Phân biệt ActionResult, ViewResult, PartialViewResult và JsonResult trong ứng dụng .NET MVC Web

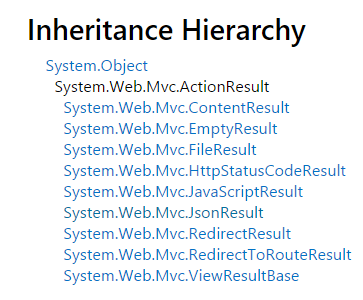
**ON**[**OCTOBER 16, 2015**](https://dominhtuanblog.wordpress.com/2015/10/16/phan-biet-action-view-partialview-json-result-trong-ung-dung-dot-net-mvc-web/)**BY**[**MINHTUANDO**](https://dominhtuanblog.wordpress.com/author/minhtuando/)

Trong quá trình xây dựng ứng dụng .NET MVC, chắc chắn ai trong chúng ta cũng đã từng thao tác với các Action trong Controller, nhưng đã bao giờ bạn tự hỏi khi nào thì ta khai báo kiểu trả về của Action là ActionResult, khi nào thì dùng ViewResult? Bài viết hôm nay ta sẽ cùng nhau tìm hiểu về 4 kiểu trả về thường gặp của một Action trong ứng dụng .NET MVC Web đó là ActionResult, ViewResult, PartialViewResult và JsonResult.

Trước đây khi bắt đầu học những kiến thức đầu tiên về .NET MVC Web, ActionResult luôn luôn là kiểu trả về mặc định của mình, bởi lúc đấy mình chỉ quan tâm làm sao cho code hoạt động như mong muốn mà không nghĩ nhiều đến việc tại sao lại là ActionResult mà không phải những kiểu khác. Sau này mình mới biết ngoài ActionResult còn có rất nhiều kiểu khác mà một Action trong MVC có thể return như: ViewResult, PartialViewResult, JsonResult, FileResult,… Và mình cũng nhận ra việc chỉ định kiểu trả về chính xác cho một Action trong MVC cũng không kém phần quan trọng.

ActionResult

Theo tài liệu của MS, ActionResult là lớp cha của tất cả các lớp Result khác.  Nói cách khác, mọi kiểu trả về có dạng \*Result đều kế thừa từ lớp ActionResult. Nếu bạn truy cập trang docs của ActionResult, bạn sẽ thấy lược đồ thừa kế như sau:



Trong phạm vi bài viết này, ta thấy cả ViewResult lẫn JsonResult đều là con của ActionResult. Do vậy, ActionResult có thể trả về View(), trả về Json(), File(), hay thường gặp nhất là Redirect.

JsonResult

Đúng như tên gọi, JsonResult được dùng để trả về dữ liệu kiểu Json. Một Action có kiểu trả về là JsonResult sẽ trả về dữ liệu có định dạng là một chuỗi Json. Các lời gọi Ajax thường sẽ gọi đến một Action có kiểu JsonResult và xử lý kết quả trả về như thao tác với mảng đối tượng JavaScript.

ViewResult

ViewResult trả về view. Sử dụng ViewResult khi ta cần render một View. Nếu trong phần thân của Action, mọi câu lệnh return đều có dạng return View(…) thì ắt hẳn ViewResult sẽ là lựa chọn trong trường hợp này.

PartialViewResult

ViewResult trả về view. PartialViewResult trả về một phần nhỏ của View. Hay nói cách khác, ta dùng PartialViewResult khi muốn render một phần nhỏ của trang web mà không phải là toàn bộ View. Hãy tưởng tượng trong View của bạn có một phần rất phức tạp cần xử lý riêng biệt, khi đấy ta chỉ cần tạo ra một tệp cshtml View khác, một Action với kiểu PartialViewResult tương ứng, xử lý phần công việc đấy và đối xử với PartialView như một View bình thường, và nhúng nó vào lại trang web thông qua phương thức :

Html.RenderPartial

Vậy khi nào dùng loại nào?

Như đã trình bày ở trên, ta sẽ dùng ViewResult và PartialViewResult khi muốn render một View, JsonResult khi muốn trả về dữ liệu kiểu Json, và ActionResult khi muốn trả về bất cứ kiểu nào trong các kiểu trên. Tóm lại:

* Hãy dùng ViewResult khi bạn chắc chắn Action đấy chỉ trả về View mà không trả về gì khác.
* Hãy dùng JsonResult khi bạn chắc chắn Action đấy chỉ trả về Json mà không trả về gì khác.
* Hãy dùng ActionResult khi Action của bạn có nhiều hơn một lệnh return, mà mỗi lệnh return có thể trả về nhiều kiểu khác nhau mà bạn không chắc chắn Action đấy sẽ trả về kiểu gì sau cùng.

Ví dụ đơn giản, bạn đang cần một màn hình thêm mới một sản phẩm. Điều bạn cần là tạo ra một Action return về một View chứa các form để nhập liệu. Rõ ràng bạn chỉ cần trả về View, bạn không cần phải có Json ở trang nhập liệu, cũng chẳng cần redirect sang chỗ khác. ViewResult sẽ là lựa chọn trong trường hợp này.

Mặt khác, khi người dùng nhập xong thông tin về sản phẩm, họ sẽ submit form lên server. Lúc này có nhiều trường hợp có thể xảy ra. Trường hợp dữ liệu hợp lệ, bạn sẽ phải thêm dữ liệu vào DB, đồng thời chuyển hướng về trang danh sách sản phẩm chẳng hạn. Nhưng rủi dữ liệu không hợp lệ, bạn lại phải trả về trang thêm sản phẩm kèm với một hoặc vài thông báo lỗi. Trường hợp 1 bạn sẽ return Redirect hoặc RedirectToAction, trường hợp 2 bạn sẽ return View. Lúc này bạn không thể chắc chắn rằng mình sẽ trả về kiểu gì. ActionResult sẽ là lựa chọn trong trường hợp này.

Tại sao không áp dụng ActionResult cho tất cả trường hợp?

Ta đã biết ViewResult chỉ trả về View. JsonResult chỉ trả về Json. ActionResult có thể trả về cả View, cả Json hay thậm chí các kiểu Result khác. Vậy tại sao không áp dụng ActionResult cho tất cả các Action trong Controller mà phải phân biệt từng loại?

Theo quan điểm cá nhân mình, có 2 lý do để chúng ta làm việc này. Thứ nhất là để code rõ ràng. Bạn thấy Action A có kiểu trả về là ActionResult, đọc mòn mắt cuối cùng thấy lệnh return nào cũng return View. Bạn nghĩ “chắc zui”. Thứ 2, hãy thử trả lời câu hỏi sau: tất cả kiểu dữ liệu nguyên thủy đều là con của object. Bạn hoàn toàn có thể khai báo như sau:

|  |  |
| --- | --- |
| 1  2  3 | object a = " ";  object b = 1.16M;  object c = 15; |

Vậy tại sao phải nhọc công khai báo string a = ” “, decimal b = 1.16M, hay int c = 15? Đến đây có lẽ bạn đã có câu trả lời. Khi ta chỉ đích danh kiểu của đối tượng mà nó thể hiện, ta tránh được nhiều vấn đề về ép kiểu cũng như khiến code dễ đọc và dễ hiểu hơn.

Bài viết được viết trong phạm vi kiến thức còn hạn chế, mọi lời nguyền rủa xin để lại ở dưới phần comment. Hẹn gặp lại các bạn ở những bài sau.