

# Web API with Swagger and Postman Demonstration

## Web API with Swagger and Postman

### 1. Create Web API using .NET Core with Swagger

#### Step 1: Create a Web API Project

- In Visual Studio:

File > New > Project > Choose ASP.NET Core Web API

Target Framework: .NET Core 6.0 or higher

#### Step 2: Install NuGet Package

Install-Package Swashbuckle.AspNetCore

#### Step 3: Configure Swagger in Program.cs

```
var builder = WebApplication.CreateBuilder(args);  
  
builder.Services.AddControllers();  
  
builder.Services.AddEndpointsApiExplorer();  
  
builder.Services.AddSwaggerGen(c =>  
{  
    c.SwaggerDoc("v1", new Microsoft.OpenApi.Models.OpenApiInfo  
    {  
        Title = "Swagger Demo",  
        Version = "v1",  
        Description = "TBD",  
        TermsOfService = new Uri("http://example.com/terms"),  
        Contact = new Microsoft.OpenApi.Models.OpenApiContact  
        {  
            Name = "John Doe",  
            Email = "john@xyzmail.com",  
            Url = new Uri("http://www.example.com")
```

## Web API with Swagger and Postman Demonstration

```
    },  
    License = new Microsoft.OpenApi.Models.OpenApiLicense  
    {  
        Name = "License Terms",  
        Url = new Uri("http://www.example.com")  
    }  
});  
});  
  
var app = builder.Build();  
if (app.Environment.IsDevelopment())  
{  
    app.UseSwagger();  
    app.UseSwaggerUI(c =>  
    {  
        c.SwaggerEndpoint("/swagger/v1/swagger.json", "Swagger Demo v1");  
    });  
}  
app.UseAuthorization();  
app.MapControllers();  
app.Run();
```

### Step 4: Create a Simple Controller

```
[Route("api/[controller]")]  
[ApiController]  
public class EmployeeController : ControllerBase  
{  
    [HttpGet]  
    [ProducesResponseType(200)]  
    public ActionResult<IEnumerable<string>> GetEmployees()
```

## Web API with Swagger and Postman Demonstration

```
{  
    return new string[] { "John", "Doe", "Jane" };  
}
```

```
[HttpPost]  
[ProducesResponseType(201)]  
public IActionResult AddEmployee([FromBody] string employee)  
{  
    return Created("", $"Employee {employee} added.");  
}  
}
```

Run the application and open: <https://localhost:<port>/swagger>

### 2. Use Postman Tool

- Open Postman and create a new Collection
- GET: <https://localhost:<port>/api/employee>
- POST: <https://localhost:<port>/api/employee>  
Header: Content-Type: application/json  
Body: "Michael"

### 3. Modify Route Attribute

Change Controller Route:

```
[Route("api/emp")]
```

Access via:

- GET: <https://localhost:<port>/api/emp>

### 4. Use of [ActionName] and Named Routes

## Web API with Swagger and Postman Demonstration

```
[HttpGet("all")]
[ActionName("AllEmployees")]
public ActionResult<IEnumerable<string>> GetAll()
{
    return new string[] { "All", "Employees", "Listed" };
}
```

Now call:

GET <https://localhost:<port>/api/emp/all>