**後端接收前端回傳表單的幾種方式(Model Binding)**

**前言:**

**在專案製作時，幾乎每個專案一定會遇到的一個課題就是處理表單，而這一次我們要探討的就是在ASP.NET MVC中，在後端要接收前端傳過來的資料幾種方式。方法很多，但哪種比較好沒有一定，我們能做的就是依照自己當下情況，選擇最適合的處理方式。**

**透過Request物件接收資料:**

URL:

http://localhost:52849/Home/Detail?id=5

Controller:

**public** ActionResult Detail()

{

ViewBag.Message = "Your contact page.";

**if** (Request.QueryString["id"] != **null**)

{

ViewBag.id = **int**.Parse(Request.QueryString["id"]);

}

**return** View();

}

**QueryString指的就是URL後的?Name=Value…，適合用在單一參數，若遇到多欄位時，較容易出錯。**

**透過RouteData接收資料:**

URL:

http://localhost:52849/Home/Detail/5

Controller:

**public** ActionResult Detail(**int**? id)

{

ViewBag.Message = "Your contact page.";

ViewBag.id = id;

**return** View();

}

**Model Binding基礎應用原理:**

User填入input且送出表單後，Controller透過設定參數名字，取得和input name對應的值。

View:

<div>

@**using** (Html.BeginForm())

{

<p>

Name:<input type="text" name="name">

<input type="submit" value="提交">

</p>

}

</div>

Controller:

[HttpPost]

**public** ActionResult Create(**string** Name)

{

**return** View();

}

Create post的Name會接收到View頁面name為name的input，經過測試繫節過程不區分大小寫(但還是盡量一致維持好習慣)。

**透過FormCollection:**

View:

<div>

@**using** (Html.BeginForm())

{

<p>

id:<input type="text" name="id">

</p>

<p>

Name:<input type="text" name="name">

</p>

<p>

age:<input type="text" name="age">

</p>

<input type="submit" value="提交">

}

</div>

Controller:

[HttpPost]

**public** ActionResult Create(FormCollection Form)

{

ViewBag.id = Form["id"];

ViewBag.name = Form["name"];

ViewBag.age = Form["age"];

**return** View();

}

可惜示弱型別，要一次傳送多欄位值，還有更好的方法。

**Model Binding透過Model:**

View:

<div>

@**using** (Html.BeginForm())

{

<p>

id:<input type="text" name="id">

</p>

<p>

Name:<input type="text" name="name">

</p>

<p>

age:<input type="text" name="age">

</p>

<input type="submit" value="提交">

}

</div>

Controller:

**public** ActionResult Create(Person person)

{

ViewBag.id = person.id;

ViewBag.name = person.name;

ViewBag.age = person.age;

**return** View();

}

直接以Model Person來接收資料，input的Name值會對應Model的名稱，這也是我們最常使用的Model Binding情況。

**多Model的Model Binding:**

View:

<div>

@**using** (Html.BeginForm())

{

<p>

夫id:<input type="text" name="man.id">

</p>

<p>

夫Name:<input type="text" name="man.name">

</p>

<p>

夫age:<input type="text" name="woman.age">

</p>

<p>

妻id:<input type="text" name="woman.id">

</p>

<p>

妻Name:<input type="text" name="woman.name">

</p>

<p>

妻age:<input type="text" name="woman.age">

</p>

<input type="submit" value="提交">

}

</div>

Controller:

[HttpPost]

**public** ActionResult Create(Person man, Person woman)

{

ViewBag.manid = man.id;

ViewBag.manname = man.name;

ViewBag.manage = man.age;

ViewBag.womanid = woman.id;

ViewBag.womanname = woman.name;

ViewBag.womanage = woman.age;

**return** View();

}

透過在View頁面的input name，指定這是那個物件可以取得的值。這招沒用過，但感覺以後找機會使用這種寫法，學習了。

**參考資料:**

ASP.NET MVC 5網站開發美學