

How to correctly use HttpClient

This is a wrong way to use **HttpClient**:

```
public class OrderService
{
    // Singleton HttpClient
    private static readonly HttpClient _client = new();

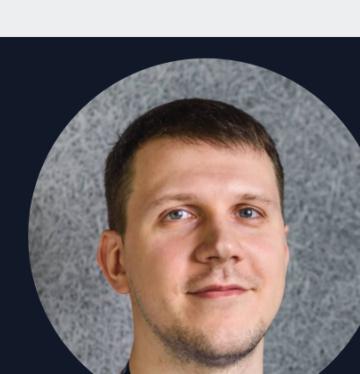
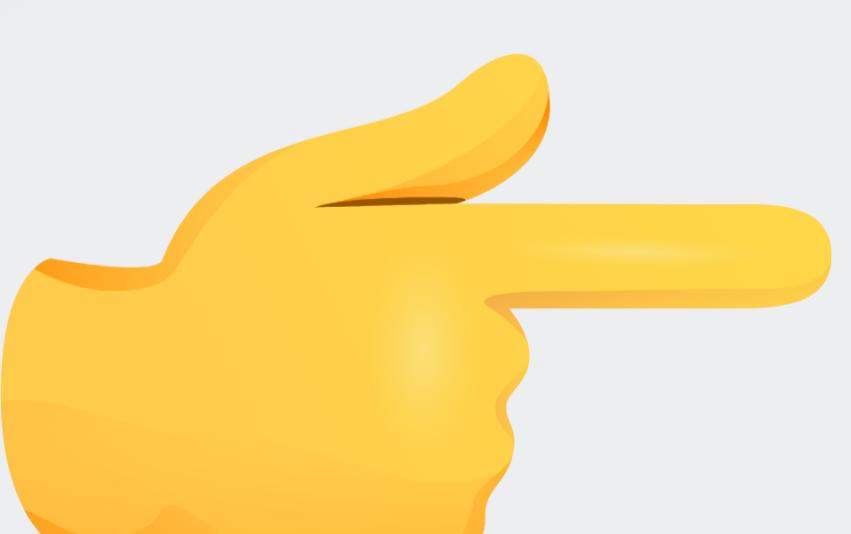
    public async Task<OrderDto> GetAsync(Guid orderId)
    {
        var response = await _client.GetAsync($"https://api.orders.com/{orderId}");
        response.EnsureSuccessStatusCode();
        var json = await response.Content.ReadAsStringAsync();
        return JsonSerializer.Deserialize<OrderDto>(json)!";
    }
}
```

Doesn't reflect DNS and Network changes 

```
public static async Task<OrderDto> GetOrderAsync(Guid orderId)
{
    // New socket every request
    await using var client = new HttpClient();

    var response = await client.GetAsync($"https://api.orders.com/{orderId}");
    response.EnsureSuccessStatusCode();
    return await response.Content.ReadFromJsonAsync<OrderDto>()!;
}
```

Can lead to Socket Exhaustion 



Anton Martyniuk

antondevtips.com

How to correctly use HttpClient

This is a **right way to use HttpClient:**

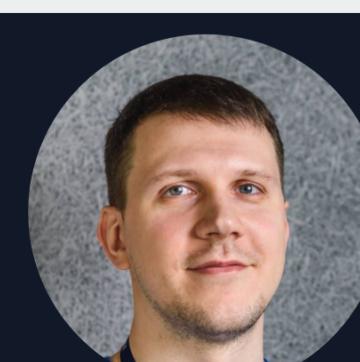
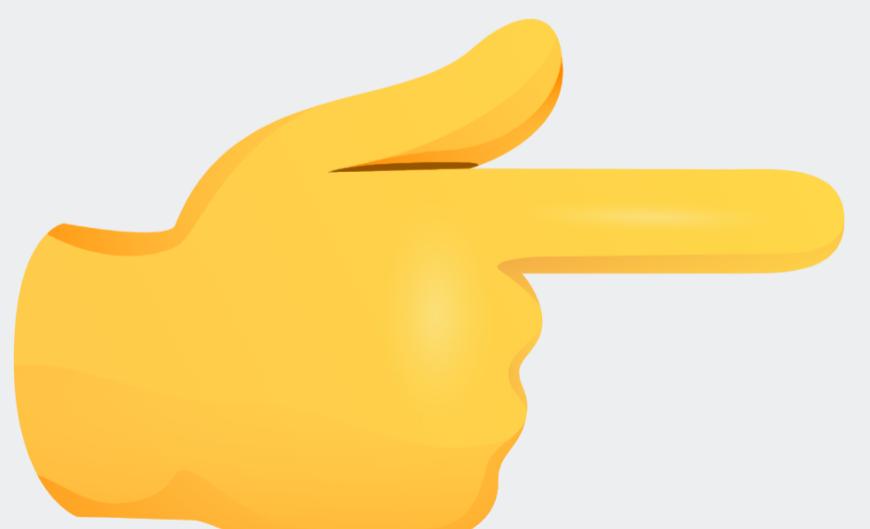
1. HttpClientFactory

```
public static class OrderService
{
    private readonly IHttpClientFactory _factory;

    public OrderService(IHttpClientFactory factory)
    {
        _factory = factory;
    }

    public static async Task<OrderDto> GetAsync(Guid orderId)
    {
        var url = $"https://api.orders.com/{orderId}";
        var client = factory.CreateClient();
        var response = await client.GetAsync();
    }
}

var builder = WebApplication.CreateBuilder(args);
builder.Services.AddHttpClient();
```



Anton Martyniuk

How to correctly use HttpClient

This is a **right way to use HttpClient:**

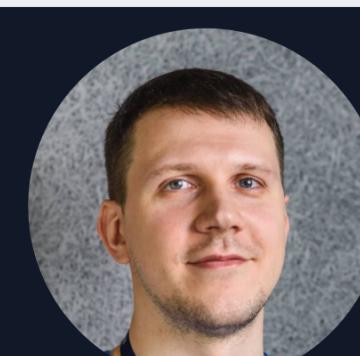
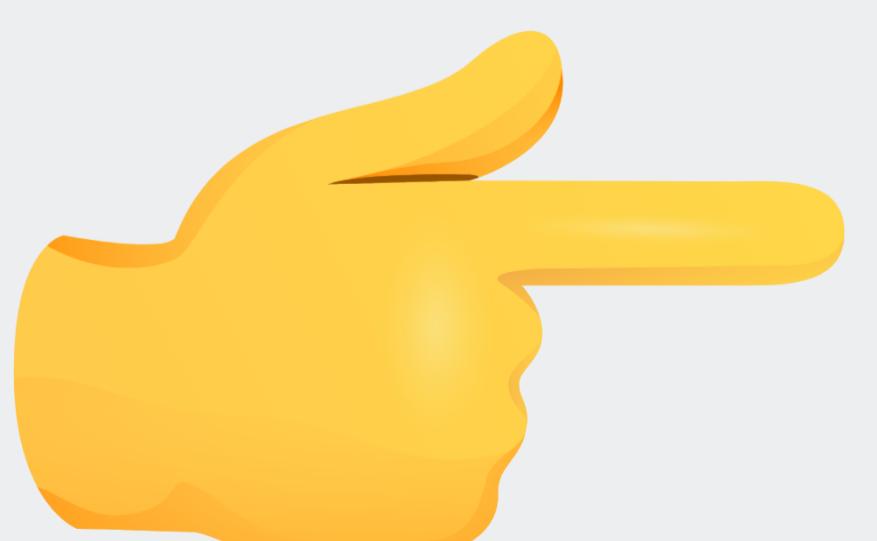
2. Typed HttpClient

```
public class OrderApiClient
{
    private readonly HttpClient _http;

    public OrderApiClient(HttpClient http)
    {
        _http = http;
        _http.BaseAddress = new Uri("https://api.orders.com/");
    }

    public async Task<OrderDto> GetAsync(Guid orderId)
    {
        var dto = await _http.GetFromJsonAsync<OrderDto>($"orders/{orderId}");
        return dto!;
    }
}

// Now you can inject OrderApiClient into your classes
var builder = WebApplication.CreateBuilder(args);
builder.Services.AddHttpClient<OrderApiClient>();
```



Anton Martyniuk

How to correctly use HttpClient

This is a **right way to use HttpClient:**

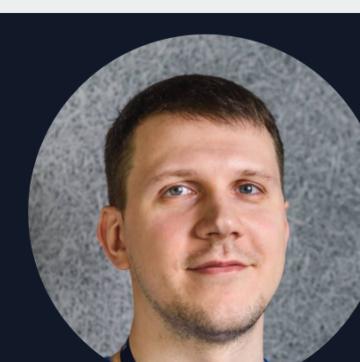
3. Refit

```
// IOrderApi.cs
public interface IOrderApi
{
    [Get("/orders/{id}")]
    Task<OrderDto> GetAsync(Guid id);
}

public record OrderDto(Guid Id, decimal Total, string Status);

// Program.cs
var builder = WebApplication.CreateBuilder(args);

builder.Services.AddRefitClient<IOrderApi>()
    .ConfigureHttpClient(c => c.BaseAddress = new Uri("https://api.orders.com"));
```



Anton Martyniuk

THANKS FOR READING



LETS CONNECT ON LINKED-IN



REPOST CONTENT TO OTHERS



**SUBSCRIBE TO [ANTONDEVTIPS.COM](#)
TO IMPROVE YOUR SKILLS IN .NET**



Anton Martyniuk
antondevtips.com

