

第91期-深入WebAPI复杂模型绑定

2020年3月19日 21:30

复杂类型绑定

需要具备公共默认构造函数和公共可写属性。

模型绑定会查找名称模式 `prefix.property_name` 的源。如果未找到，它将仅查找不含前缀的 `properties_name`。

```
public class Instructor
{
    public int ID { get; set; }
    public string LastName { get; set; }
    public string FirstName { get; set; }
}
```

前缀=参数名称

```
public IActionResult OnPost(int? id, Instructor instructorToUpdate)
```

模型绑定从查找键 `instructorToUpdate.ID` 的源开始操作。如果未找到，它将查找不含前缀的 ID。

前缀=属性名称

```
[BindProperty]
public Instructor Instructor { get; set; }
```

模型绑定从查找键 `Instructor.ID` 的源开始操作。如果未找到，它将查找不含前缀的 ID。

自定义前缀

```
public IActionResult OnPost([Bind(Prefix = "Instructor")] Instructor
instructorToUpdate)
```

模型绑定从查找键 `Instructor.ID` 的源开始操作。如果未找到，它将查找不含前缀的 ID。

复杂类型目标的属性

如果源是**表单数据**，以下属性会影响模型绑定。它们不会影响处理发布的 JSON 和 XML 请求

正文的输入格式化程序。

[BindRequired] 只能应用于模型属性，不能应用于方法参数。 如果无法对模型属性进行绑定，则会导致模型绑定添加模型状态错误。

[BindNever] 只能应用于模型属性，不能应用于方法参数。 防止模型绑定设置模型的属性。

[Bind]可应用于类或方法参数，指定模型绑定中应包含的模型属性。

```
[Bind("LastName,FirstMidName,HireDate")]  
public class Instructor
```

集合类型绑定

对于简单类型集合的目标，模型绑定将查找 parameter_name 或 property_name 的匹配项。如果找不到匹配项，它将查找某种不含前缀的受支持的格式。

假设要绑定的参数是名为 `selectedCourses` 的数组

```
public IActionResult OnPost(int? id, int[] selectedCourses)
```

表单或查询字符串可以传递如下格式

`selectedCourses=1050&selectedCourses=2000`

`selectedCourses[0]=1050&selectedCourses[1]=2000`

`[0]=1050&[1]=2000`

`selectedCourses[a]=1050&selectedCourses[b]=2000
&selectedCourses.index=a&selectedCourses.index=b`

`[a]=1050&[b]=2000&index=a&index=b`

表单可以传递如下格式数据

`selectedCourses[]=1050&selectedCourses[]=2000`

使用下标数字的数据格式 (... [0] ... [1] ...) 必须确保从零开始按顺序进行编号。如果下标编号中存在任何间隔，则间隔后的所有项都将被忽略。

字典模型绑定

对于 Dictionary 目标，模型绑定会查找 parameter_name 或 property_name 的匹配项。如果找不

到匹配项，它将查找某种不含前缀的受支持的格式。

```
public IActionResult OnPost(int? id, Dictionary<int, string> selectedCourses)
```

```
selectedCourses[1050]=Chemistry&selectedCourses[2000]=Economics
```

```
[1050]=Chemistry&selectedCourses[2000]=Economics
```

```
selectedCourses[0].Key=1050&selectedCourses[0].Value=Chemistry&  
selectedCourses[1].Key=2000&selectedCourses[1].Value=Economics
```

特殊数据类型模型绑定

IFormFile 和 IFormFileCollection、CancellationToken、IFormCollection。

使用 [Consumes] 属性定义支持的请求内容类型

```
[HttpPost]  
[Consumes("application/xml")]  
public IActionResult CreateProduct(Product product)
```

路由到此操作的请求必须指定 application/xml 的 Content-Type 头。如果请求未指定 application/xml 的 Content-Type 头，会生成 415 不支持的媒体类型响应。

使用输入格式化程序自定义模型绑定

使用 System.Text.Json 时，若要自定义模型绑定过程，派生 JsonConverter<T> 类。

从模型绑定中排除指定类型

```
services.AddRazorPages()  
    .AddMvcOptions(options =>  
{  
    options.ModelMetadataDetailsProviders.Add(  
        new ExcludeBindingMetadataProvider(typeof(System.Version)));  
    options.ModelMetadataDetailsProviders.Add(  
        new SuppressChildValidationMetadataProvider(typeof(System.Guid)));  
});
```

手动模型绑定

```
[HttpPost]
```

```
public async Task<ActionResult<Pet>> Post([FromForm] Pet pet)
{
    Pet newPet = new Pet();

    await TryUpdateModelAsync(newPet);

    TryValidateModel(newPet);

    return CreatedAtAction(nameof(Get), pet);
}
```