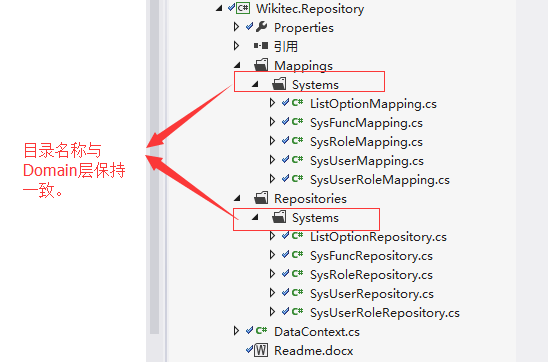
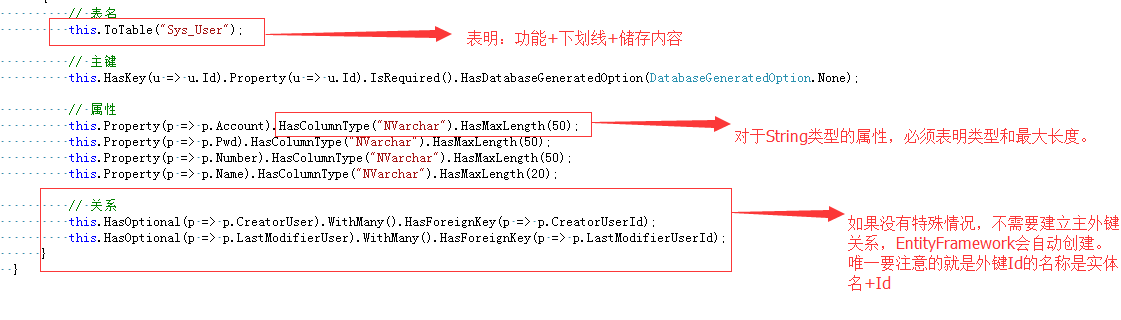
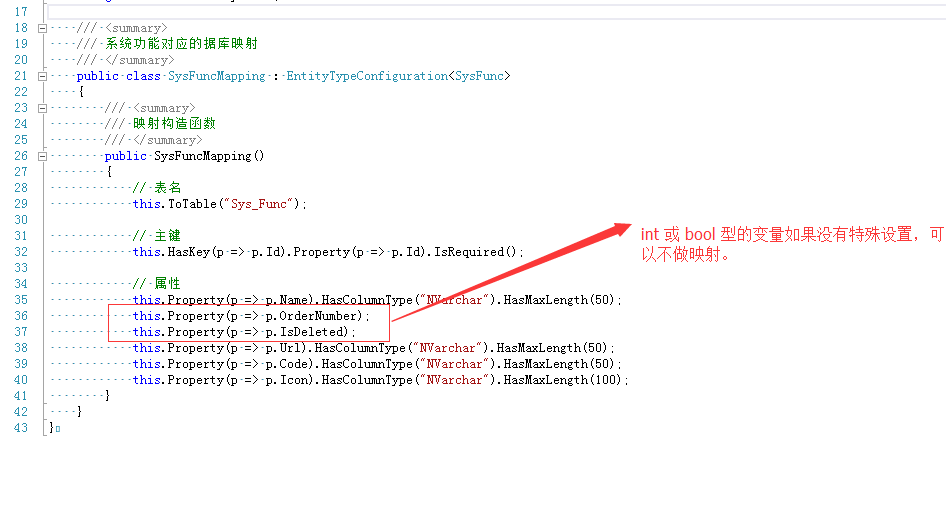
1. Repository层简介：Repository层采用的是基于泛型的仓储模式的实现。用于封装对数据库的查询、更新等操作。通过功能强大的RepositoryBase基类，几乎不需要我们写数据访问的代码，至于事物的控制是放在application层的。
2. Repository目录结构
   1. Mappings：实体和数据库的映射关系。
   2. Repositories：数据库访问器的实现
   3. DataContext.cs 数据库上下文，注册映射对象。
   4. UnitOfWork：事物提交单元



1. Mapping说明
   1. 需要在对象关系中指定表明，规则是：功能+下划线+存储内容
   2. 对于String型的属性，必须确定类型和最大长度
   3. 如果无特殊情况，不需要添加主外键关系。EntityFramework会自动创建。
   4. 对于多对多关系需要手动创建第三张表



* 1. 对于int和bool值的属性，如果无特殊操作则不需要做映射。



1. Repositories说明
   1. 对于多机构项目，目前有两个基类文件，分别是RepositoryBase和RepositoryBaseForOrg。两个基类的区别是ForOrg那个会自动添加当前用户的OrgID作为过滤条件。只能查询出当前机构的数据信息。
   2. 注意引用的命名空间：Wikitec.Infrastructure.Data.IQueryable;
2. 其它
   1. DataContext必须继承自AppDataContext 才会自动保存创建人、创建时间、修改人、修改时间等信息。