1. Create new Asp.net project
2. Install NuGet package: dotnet add package Swashbuckle.AspNetCore
3. Add new controller
   1. EmployeeController.cs

| using Microsoft.AspNetCore.Mvc;  using WebApplication1.Models;  namespace WebApplication1.Controllers  {  [ApiController]  [Route("api/emp")] // Will be /api/employee  public class EmployeeController : ControllerBase  {  private static List<Employee> employees = new()  {  new Employee { Id = 1, Name = "arun", Department = "IT" },  new Employee { Id = 2, Name = "Neha", Department = "HR" }  };  [HttpGet]  [ProducesResponseType(typeof(IEnumerable<Employee>), 200)]  public ActionResult<IEnumerable<Employee>> Get()  {  return Ok(employees);  }  [HttpPost]  [ProducesResponseType(typeof(Employee), 201)]  public ActionResult<Employee> Post(Employee emp)  {  emp.Id = employees.Max(e => e.Id) + 1;  employees.Add(emp);  return CreatedAtAction(nameof(Get), new { id = emp.Id }, emp);  }  }} |
| --- |

1. New folder Model and add a class as
   1. Employee.cs

| namespace WebApplication1.Models  {public class Employee  {  public int Id { get; set; }  public string Name { get; set; }  public string Department { get; set; }  }  } |
| --- |

1. program.cs

| using Microsoft.OpenApi.Models;  var builder = WebApplication.CreateBuilder(args);  builder.WebHost.UseUrls("http://localhost:5050");  builder.Services.AddControllers();  builder.Services.AddEndpointsApiExplorer();  builder.Services.AddSwaggerGen(c =>  {  c.SwaggerDoc("v1", new OpenApiInfo  {  Title = "Swagger Demo",  Version = "v1",  Description = "TBD",  TermsOfService = new Uri("https://none"),  Contact = new OpenApiContact  {  Name = "John Doe",  Email = "john@xyzmail.com",  Url = new Uri("https://www.example.com")  },  License = new OpenApiLicense  {  Name = "License Terms",  Url = new Uri("https://www.example.com")  }  });  });  var app = builder.Build();  app.UseSwagger();  app.UseSwaggerUI(c =>  {  c.SwaggerEndpoint("/swagger/v1/swagger.json", "Swagger Demo");  });  app.UseAuthorization();  app.MapControllers();  app.Run(); |
| --- |

