1. WebApi\_Handson Objectives Answers

# 1. Explain the concept of RESTful web service, Web API & Microservice

- REST (Representational State Transfer) is an architectural style for building APIs over HTTP.  
- Web API is a framework in .NET for building RESTful services.  
- Microservices are small, independent services that handle specific business functionality.  
  
Features of REST architecture:  
• Stateless communication  
• Message-based using HTTP  
• Not restricted to XML (supports JSON, etc.)  
  
Difference between WebService & WebAPI:  
• WebService uses SOAP/XML  
• WebAPI uses HTTP/JSON  
• WebAPI is lightweight and RESTful

# 2. Explain what is HttpRequest & HttpResponse

- HttpRequest represents data sent by the client (URL, method, headers, body).  
- HttpResponse is what the server returns (status code, body, headers).

# 3. List the types of Action Verbs

- HttpGet: Retrieves data  
- HttpPost: Sends data to create a resource  
- HttpPut: Updates existing data  
- HttpDelete: Deletes a resource  
  
These are declared using attributes like [HttpGet], [HttpPost], etc. in Web API controllers.

# 4. List the types of HttpStatusCodes used in WebAPI

- 200 OK: Request successful (use return Ok())  
- 400 BadRequest: Invalid client request (use return BadRequest())  
- 401 Unauthorized: Authentication failed (use return Unauthorized())  
- 500 InternalServerError: Server-side error (use return StatusCode(500))

# 5. Demonstrate creation of a simple WebAPI - With Read, Write actions

Structure:  
- Controller inherits from ControllerBase or ApiController  
- Uses action methods with HTTP verb attributes  
  
Example:  
[ApiController]  
[Route("api/[controller]")]  
public class ValuesController : ControllerBase {  
 [HttpGet] public ActionResult<IEnumerable<string>> Get() => new[] { "value1", "value2" };  
 [HttpGet("{id}")] public ActionResult<string> Get(int id) => $"value {id}";  
 [HttpPost] public IActionResult Post([FromBody] string value) => Ok($"Received: {value}");  
 [HttpPut("{id}")] public IActionResult Put(int id, [FromBody] string value) => Ok($"Updated {id}");  
 [HttpDelete("{id}")] public IActionResult Delete(int id) => Ok($"Deleted {id}");  
}

# 6. Explain the types of Configuration files of WebAPI

- Program.cs/Startup.cs: Configures services and HTTP pipeline  
- appsettings.json: Stores app-level configuration (e.g., logging, connection strings)  
- launchSettings.json: Configures environment settings and launch URLs  
  
In .NET Framework 4.5:  
- WebApiConfig.cs: Configures Web API routes  
- RouteConfig.cs: Used in MVC, not Web API