Read Csv

```csharp
1 reference
internal static async Task ReadProductsAsync()
{
    var lines = await File.ReadAllLinesAsync(ProductsFileName.GetFullNameInApplicationTree());
    products = lines.Skip(1)
        .Select(line => line.Split(';'))
        .Select(cols => new Product
        {
            Name = cols[0],
            Price = double.Parse(cols[1])
        })
        .Distinct()
        .ToList();
}
```

GetHTTP

```csharp
1 reference
private static async Task getHttpCategories()
{
    httpClient = new HttpClient();
    httpClient.BaseAddress = new Uri(ApiURL);
    var task = await httpClient.GetAsync("jokes/categories");
    var msg = await task.Content.ReadAsByteArrayAsync();

    categoriesString = JsonSerializer.Deserialize<List<string>>(msg);
}
```

ImportData

```csharp
1 reference
private static async Task ImportDataAsync()
{
    using var context = new ApplicationDbContext();
    await context.Facts.AddRangeAsync(jokes);
    await context.SaveChangesAsync();
    Console.WriteLine("Database filled!");
}
```

Entities

```csharp
5 references
public class Buffet : EntityObject
{
    [MaxLength(3), NotNull]
    7 references
    public string Location { get; set; }
    3 references
    public override bool Equals(object obj)
    {
        return obj is Buffet buffet && Location == buffet.Location;
    }
    2 references
    public override int GetHashCode()
    {
        return HashCode.Combine(Location);
    }
}
```

RecreateDatabase

```csharp
1 reference
private static async Task RecreateDatabaseAsync()
{
    using var context = new ApplicationDbContext();
    await context.Database.EnsureDeletedAsync();
    await context.Database.MigrateAsync();
}
```

EntityObj

```csharp
4 references
public class EntityObject : IEntityObject
{
    [Key]
    1 reference
    public int Id { get; set; }
    [Timestamp]
    1 reference
    public byte[] RowVersion
    {
        get;
        set;
    }
}
```

JsonProp Name

```csharp
[JsonPropertyName("url")]
[MaxLength(1024)]
0 references
public string Url { get; set; }
```

HTTP Variablen

```csharp
static HttpClient httpClient;
private const string ApiURL = "https://api.chucknorris.io";
```