Truyền dữ liệu từ Controller sang View

Các các truyền dữ liệu

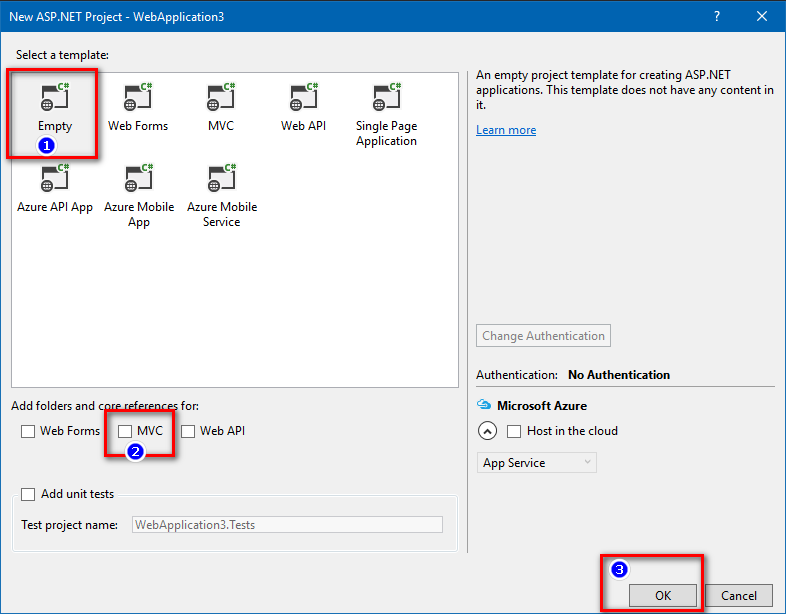
1. ViewBag
2. ViewData
3. TempData
4. Model

# Sử dụng viewBag

Thực hiện việc hiển thị thông tin 1 nhân viên lên trên trình duyệt sử dụng ASP.NET Web Application theo mô hình MCV bằng cách truyền dữ liệu từ Controller sang View sử dụng ViewBag

Hướng dẫn các thực hiện

Bước 1: Tạo một project ASP.NET Web Application. Chọn Template Empty theo mô hình MVC.



Bước 2: Tạo class có tên là Employee trong thư mục Models của Project. Lớp nhân viên sử dụng để lưu trữ thông tin một nhân viên gồm

public class Employee

{

public string EmpID { get; set; }

public string EmpFirstName{get;set;}

public string EmpLastName{ get; set;}

}

}

Bước 3: Tạo Controller có tên là HomeController trong thư mục Controller của Project và tiến hành viết code như sau:

public ActionResult HienThiViewBag()

{

Employee \_employee = new Employee()

{

EmpID = "123",

EmpFirstName = "Nguyễn Minh",

EmpLastName = "Phúc"

};

ViewBag.Employee=\_employee;

return View();

}

Giải thích:

* Trong Action HienThiViewBag tạo 1 đối tượng có tên là employee từ lớp Employee của Models, chứa các giá trị cho các properties của lớp.
* Sử dụng đối tượng ViewBag.Employee để nhận giá trị từ đối tượng \_employee.
* Return View(): Trả về giá trị cho View.

Bước 4: Tạo View tương ứng với Action HienThiViewBag trong Controller Home để hiển thị dữ liệu. và viết code hiển thị như sau:

@{

ViewBag.Title = "HienThiViewBag";

}

<h2>Hiển thị danh sách Employee sử dụng ViewBag</h2>

<p>

<table border="1">

<tr>

<th>ID</th>

<th>FirstName</th>

<th>LastName</th>

</tr>

<tr>

<td>@ViewBag.Employee.EmpID</td>

<td>@ViewBag.Employee.EmpFirstName</td>

<td>@ViewBag.Employee.EmpLastName</td>

</tr>

</table>

</p>

Giải thích:

* Dữ liệu được hiển thị lên Table.
* Khi nhúng code C# vào trong page theo view Engine Razor ta sử dụng @ để thêm vào trước ví dụ: @ViewBag.Employee.EmpID

Mở rộng thêm ví dụ:

Thay vì hiển thị 1 nhân viên ta có thể sử dụng đối tượng ViewBag để truyền bất cứ giá trị gì qua view kể cả danh sách các giá trị.

Ví dụ tiếp theo sẽ truyền 1 danh sách nhân viên sang View và hiển thị vào trong 1 table.

Hướng dẫn: Tương tự các bước trong bài 1. Trong HomeController Thêm 1 Action có tên là HienThiDanhSachNhanVienViewBag

public ActionResult HienThiDanhSachNhanVienViewBag()

{

//Khởi tạo 1 danh sách nhân viên đồng thời gán giá trị vào trong danh sách

List<Employee> list = new List<Employee>(){

new Employee(){EmpID="1",EmpFirstName="Nguyen Van",EmpLastName="Hoàng"},

new Employee(){EmpID="2",EmpFirstName="Trần Van",EmpLastName="Em"},

new Employee(){EmpID="3",EmpFirstName="Lê Văn",EmpLastName="Anh"}

};

ViewBag.List = list;//Đây là lệnh gán 1 danh sách vào ViewBag

return View();//Trả giá trị về cho View

Sau khi tạo trong Action trong HomeController. Tiến hành tạo 1 view từ Action đó.

@{

ViewBag.Title = "HienThiDanhSachNhanVien\_ViewBag";

Layout = "~/Views/Shared/\_Layout.cshtml";

}

<h2>HienThiDanhSachNhanVien\_ViewBag</h2>

<h2>Hiển thị danh sách Employee sử dụng ViewBag</h2>

<p>

<table border="1">

<tr>

<th>ID</th>

<th>FirstName</th>

<th>LastName</th>

</tr>

@foreach (var item in ViewBag.List)

{

<tr>

<td>@item.EmpID</td>

<td>@item.EmpFirstName</td>

<td>@item.EmpLastName</td>

</tr>

}

</table>

</p>

Ghi chú để tiện cho việc mở link mà không phải chỉnh URL trên trình duyệt. Sinh Viên thực hiện view cho Action Index trong HomeControl để mở những view còn lại trong bài ví dụ:

Sử dụng ActionLink để thực hiện theo mẫu

View Index.cshtml

@{

ViewBag.Title = "Index";

}

<h2>Index</h2>

@Html.ActionLink("Ví dụ chuyển dữ liệu ViewBag", "HienThiViewBag", "Home");

<br />

@Html.ActionLink("Ví dụ hiển thị danh sách nhân viên sử dụng ViewBag", "HienThiDanhSachNhanVien\_ViewBag", "Home");

Giải thích:

* Tham số đầu: text hiện thị trên view index.
* Tham số 2: tên Action trong HomeController
* Tham số 3: tên HomeController.

# Sử dụng ViewData

Một cách khác để truyền dữ liệu từ Controller sang View Là sử dụng ViewData.

Cách dùng này tương tự như sử dụng ViewBag. Nhưng khác nhau về cách đặt tên và sử dụng.

Ví dụ về sử dụng ViewData

Bước 1: Tạo model giống ở trên.

Bước 2: Tạo Action trong Controller

public ActionResult HienThiViewData()

{

Employee \_employee = new Employee()

{

EmpID = "123",

EmpFirstName = "Nguyễn Minh",

EmpLastName = "Phúc"

};

ViewData["EmpID"] = \_employee.EmpID;

ViewData["EmpFirstName"] = \_employee.EmpFirstName;

ViewData["EmpLastName"] = \_employee.EmpLastName;

return View();

}

Hoặc truyền 1 danh sách

public ActionResult HienThiDanhSachNhanVienViewData()

{

List<Employee> list = new List<Employee>()

{

new Employee(){EmpID="1",EmpFirstName="Nguyen Van",EmpLastName="Hoàng"},

new Employee(){EmpID="2",EmpFirstName="Trần Van",EmpLastName="Em"},

new Employee(){EmpID="3",EmpFirstName="Lê Văn",EmpLastName="Anh"}

};

ViewData["List"] = list;

return View();

}

Bước 3: Hiển thị dữ liệu lên View

@{

ViewBag.Title = "HienThiViewData";

Layout = "~/Views/Shared/\_Layout.cshtml";

}

<h2>HienThiViewData</h2>

<p>

<table border="1">

<tr>

<th>ID</th>

<th>FirstName</th>

<th>LastName</th>

</tr>

<tr>

<td>@ViewData["EmpID"]</td>

<td>@ViewData["EmpFirstName"]</td>

<td>@ViewData["EmpLastName"]</td>

</tr>

</table>

</p>

Hiển thị danh sách sử dụng vòng lặp foreach

@{

ViewBag.Title = "HienThiDanhSachNhanVienViewData";

Layout = "~/Views/Shared/\_Layout.cshtml";

}

<h2>HienThiDanhSachNhanVienViewData</h2>

<table border="1" style="background-color:@ViewBag.Color">

<tr>

<th>ID</th>

<th>FirstName</th>

<th>LastName</th>

</tr>

//Phải ép kiểm ViewData về kiểu dữ liệu đúng

@foreach (var item in ViewData["List"] as List<CachTruyenDataControllerToView.Models.Employee> )

{

<tr>

<td>@item.EmpID</td>

<td>@item.EmpFirstName</td>

<td>@item.EmpLastName</td>

</tr>

}