Sql server, .net and c# video tutorial

Free C#, .Net and Sql server video tutorial for beginners and intermediate programmers.

Support us   .Net Basics   C# SQL   ASP.NET   ADO.NET   MVC   Slides   C# Programs   Subscribe   Buy DVD   G+

Custom method names in ASP.NET Web API

Suggested Videos
Part 7 - Implementing post method in ASP.NET Web API
Part 8 - Implementing Delete method in ASP.NET Web API
Part 9 - Implementing PUT method in ASP.NET Web API

In this video we will discuss **how to create custom method names in an ASP.NET Web API Controller**

First let's understand the default convention used by ASP.NET Web API to map HTTP verbs GET, PUT, POST and DELETE to methods in a controller.

By default, the HTTP verb GET is mapped to a method in a controller that has the name Get() or starts with the word Get. In the following EmployeesController, the method is named Get() so by convention this is mapped to the HTTP verb GET. Even if you rename it to GetEmployees() or GetSomething() it will still be mapped to the HTTP verb GET as long as the name of the method is prefixed with the word Get. The word Get is case-insensitive. It can be lowercase, uppercase or a mix of both.

```
public class EmployeesController : ApiController
{
    public IEnumerable<Employee> Get()
    {
        using (EmployeeDBEntities entities = new EmployeeDBEntities())
        {
            return entities.Employees.ToList();
        }
    }
}
```

If the method is not named Get or if it does not start with the word get then Web API does not know the method name to which the GET request must be mapped and the request fails with an error message stating The requested resource does not support http method 'GET' with the status code 405 Method Not Allowed. In the following example, we have renamed Get() method to LoadEmployees(). When we issue a GET request the request will fail, because ASP.NET Web API does not know it has to map the GET request to this method.

```
public IEnumerable<Employee> LoadEmployees()
{
    using (EmployeeDBEntities entities = new EmployeeDBEntities())
    {
        return entities.Employees.ToList();
    }
}
```

Angular CRUD

Angular Project Setup | Text | Slides

Reading data | Text | Slides

Routing & Navigation | Text | Slides

Angular base href | Text | Slides

SOLID Principles

SOLID Introduction | Text | Slides

Single Responsibility | Text | Slides

Important Videos

The Gift of Education

Web application for your business

How to become .NET developer

Resources available to help you

Dot Net Video Tutorials

Angular CLI Tutorial

Angular 2 Tutorial

Design Patterns

ASP.NET Web API

Bootstrap

AngularJS Tutorial

jQuery Tutorial

JavaScript with ASP.NET Tutorial

To instruct Web API to map HTTP verb GET to LoadEmployees() method, decorate the method with [HttpGet] attribute.

```
[HttpGet]
public IEnumerable<Employee> LoadEmployees()
{
    using (EmployeeDBEntities entities = new EmployeeDBEntities())
    {
        return entities.Employees.ToList();
    }
}
```

In the following example, we have added another method LoadEmployeeById(int id) to EmployeesController.

```
public class EmployeesController : ApiController
{
    [HttpGet]
    public IEnumerable<Employee> LoadEmployees()
    {
        using (EmployeeDBEntities entities = new EmployeeDBEntities())
        {
            return entities.Employees.ToList();
        }
    }

    public HttpResponseMessage LoadEmployeeById(int id)
    {
        using (EmployeeDBEntities entities = new EmployeeDBEntities())
        {
            var entity = entities.Employees.FirstOrDefault(e => e.ID == id);
            if (entity != null)
            {
                return Request.CreateResponse(HttpStatusCode.OK, entity);
            }
            else
            {
                return Request.CreateErrorResponse(HttpStatusCode.NotFound,
                    "Employee with Id " + id.ToString() + " not found");
            }
        }
    }
}
```

When we navigate to the following URI, notice we are getting all the Employees, instead of just the Employee with Id=1. This is because in this case the GET request is mapped to LoadEmployees() and not LoadEmployeeById(int id). If you want the GET request to be mapped to LoadEmployeeById(int id) when the id parameter is specified in the URI, decorate LoadEmployeeById(int id) method also with [HttpGet] attribute.
http://localhost:35171/api/employees/1

Attributes that are used to map your custom named methods in the controller class to GET, POST, PUT and DELETE http verbs.

Attribute	Maps to http verb
[HttpGet]	GET
[HttpPost]	POST
[HttpPut]	PUT
[HttpDelete]	DELETE

JavaScript Tutorial

Charts Tutorial

LINQ

LINQ to SQL

LINQ to XML

Entity Framework

WCF

ASP.NET Web Services

Dot Net Basics

C#

SQL Server

ADO.NET

ASP.NET

GridView

ASP.NET MVC

Visual Studio Tips and Tricks

Dot Net Interview Questions

Slides

Entity Framework

WCF

ASP.NET Web Services

Dot Net Basics

C#

SQL Server

ADO.NET

ASP.NET

GridView

ASP.NET MVC
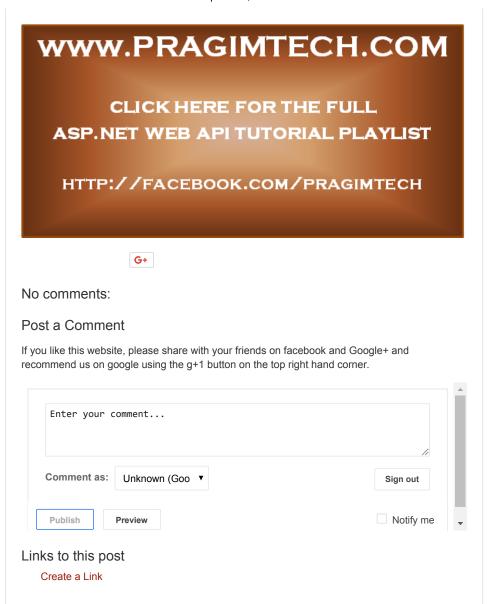
Visual Studio Tips and Tricks

Java Video Tutorials

Part 1 : Video | Text | Slides

Part 2 : Video | Text | Slides

Part 3 : Video | Text | Slides

Interview Questions

C#

SQL Server

Written Test

CBSE Class 9 Maths

Number Systems

CPT Tutorial

Part 1 : Video | Text | Slides

No comments:

Post a Comment

If you like this website, please share with your friends on facebook and Google+ and recommend us on google using the g+1 button on the top right hand corner.

```
Enter your comment...
```

Comment as: Unknown (Goo ▼   **Sign out**

Publish   Preview   ☐ Notify me

Links to this post

Create a Link

Newer Post   Home   Older Post

Subscribe to: Post Comments (Atom)

Powered by Blogger.