

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](#)

Online fake REST API

Suggested Videos

[Part 60 - Angular delete form](#) | [Text](#) | [Slides](#)

[Part 61 - Angular accordion example](#) | [Text](#) | [Slides](#)

[Part 62 - Angular content projection](#) | [Text](#) | [Slides](#)

In this video we will discuss **creating a fake online REST API**.

What is REST API

REST stands for **R**epresentational **S**tate **T**ransfer. REST is an architectural pattern for creating an API that uses HTTP as its underlying communication method.

The REST architectural pattern specifies a set of constraints that a system should adhere to. Some of these constraints are Client Server constraint, Stateless constraint, Cacheable constraint, Uniform Interface constraint etc. We discussed these constraints in [Part 1](#) of [ASP.NET Web API tutorial](#). Let's quickly recap the Uniform Interface constraint.

Uniform Interface - The uniform interface constraint defines the interface between the client and the server. To understand the uniform interface constraint, we need to



PRAGIM
TECHNOLOGIES
Training + Placements

Best software training and placements in marathahalli, bangalore. For further details please call 09945699393.

Complete Tutorials

[JavaScript tutorial](#)

[Bootstrap tutorial](#)

[Angular tutorial for beginners](#)

[Angular 5 Tutorial for beginners](#)

Important Videos

[The Gift of Education](#)

[Web application for your business](#)

understand what a resource is and the HTTP verbs - GET, PUT, POST & DELETE.

In the context of a REST API, a resource typically represents a data entity like Product, Employee, Customer etc. The HTTP verb (GET, PUT, POST, DELETE) that is sent with each request tells the API what to do with the resource. Each resource is identified by a specific URI (Uniform Resource Identifier) or URL (Uniform Resource Locator). The following table shows some typical requests that you see in an API.

Resource	Verb	Outcome
/Employees	GET	Gets list of employees
/Employees/1	GET	Gets employee with Id = 1
/Employees	POST	Creates a new employee
/Employees/1	PUT	Updates employee with Id = 1
/Employees/1	DELETE	Deletes employee with Id = 1

Depending on the server side technology you use, there are many frameworks that we can use to build a REST API. For example, if your server side technology is Microsoft Dot Net, you can use WCF or ASP.NET Web API to create a REST API.

Since this is an Angular course, and to stay focused on it, let's create a fake REST API using JSON Server. In our upcoming videos, we will perform all the CRUD operations using this fake REST API.

The following is the JSON Server Github page

<https://github.com/typicode/json-server>

Execute the following command to install JSON server

`npm install -g json-server`

Execute the following command to start the server

`json-server --watch db.json`

This automatically creates db.json file in the root project folder. Copy and paste the

[How to become .NET developer](#)

[Resources available to help you](#)

Dot Net Video Tutorials

[ASP.NET Core Tutorial](#)

[Angular 6 Tutorial](#)

[Angular CRUD Tutorial](#)

[Angular CLI Tutorial](#)

[Angular 2 Tutorial](#)

[Design Patterns](#)

[SOLID Principles](#)

[ASP.NET Web API](#)

[Bootstrap](#)

[AngularJS Tutorial](#)

[jQuery Tutorial](#)

[JavaScript with ASP.NET Tutorial](#)

[JavaScript Tutorial](#)

[Charts Tutorial](#)

[LINQ](#)

[LINQ to SQL](#)

[LINQ to XML](#)

[Entity Framework](#)

following JSON data in db.json file.

```
{
  "employees": [
    {
      "id": 1,
      "name": "Mark",
      "gender": "Male",
      "contactPreference": "Email",
      "email": "mark@pragimtech.com",
      "dateOfBirth": "1988/10/25",
      "department": "3",
      "isActive": true,
      "photoPath": "assets/images/mark.png"
    },
    {
      "id": 2,
      "name": "Mary",
      "gender": "Female",
      "contactPreference": "Phone",
      "phoneNumber": 2345978640,
      "dateOfBirth": "1979/11/20",
      "department": "2",
      "isActive": true,
      "photoPath": "assets/images/mary.png"
    },
    {
      "id": 3,
      "name": "John",
      "gender": "Male",
      "contactPreference": "Phone",
      "phoneNumber": 5432978640,
      "dateOfBirth": "1976/3/25",
      "department": "3",
      "isActive": false,
      "photoPath": "assets/images/john.png"
    }
  ]
}
```

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

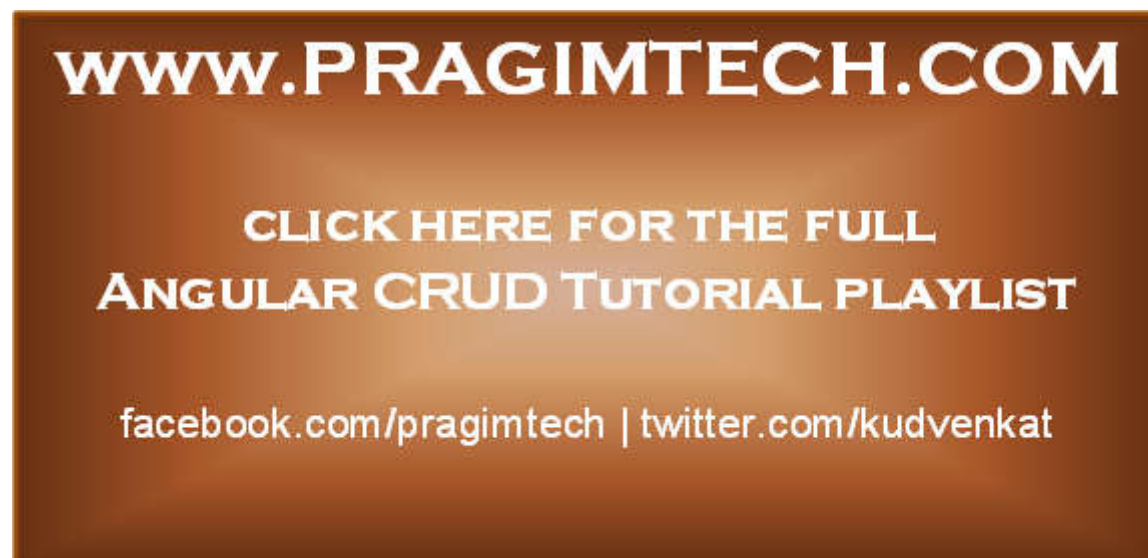
ADO.NET

ASP.NET

```
}
```

At this point, fire up the browser and navigate to <http://localhost:3000/employees/> to see the list of all employees. You can test this REST API using a tool like fiddler.

In our upcoming videos we will discuss performing CRUD operation using this fake REST API.



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.

[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](#)

Links to this post

[Create a Link](#)

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Powered by [Blogger](#).

