

MVC Multiple Models in One View

Asked 4 years, 11 months ago Active 2 years, 10 months ago Viewed 13k times

I want to reach multiple models in one view. I have DAL folder and DbContext.

10

```
class CvContext : DbContext
{
    public CvContext() : base("CvContext")
    {
    }

    public DbSet<LinkModel> Links { get; set; }
    public DbSet<AboutModel> Abouts { get; set; }
    public DbSet<PortfolioModel> Portfolios { get; set; }
    public DbSet<SkillModel> Skills { get; set; }

    protected override void OnModelCreating(DbModelBuilder modelBuilder)
    {
        modelBuilder.Conventions.Remove<PluralizingTableNameConvention>();
    }
}
```

And HomeController

```
public class HomeController : Controller
{
    private CvContext db = new CvContext();

    public ActionResult Index()
    {
        return View(db.Links.ToList());
    }
}
```

Index.cshtml

```
@model IEnumerable<MvcCv.Models.LinkModel>

<ul>
    @foreach (var item in Model)
```

```

{
    <li>
        <a href="@Html.DisplayFor(modelItem => item.LinkUrl)">
            @Html.DisplayFor(modelItem => item.LinkName)
            <span class="icon"></span>
            <span class="menu-icon">
                
            </span>
        </a>
    </li>
}
</ul>

```

How can i reach all models? I will use foreach for item in Model like Links. Thanks.

c#

asp.net

asp.net-mvc

entity-framework

razor

edited Jun 18 '15 at 22:44



hutchonoid

30.1k ● 14 ● 88 ● 96

asked Jun 18 '15 at 20:22



Berker

536 ● 1 ● 6 ● 23

And also @Html.DisplayFor helper is good choice for this example? – [Berker](#) Jun 18 '15 at 20:23

4 Create a ViewModel that wraps the other models... – [Steve's a D](#) Jun 18 '15 at 20:23

Like CvContext? But inside Model not DAL right? – [Berker](#) Jun 18 '15 at 20:25 ✎

4 Yeah, don't pass your dal around, create a class, that has 4 lists. Populate these lists from your DAL (do this in your controllers action method) and pass this new class to your view.... this is the concept of a view model... it's a model, but it contains ALL and ONLY information that a specific view needs... – [Steve's a D](#) Jun 18 '15 at 20:28 ✎

thanks i will try this – [Berker](#) Jun 18 '15 at 20:47

2 Answers

Active

Oldest

Votes


You should create a view model as follows:

16

```

public class FooViewModel
{
    public IEnumerable<LinkModel> Links { get; set; }
}

```



```
public IEnumerable<AboutModel> Abouts { get; set; }
public IEnumerable<PortfolioModel> Portfolios { get; set; }
public IEnumerable<SkillModel> Skills { get; set; }
}
```



Then from your controller populate them as to your requirements, as an example:

```
public ActionResult Index()
{
    var model = new FooViewModel();
    model.Links = db.Links.ToList();
    model.Abouts = db.Abouts.ToList();
    model.Portfolios = db.Portfolios.ToList();
    model.Skills = db.Skills.ToList();
    return View(model);
}
```

Then change the model in your view to `FooViewModel` and all your properties will be available in there.

```
@model FooViewModel

<ul>
    @foreach (var item in Model.Links)
    {
        <li>
            @item
        </li>
    }
</ul>

<ul>
    @foreach (var item in Model.Links)
    {
        <li>
            @item
        </li>
    }
</ul>

// ....etc, obviously change the outputs as needed.
```

edited Jul 11 '15 at 0:10

answered Jun 18 '15 at 21:10



hutchonoid

30.1k ● 14 ● 88 ● 96

For every model combination we have to create ViewModel according to this solution but would it be good option? What about dynamic model? –
[hasan](#) Jun 5 '17 at 21:28

//suppose you have two Models

```
public class student
{
    public int Id
    public string Name{get;set;}
}
```

```
public class class
{
    public int Id
    public string Name{get;set;}
}
```

// Now combine these two class Model in single Model for example:

```
public class Mixmodel
{
    public Student student {get;set;}
    public Class class {get;set;}
}
```

//here is the Home controller of the Index view

```
@model projectName.MixModel
```

```
@foreach(var item in Model.class)
{
    @html.displayfor(item.class.Name)
}
```

```
@foreach(var item in Model.student)
{
    @html.displayfor(item.student.Name)
}
```

answered Jul 20 '17 at 7:42

 [saurav singh](#)
208 ● 3 ● 12

