WebApi Handson 4 Question and Answers

1. Demonstrate creation of an Action method to perform data create, update & delete

- Use of [FromBody] for Input
  - [FromBody] binds incoming JSON data to a C# model class.
  - Required for POST and PUT where body content is JSON.

- Model Class Example

```
public class Employee
{
    public int Id { get; set; }
    public string Name { get; set; }
    public int Salary { get; set; }
}
```

- Hardcoded Employee List

```
private static List<Employee> empList = new List<Employee>
{
    new Employee { Id = 1, Name = "John", Salary = 50000 },
    new Employee { Id = 2, Name = "Jane", Salary = 60000 }
};
```

- Create - POST Method

```
[HttpPost]
public IActionResult Create([FromBody] Employee emp)
{
    empList.Add(emp);
    return Ok(emp);
}
```

- Update - PUT Method

```
[HttpPut("{id}")]
public IActionResult Update(int id, [FromBody] Employee emp)
{
    var existing = empList.FirstOrDefault(e => e.Id == id);
    if (existing == null) return NotFound();
    existing.Name = emp.Name;
    existing.Salary = emp.Salary;
    return Ok(existing);
}
```

- Delete - DELETE Method

```
[HttpDelete("{id}")]
public IActionResult Delete(int id)
{
    var emp = empList.FirstOrDefault(e => e.Id == id);
    if (emp == null) return NotFound();
    empList.Remove(emp);
    return Ok("Deleted");
}
```

- Test in Swagger & Postman
  - Swagger:
    - Run the app → go to /swagger
    - Try each action (POST/PUT/DELETE)

  - Postman:
    - Set method type (POST, PUT, DELETE)
    - Set JSON in Body → raw → application/json
    - Add request headers if needed (e.g., Authorization)