```csharp
private static Lazy<AppHelper> _AppHelperObj = new Lazy<AppHelper>();


public static AppHelper AppHelperObj => _AppHelperObj.Value;

-
public delegate int CheckStatus();

-
var empList = db.Employees.Where(x=> x.Experience >= 7);

if(empList.Any()){
        empList.ForEach(x=> x.AnyColumn = "new value");
        await db.SaveChangesAsync();
}

-new bulk update in (EF Core 7+)
var empList = db.Employees.Where(x=> x.Experience >= 7).ExecuteUpdateAsync(s =>
s.SetProperty(e => e.Bonus, 15000));

emp=empid,depid,sal,  dep=depid

var salWithDepId = (from e in db.Employees
                    join d in db.Departments on e.DepId equals d.DepId
                    group e by e.DepId into g
                    select new
                    {
                        DepId = g.Key,
                        TotalSalary = g.Sum(x => x.Sal)
                    }).ToList();

-
Product=id,name,descp


[ApiController]
[Route("api/Product")]
[Authorize]
public class ProductController : ControllerBase{

private readonly IProductService _productServe;
public  ProductController(IProductService productServe){
        _productServe = productServe;
}

        [HttpGet]
        public async Task<IActionResult> GetProducts(){
          var products = await _productServe.GetProducts();
          return Ok(products);
        }

        [HttpPost]
        public async Task<IActionResult> AdProduct([FromBody]ProductDto product)
{
```

```
            bool addedProduct = await _productServe.AddProduct(product);
            return Ok(addedProduct);
        }


        [HttpPatch]
        public async Task<IActionResult>
AdProduct([FromBody]JsonPatchDocument<ProductDto> product){


        }
}
```