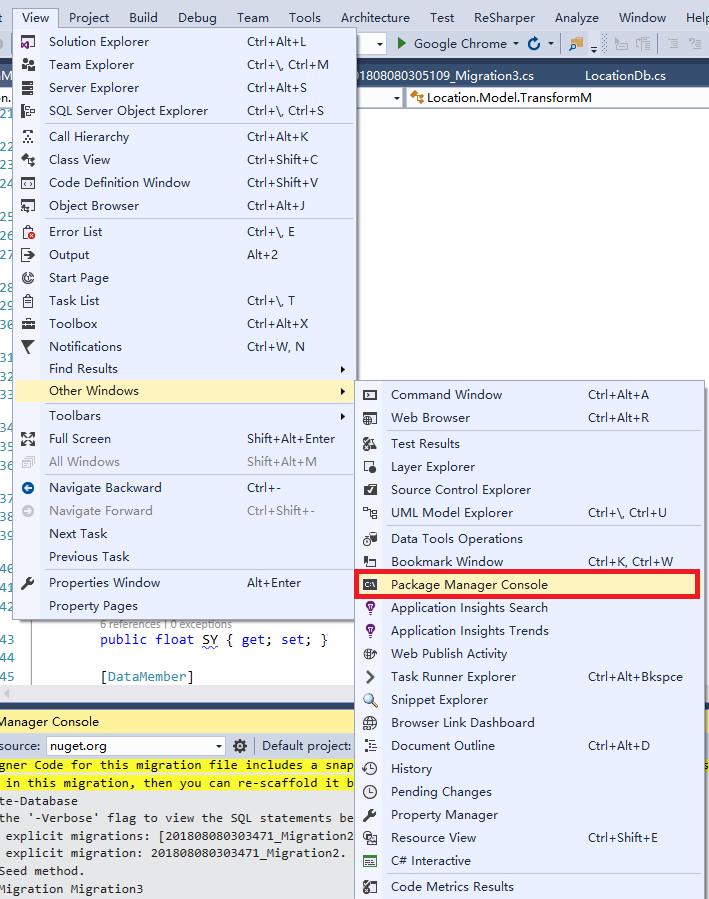
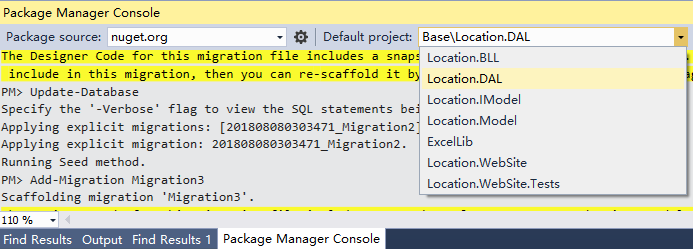
参考：<https://www.cnblogs.com/mstmdev/p/5471269.html>

不过我们自己的项目稍微有所不同

1. 打开程序包管理器，



1. 切换到有DbContext的项目，因为我们的DbContext不在启动项目中。



1. 启动数据迁移

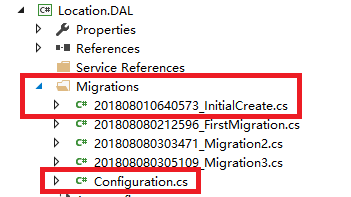
Enable-Migrations -ContextTypeName Location.DAL.LocationDb

因为我们有多个DbContext,要指定一个。

前面不切换项目的话使用指令：

Enable-Migrations -ProjectName Location.DAL -ContextTypeName Location.DAL.LocationDb

这时会创建一个Migrations文件夹，里面会创建一个Configuration.cs文件和...\_InitialCreate.cs文件。



Configuration.cs文件是配置信息，...\_InitialCreate.cs文件是当前数据库结构

1. 创建数据迁移

输入：Add-Migration FirstMigration

FirstMigration可以是其他名称

会在Migrations文件夹中生成...\_FirstMigration.cs文件，文件中是当前类结构和数据库的差异。

1. 更新数据库

输入：Update-Database

数据库根据最后一个数据迁移文件修改数据库结构，不会丢失数据。

1. 设置为自动数据迁移
2. 修改Configuration.cs文件

public Configuration()

{

//AutomaticMigrationsEnabled = false;

AutomaticMigrationsEnabled = true;

AutomaticMigrationDataLossAllowed = true;

ContextKey = "Location.DAL.LocationDb";

}

1. 修改DbContext

public LocationDb() : base("LocationConnection")

{

Database.SetInitializer<LocationDb>(new MigrateDatabaseToLatestVersion<LocationDb, Configuration>());//自动数据迁移

}