

Blog.cs

```
using BestBlogs.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace BestBlogs.Controllers
{
    public class BlogsController : Controller
    {
        // GET: Blogs

        public static BlogDbContext db;

        public ActionResult AddBlog()
        {
            return View();
        }

        [HttpPost]
        public ActionResult AddBlog(Blog blog)
        {
            db = new BlogDbContext();

            db.Blogs.Add(blog);

            if (db.SaveChanges() > 0)
            {
                ViewBag.Message = "Blog Created";

                ModelState.Clear();
            }
            else
            {
                ViewBag.Message = "Blog Not Created";
            }
        }
    }
}
```

```
}
```

```
return View();
```

```
}
```

```
public ActionResult BestBlogs()
```

```
{
```

```
    db = new BlogDbContext();
```

```
    List<Blog> blogs = db.Blogs.Where(u => (u.UpVotes > 100) && u.DownVotes < 25).ToList<Blog>();
```

```
    return View(blogs);
```

```
}
```

```
}
```

```
}
```

{Using system}

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel.DataAnnotations;  
using System.Linq;  
using System.Web;
```

```
namespace BestBlogs.Models
```

```
{  
    public class Blog  
    {  
        [Key]  
        public int BlogId { get; set; }  
        public String BlogContent { get; set; }  
        public String BlogType { get; set; }  
        public String PostedBy { get; set; }  
        public int UpVotes { get; set; }  
        public int DownVotes { get; set; }  
    }  
}
```

Db Connection

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Data.Entity;
```

```
using System.Linq;
```

```
using System.Web;
```

```
namespace BestBlogs.Models
```

```
{
```

```
    public class BlogDbContext:DbContext
```

```
    {
```

```
        public BlogDbContext():base("Blogs_DB")
```

```
        {
```

```
        }
```

```
        public DbSet<Blog> Blogs { get; set; }
```

```
    }
```

```
}
```

//BestBlogs

```
@model IEnumerable<BestBlogs.Models.Blog>
```

```
@{  
    Layout = null;  
}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <meta name="viewport" content="width=device-width" />
```

```
    <title>Best Blogs</title>
```

```
</head>
```

```
<body>
```

```
    <p><h2> Best Blogs</h2></p>
```

```
    <table border="1" id="BestBlogs">
```

```
        <tr>
```

```
            <th>
```

```
                @Html.DisplayNameFor(model => model.BlogContent)
```

```
            </th>
```

```
            <th>
```

```
                @Html.DisplayNameFor(model => model.BlogType)
```

```
            </th>
```

```
            <th>
```

```
                @Html.DisplayNameFor(model => model.PostedBy)
```

```
            </th>
```

```
            <th>
```

```
                @Html.DisplayNameFor(model => model.UpVotes)
```

```
</th>
<th>
    @Html.DisplayNameFor(model => model.DownVotes)
</th>
```

```
</tr>
```

```
@foreach (var item in Model) {
    <tr>
        <td>
            @Html.DisplayFor(modelItem => item.BlogContent)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.BlogType)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.PostedBy)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.UpVotes)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.DownVotes)
        </td>
```

```
</tr>
```

```
}
```

```
</table>
```

```
</body>
```

```
</html>
```

//AddBlog

```
@model BestBlogs.Models.Blog
```

```
@{  
    Layout = null;  
}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <meta name="viewport" content="width=device-width" />
```

```
    <title>Create Blog</title>
```

```
</head>
```

```
<body>
```

```
    <p>
```

```
        <h2>Create Blog</h2>
```

```
    </p>
```

```
    <form method="post">
```

```
        <table>
```

```
            <tr>
```

```
                <td>Blog Content </td>
```

```
                <td>@Html.TextBoxFor(m => m.BlogContent, new { id = "BlogContent" })</td>
```

```
            </tr>
```

```
            <tr>
```

```
                <td>Blog Type </td>
```

```
                <td>@Html.TextBoxFor(m => m.BlogType, new { id = "Type" })</td>
```

```
            </tr>
```

```
        <tr>
```

```

        <td>Posted By </td>

        <td>@Html.TextBoxFor(m => m.PostedBy, new { id = "PostedBy" })</td>
    </tr>

    <tr>

        <td>Up Votes </td>

        <td>@Html.TextBoxFor(m => m.UpVotes, new { id = "UpVotes" })</td>
    </tr>

    <tr>

        <td>Down Votes </td>

        <td>@Html.TextBoxFor(m => m.DownVotes, new { id = "DownVotes" })</td>
    </tr>

    <tr>

        <td></td>

        <td><input type="submit" value="Add" id="Add" /></td>
    </tr>

    <tr>

        <td></td>

        <td><input type="button" value="Best Blog" id="BestBlog"
onclick="window.location='@Url.Action("BestBlogs","Blogs")'" /></td>
    </tr>
</table>

</form>

<span id="blogMessage">
    @if (ViewBag.Message != null)
    {
        @ViewBag.Message
    }
</span>

</body>

</html>

```