1. WebApi\_Handson Objectives Answers

# 1. Demonstrate Swagger installation to WebAPI and WebAPI listing on browser

- Install NuGet package: Swashbuckle.AspNetCore  
- Add in Program.cs or Startup.cs:  
 services.AddSwaggerGen(...)  
  
- In app configuration:  
 app.UseSwagger();  
 app.UseSwaggerUI(...)  
  
- Add [ProducesResponseType(typeof(Type), StatusCode)] on methods for better documentation.  
- Run the application and navigate to: https://localhost:[port]/swagger to see API listing.

# 2. Demonstrate the usage of Postman tool to hit WebAPI methods

- Open Postman and create a new Request or Collection.  
- Select Request Type (GET, POST, etc.) from the dropdown.  
- Enter API endpoint (e.g., https://localhost:port/api/employee).  
- Use 'Headers' tab to set Authorization headers if needed.  
- Use 'Body' tab to send JSON input for POST/PUT.  
- The center pane displays request/response tabs (Body, Headers, Status).  
- You can save requests into Collections for reuse.

# 3. Demonstrate the usage of Route and Explain Name attribute in Http requests

- Use [Route("api/[controller]")] to define controller routes.  
- Use [HttpGet("custompath", Name = "FriendlyName")] to assign a user-friendly route name.  
- The Name attribute helps with link generation and documentation.  
- Use [ActionName("CustomAction")] to allow multiple methods with same HTTP verb but different logic:  
  
 Example:  
 [HttpGet]  
 [ActionName("GetById")]  
 public IActionResult GetById(int id) { ... }  
  
 [HttpGet]  
 [ActionName("GetByName")]  
 public IActionResult GetByName(string name) { ... }

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

- Create a controller (e.g., EmployeeController) inheriting from ControllerBase.  
- Add methods with appropriate HTTP verb attributes:  
  
 [HttpGet] → public IActionResult Get() { return Ok(...); }  
 [HttpPost] → public IActionResult Post([FromBody] Employee emp) { return Ok(...); }  
 [HttpPut] → public IActionResult Put(int id, [FromBody] Employee emp) { ... }  
 [HttpDelete] → public IActionResult Delete(int id) { ... }  
  
- Test endpoints via Swagger or Postman to verify functionality.