



[ASP.NET Forums](#) / [General ASP.NET](#) / [MVC](#) / [What is the correct approach use ViewModel and DTO in MVC.](#)

What is the correct approach use ViewModel and DTO in MVC. [Answered]

6 replies

Last post Apr 30, 2015 10:11 AM by [bruce \(sqlwork.com\)](#)

What is the correct approach use ViewModel and DTO in MVC.

Apr 24, 2015 08:06 AM | pathipati

Which is the correct approach using ViewModel and DTO in MVC.

Approach #1:

```
public class PersonDTO
{
    public int Id { get; set; }
    public string FullName { get; set; }
}

public class PersonViewModel
{
    [Required]
    public int Id { get; set; }

    [Required]
    public string FirstName { get; set; }

    [Required]
    public string LastName { get; set; }
}

public ActionResult AddPerson(PersonViewModel personViewModel)
{
    PersonDTO personDTO = new PersonDTO()
    {
        Id = personViewModel.Id,
        FullName = personViewModel.FirstName + " " + personViewModel.LastName
    };
    BusinessLayer.AddPerson(personDTO);
}
```

Approach #2:

```
public class PersonDTO
{
    public int Id { get; set; }
    public string FullName { get; set; }
}

public class PersonViewModel
{
    [Required]
    public int Id { get; set; }

    [Required]
    public string FirstName { get; set; }

    [Required]
    public string LastName { get; set; }

    public PersonDTO Person { get; set; }
}

public ActionResult AddPerson(PersonViewModel personViewModel)
{
    PersonDTO objPersonDTO = new PersonDTO();
    objPersonDTO.FullName = personViewModel.FirstName + " " + personViewModel.LastName;

    personViewModel.Person = objPersonDTO;

    BusinessLayer.AddPerson(personViewModel);
}
```

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 24, 2015 08:15 AM | Mikesdotnetting

The first approach is correct. Apart from the pointless assignment of an object to another type and then back again in the second example, your business layer should know nothing about viewmodels.

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 24, 2015 08:19 AM | pathipati

Thank You Mike.

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 24, 2015 10:44 AM | bruce (sqlwork.com)

they both have a security hole. the id is a simple int passed to the view. any hacker can post with a different id, and effect other peoples records. you should avoid passing actual keys to the view, or on the server side validate that the posting user has access to the key.

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 27, 2015 02:59 AM | pathipati

Bruce, Can you give me the example please!!

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 30, 2015 05:28 AM | Pengzhen Song - MSFT

Hi,

In my opinion, we often don't allow customer to see and set the id directly. To achieve this, we can simply decorate the Id property within the PersonViewModel class with HiddenInput

attribute.

For more information, please refer to the document:

<http://www.codeproject.com/Articles/786607/ASP-Net-MVC-HiddenInput-and-ReadOnly-attributes>
(<http://www.codeproject.com/Articles/786607/ASP-Net-MVC-HiddenInput-and-ReadOnly-attributes>)

Re: What is the correct approach use ViewModel and DTO in MVC.

Apr 30, 2015 10:11 AM | bruce (sqlwork.com)

The hidden approach is not secure unless you also encrypt the value. All web applications are public APIs that can be called without using your forms. Or using a tool like chrome's tamper monkey you can modify the actual page html.