* **Explain the concept of RESTful web service, Web API & Microservice:**

RESt- Representational state transfer is an architecture style for distribution hypermedia systems. A RESTful service is a web service that defines how web services should be designed for optimal performance , scalability and modifiability.

* **Features of REST architecture:**
  + **Client-Server:** A strict separation between the client and the server.
  + **Stateless:** Each request from a client to the server must contain all the information needed to understand the request.
  + **Cacheable:** Responses from the server should explicitly or implicitly define themselves as cacheable or non-cacheable.
  + **Layered System:** A client cannot ordinarily tell whether it is connected directly to the end server, or to an intermediary along the way.
* **Explain what is HttpRequest & HttpResponse:**
  + **HttpRequest:** An HttpRequest is a message sent from a client (e.g., a web browser, a mobile app, another server) to a server to request an action on a specific resource.
  + An HttpResponse is a message sent from the server back to the client in response to an HttpRequest.
* **List the types of Action Verbs:**
  + **HttpGet:** Used to request data from a specified resource. It should *only* retrieve data and have no other side effects on the server.
  + **HttpPost**:Used to submit data to be processed to a specified resource, typically resulting in the creation of a new resource.
  + **HttpDelete**:Used to request the removal (deletion) of a specified resource.
* **List the types of HttpStatusCodes used in WebAPI:**
  + **100 series:**informative
  + **200 series:** success .
  + **400 series** : Client Error
  + **500 series** : server Error
* **List the types of HttpStatusCodes used in WebAPI:**
  + Create New Project in vs and choose ASP.NET core
  + Create new controller and name it as ValuesController.cs.And add classes [product.cs](http://product.cs)
* **ProductController.cs**

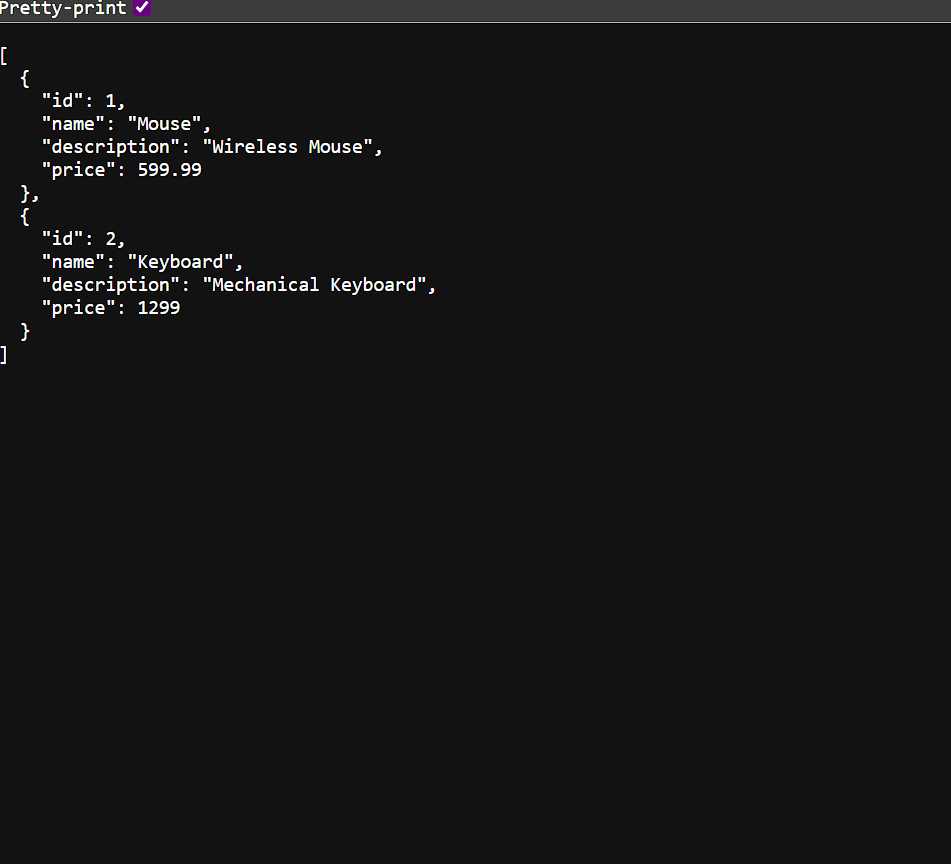
| using Microsoft.AspNetCore.Mvc;  using System.Collections.Generic;  using WebApplication1.Models;  namespace WebApplication1.Controllers  {  [ApiController]  [Route("api/[controller]")]  public class ProductController : ControllerBase  {  private static List<Product> products = new List<Product>  {  new Product { Id = 1, Name = "Mouse", Description = "Wireless Mouse", Price = 599.99 },  new Product { Id = 2, Name = "Keyboard", Description = "Mechanical Keyboard", Price = 1299.00 }  };  [HttpGet]  public ActionResult<IEnumerable<Product>> Get()  {  return Ok(products);  }  [HttpGet("{id}")]  public ActionResult<Product> GetById(int id)  {  var product = products.Find(p => p.Id == id);  if (product == null)  return NotFound();  return Ok(product);  }  [HttpPost]  public ActionResult<Product> Post(Product product)  {  products.Add(product);  return CreatedAtAction(nameof(GetById), new { id = product.Id }, product);  }  [HttpPut("{id}")]  public IActionResult Put(int id, Product updatedProduct)  {  var index = products.FindIndex(p => p.Id == id);  if (index == -1)  return NotFound();  products[index] = updatedProduct;  return NoContent();  }  [HttpDelete("{id}")]  public IActionResult Delete(int id)  {  var product = products.Find(p => p.Id == id);  if (product == null)  return NotFound();  products.Remove(product);  return NoContent();  }  }  } |
| --- |

* **Program.cs**

| var builder = WebApplication.CreateBuilder(args);  builder.WebHost.UseUrls("http://localhost:5000");  builder.Services.AddControllers();  var app = builder.Build();  app.UseHttpsRedirection();  app.UseAuthorization();  app.MapControllers();  app.Run(); |
| --- |

* **product.cs**

| **using System;**  **namespace WebApplication1.Models**  **{**  **public class Product**  **{**  **public int Id { get; set; }**  **public string Name { get; set; }**  **public string Description { get; set; }**  **public double Price { get; set; }**  **}**  **}** |
| --- |

****