

EF Core 反向工程自訂義T4 Template Model 轉換Enum

Created @2024年6月2日 上午1:15

Tags

1. 我先在保哥的範本資料庫Course新增欄位CourseType

資料行名稱	資料類型	允許 Null
CourseID	int	<input type="checkbox"/>
Title	nvarchar(50)	<input checked="" type="checkbox"/>
Credits	int	<input type="checkbox"/>
DepartmentID	int	<input type="checkbox"/>
CourseType	int	<input type="checkbox"/>
		<input type="checkbox"/>

資料變更為

	CourseID	Title	Credits	DepartmentID	CourseType
1	1	Entity Framework 6 開發實戰	1	5	1
2	2	Git新手入門	1	5	2
3	3	Git進階版控流程	2	5	2
4	4	ASP.NET MVC 5 開發實戰	5	1	1
5	5	Javascript新手入門	1	1	3
6	6	ASP.NET Core 3 開發實戰：從入門到進階	5	1	1
7	7	LINQ 語法入門	3	1	1
8	8	C# 開發實戰：非同步程式開發技巧	3	5	1
9	9	.NET / C# 開發實戰：掌握相依性注入的觀念與開發技巧	4	5	1
10	10	Angular 新手開發練功坊	5	5	3
11	12	十項超實用 Excel 技能 — 節省 50% 的工作效率	2	2	4
12	13	Windows Containers 微服務架構實戰	4	13	5

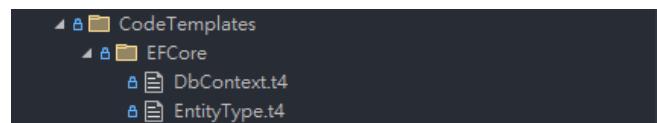
2. 在專案新增Enums 資料夾，並新增CourseType.cs

```
CourseType.cs  GlobalUsing.cs  Course.cs  Entity.cs
Demo.API
1  namespace Demo.API.Enums;
2
3  public enum CourseType
4  {
5      程式開發 = 1,
6      版本控制 = 2,
7      網頁開發 = 3,
8      生產力工具 = 4,
9      系統架構 = 5
10 }
11
```

3. 利用EF Core Power Tools，進行反向代理，並且點選使用模板自訂程式碼



3. 會產生以下資料夾與檔案，附檔名為T4



4. 安裝延伸模組 T4，這樣T4 語法Code才會變色，不然很難Debug



裝完會變成這樣

```

<#@ assembly name="Microsoft.EntityFrameworkCore" #>
<#@ assembly name="Microsoft.EntityFrameworkCore.Design" #>
<#@ assembly name="Microsoft.EntityFrameworkCore.Relational" #>
<#@ assembly name="Microsoft.Extensions.DependencyInjection.Abstractions" #>
<#@ parameter name="EntityType" type="Microsoft.EntityFrameworkCore.Metadata.IEntityType" #>
<#@ parameter name="Options" type="Microsoft.EntityFrameworkCore.Scaffolding.ModelCodeGenerationOptions" #>
<#@ parameter name="NamespaceHint" type="System.String" #>
<#@ import namespace="System.Collections.Generic" #>
<#@ import namespace="System.ComponentModel.DataAnnotations" #>
<#@ import namespace="System.Linq" #>
<#@ import namespace="System.Text" #>
<#@ import namespace="Microsoft.EntityFrameworkCore" #>
<#@ import namespace="Microsoft.EntityFrameworkCore.Design" #>
<#@ import namespace="Microsoft.Extensions.DependencyInjection" #>

// Template version: 800 - please do NOT remove this line
if (EntityType.IsSimpleManyToManyJoinEntityType())
{
    // Don't scaffold these
    return "";
}

var renames = new List<(string EntityTypeName, string PropertyName, string NewPropertyName)>
{
    ("PlatformIdentifierMapping", "PlatformPlatformType", "DestinationPlatform"),
    ("PlatformIdentifierMapping", "PlatformPlatformTypeNavigation", "SourcePlatform"),
};

```

5. 在作者GitHub的 Issue 有新增新的T4 模板 <https://github.com/ErikEJ/EFCorePowerTools/issues/1499>

I have uploaded the start of a "Power" template, with the following features:

- Enum support
- Property renaming
- File header

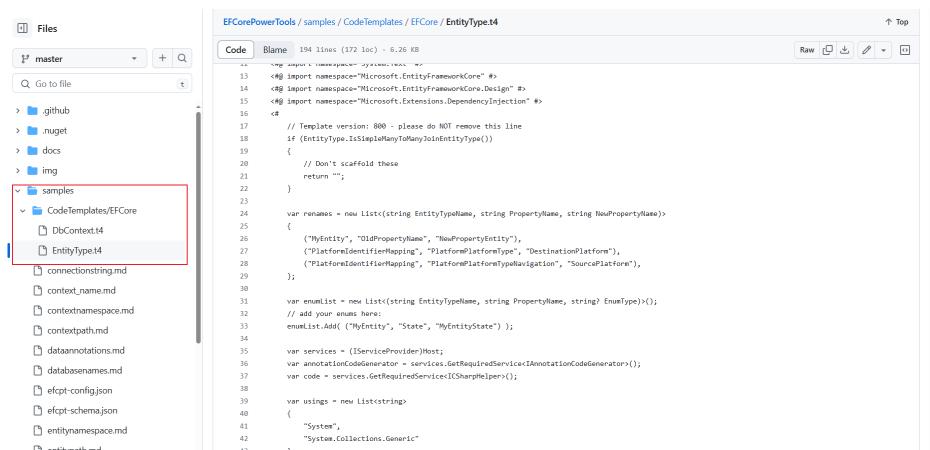
<https://github.com/ErikEJ/EFCorePowerTools/tree/master/samples/CodeTemplates/EFCore>

在這個連結

<https://github.com/ErikEJ/EFCorePowerTools/blob/master/samples/CodeTemplates/EFCore/EntityType.t4>

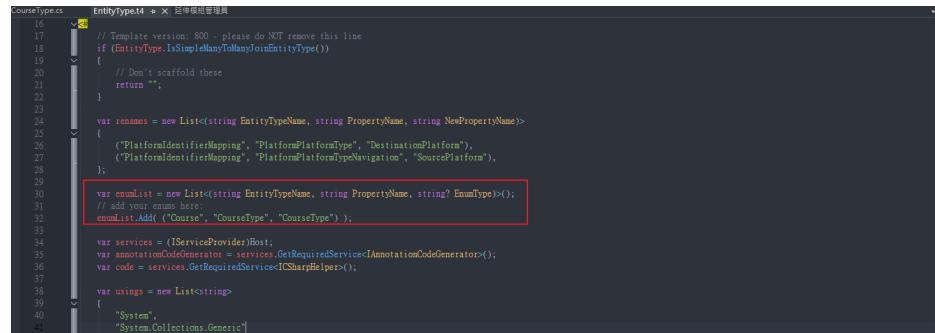
我們直接去複製這兩份檔案 EntityType.t4 跟 DbContext.t4 並覆蓋剛剛產生的 EntityType.t4 跟 DbContext.t4

差異就是作者新增的這三個功能我有請AI解釋一下，<https://chatgpt.com/share/46710a0e-5e10-476b-8e06-6d4bad01f6e5>



6. 我這次需求是型別要轉換EnumType，作者很貼心幫我們註解區塊在哪😊

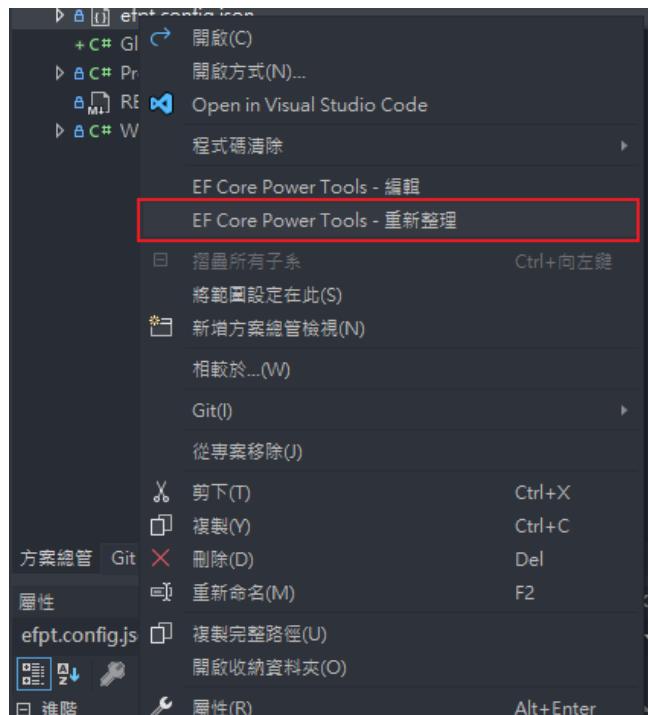
```
// (Entity名稱, Entity 內屬性名稱, Enum 名稱)
enumList.Add( ("Course", "CourseType", "CourseType") );
```



```
EntityType.cs  Entity-Type.t4  Entity 儲存管理員
16 // Template version: 800 - please do NOT remove this line
17 if (EntityType.IsSimpleManyToManyJoinEntityType())
18 {
19     // Don't scaffold these
20     return "";
21 }
22 var renames = new List<(string EntityTypeName, string PropertyName, string NewPropertyName)>
23 {
24     ("PlatformIdentifierMapping", "PlatformPlatformType", "DestinationPlatform"),
25     ("PlatformIdentifierMapping", "PlatformPlatformTypeNavigation", "SourcePlatform"),
26 };
27
28 var enumList = new List<(string EntityTypeName, string PropertyName, string? EnumType)>();
29 // add your enums here.
30 enumList.Add( ("Course", "CourseType", "CourseType") );
31
32 var services = (IServiceProvider)Host;
33 var annotationCodeGenerator = services.GetRequiredService<IAnotationCodeGenerator>();
34 var code = services.GetRequiredService<ICSharpHelper>();
35
36 var usings = new List<string>
37 {
38     "System",
39     "System.Collections.Generic"
40 };
41
```

7. 最後我們在efpt.config.json 右鍵點選 EF Core Power Tools - 重新整理即可，AI一直跟我說儲存 EntityType.t4就可以自動更新😊，答案不是，我後來在YT找到作者有介紹怎麼更新

<https://www.youtube.com/watch?v=PUexkGWErNk>



8. 最後有更新完成

```
1  using System.Collections.Generic;
2
3  namespace Demo.API.Models;
4
5  public partial class Course
6  {
7      // 屬性參考
8      public int CourseID { get; set; }
9
10     // 屬性參考
11     public string Title { get; set; }
12
13     // 屬性參考
14     public int Credits { get; set; }
15
16     // 屬性參考
17     public int DepartmentID { get; set; }
18
19     // 屬性參考
20     public CourseType CourseType { get; set; }
21
22     // 屬性參考
23     public virtual Department Department { get; set; }
24
25     // 屬性參考
26     public virtual ICollection<Enrollment> Enrollment { get; set; } = new List<Enrollment>();
27
28     // 屬性參考
29     public virtual ICollection<Person> Instructor { get; set; } = new List<Person>();
30 }
```

測試一下代碼如何，確實已經轉換成enum

```
[HttpGet("{id}")]
public string Get(int id)
{
    var course = _context.Course.Find(id);
    if (course.CourseType == CourseType.程式開發)
        return "value";
}
```

並確認一下API有無問題

Status: 200 OK Time: 2181.12 ms Size: 1869 bytes

Formatted Raw Headers Request

Body

application/json; charset=utf-8, 1869 bytes

```
[  
 {  
     "courseID": 1,  
     "title": "Entity Framework 6 開發實戰",  
     "credits": 1,  
     "departmentID": 5,  
     "courseType": 1,  
     "department": null,  
     "enrollment": [],  
     "instructor": []  
 },  
 {  
     "courseID": 2,  
     "title": "Git新手入門",  
     "credits": 1,  
     "departmentID": 5,  
     "courseType": 2,  
     "department": null,  
     "enrollment": [],  
     "instructor": []  
 }]
```

貼心跟大家說一下，可以加命名空間儲存在GlobalUsing檔案中，這樣就不用每個檔還要using



The screenshot shows a code editor window with the following details:

- File tab: GlobalUsing.cs
- Code content:

```
global using Demo.API.Enums;
```
- Other tabs: Course.cs, EntityType.t4, 延伸模組管理員
- Project Explorer: Demo.API