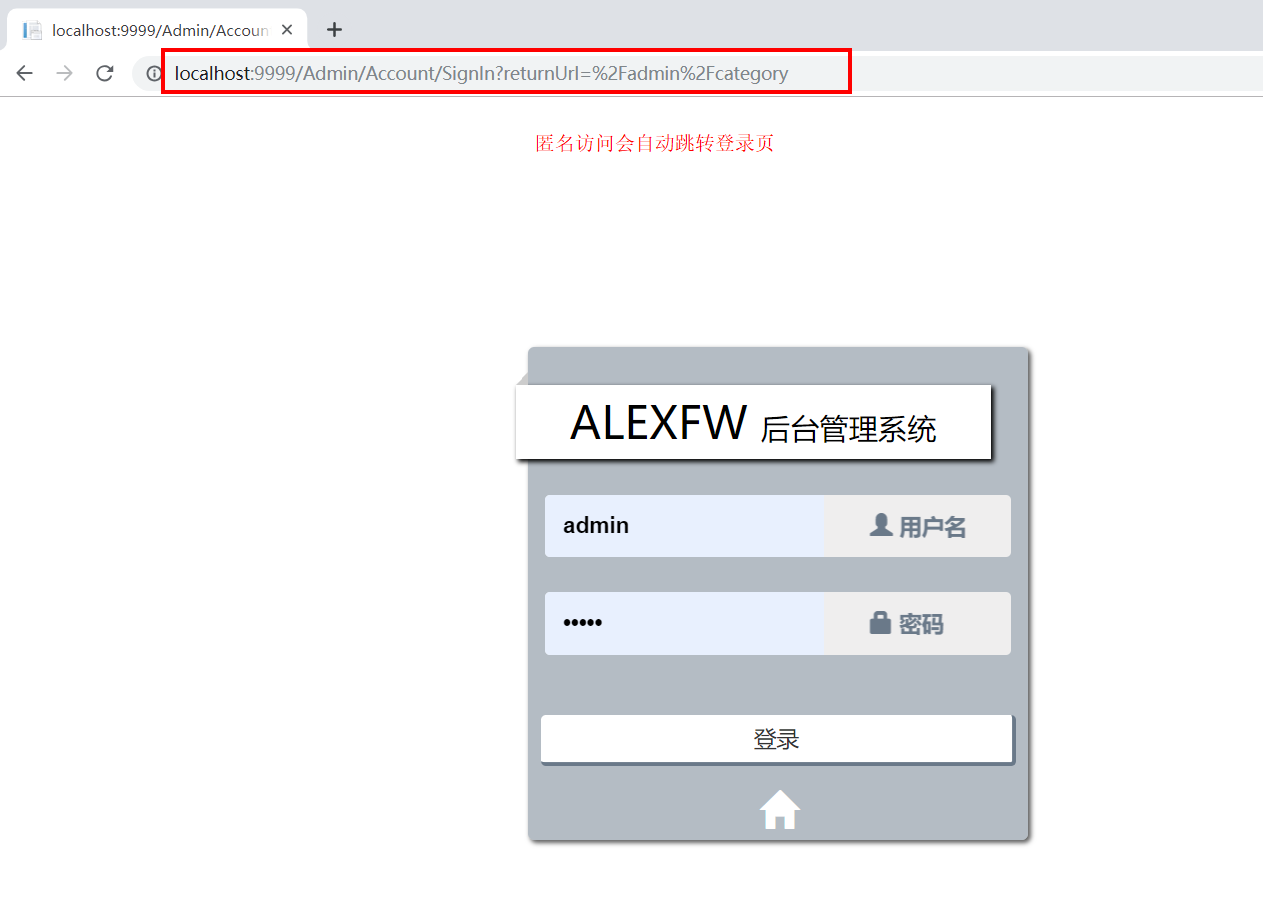
权限值在领域模型中利用Flags特性定义





实体中利用EntityAuthentication特性定义权限





用户不是“管理员”，受限访问



# 例子



在Models文件夹里添加EmployeeGroup员工组类。

EmployeeGroup继承EntityBase。

[复制代码](javascript:void(0);)

[DisplayName("员工组")]

[DisplayColumn("GroupName", "GroupName")]

public class EmployeeGroup : EntityBase

{

[Required]

[Display(Name = "员工组名称", Order = 0)]

public virtual string GroupName { get; set; }

[Hide]

public virtual ICollection<Employee> Employees { get; set; }

}

[复制代码](javascript:void(0);)

在Models文件夹里添加Employee员工组类。

Employee同样继承EntityBase。

[复制代码](javascript:void(0);)

[DisplayName("员工")]

[DisplayColumn("Name", "Name")]

public class Employee : EntityBase

{

[Display(Name = "员工名称", Order = 0)]

[Required]

public virtual string Name { get; set; }

[Display(Name = "性别", Order = 10)]

[CustomDataType(CustomDataType.Sex)]

public virtual bool Sex { get; set; }

[Required]

[Display(Name = "员工工号", Order = 20)]

public virtual string JobNumber { get; set; }

[Column(TypeName = "datetime2")]

[Display(Name = "出生日期", Order = 30)]

public virtual DateTime Birth { get; set; }

[Display(Name = "婚否", Order = 40)]

public virtual bool Marital { get; set; }

[Required]

[Display(Name = "部门", Order = 50)]

public virtual EmployeeGroup Group { get; set; }

[Display(Name = "联系电话", Order = 60)]

public virtual string Tel { get; set; }

[Display(Name = "电子邮件", Order = 70)]

public virtual string Email { get; set; }

[Display(Name = "QQ", Order = 80)]

public virtual string QQ { get; set; }

}

[复制代码](javascript:void(0);)

public class DataContext : DbContext

{

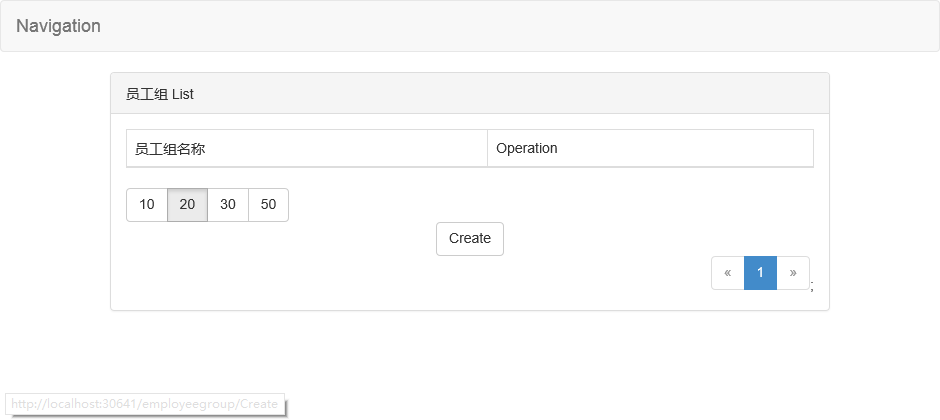
public DbSet<Employee> Employee { get; set; }

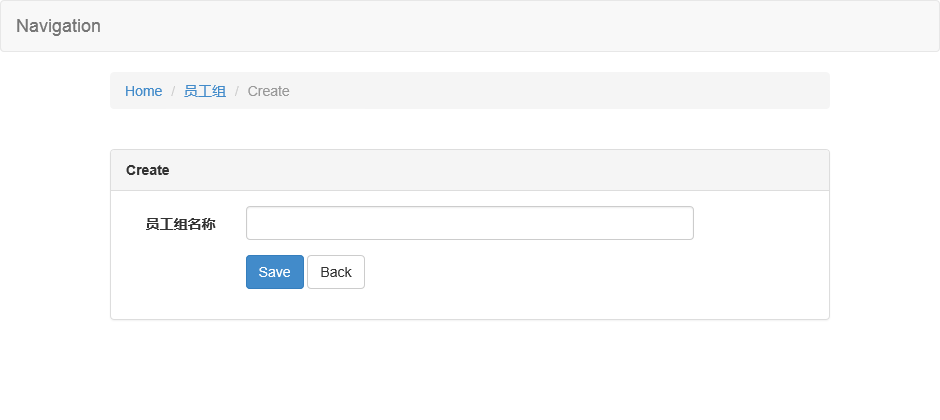
public DbSet<EmployeeGroup> EmployeeGroup { get; set; }

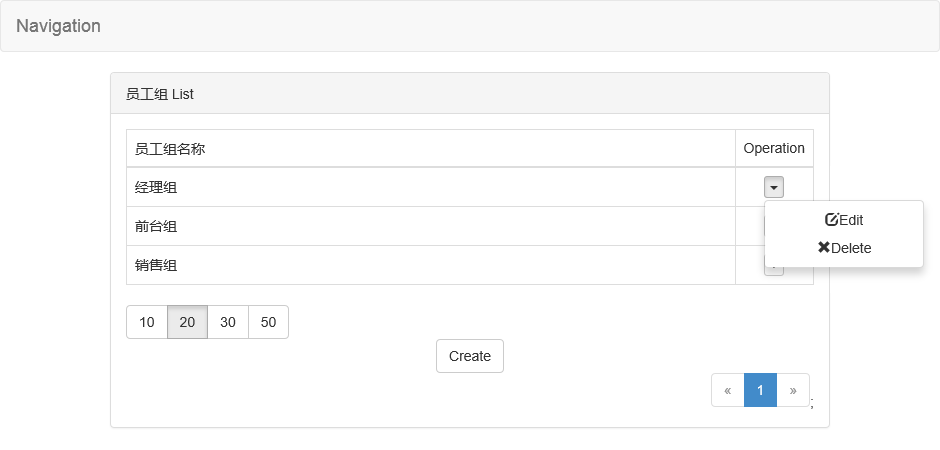
}

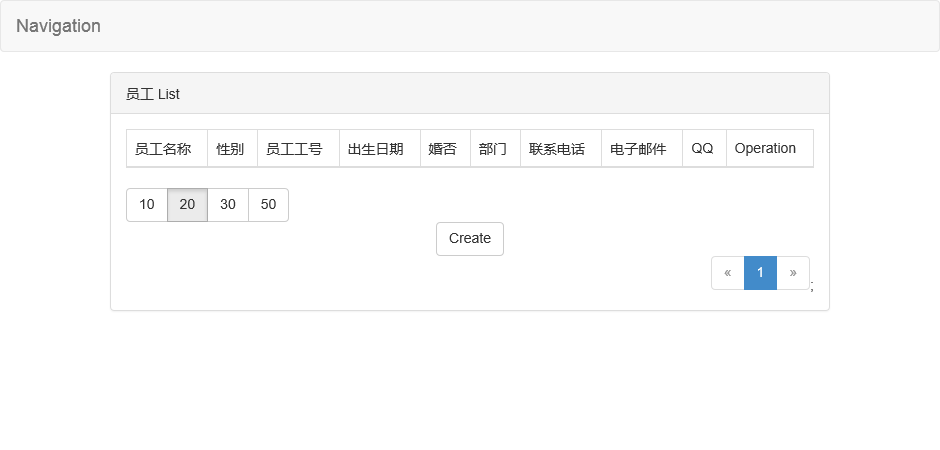
[复制代码](javascript:void(0);)

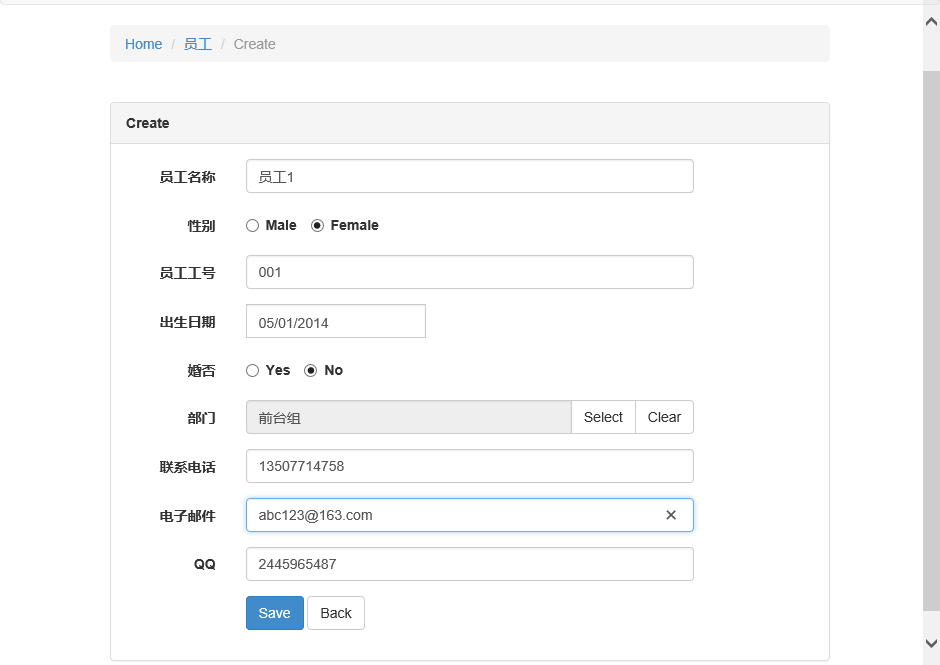
现在，您已经创建好项目了，您可以访问/Employee或/EmployeeGroup地址对实体进行编辑。

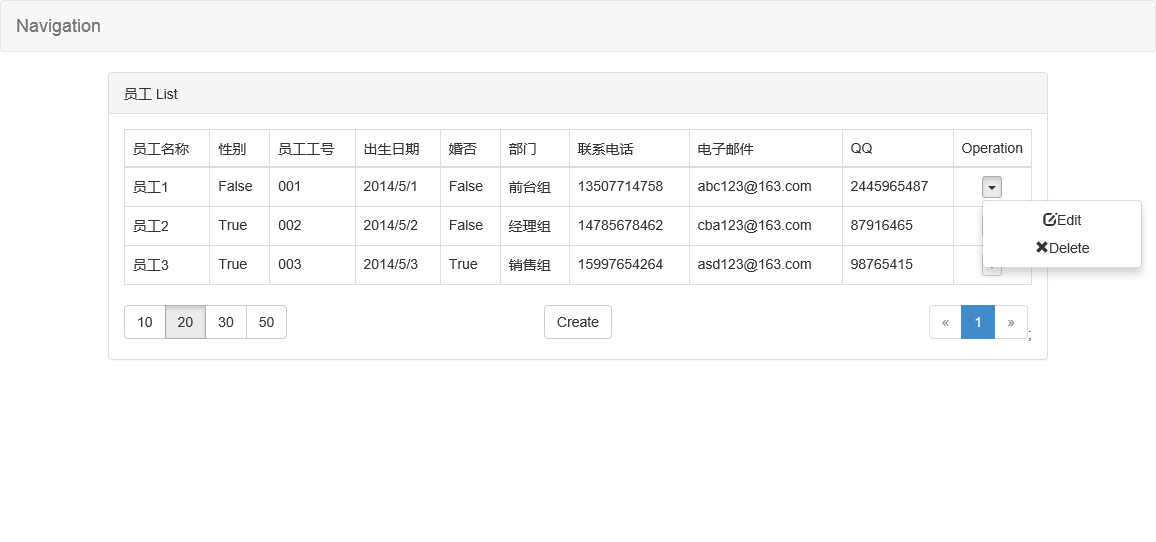












这是个最基础的示例，实际情况您需要对\_Layout.cshtml进行编辑，更有可能创建控制器实现其它功能。