**Вызов методов API**

public async Task<IEnumerable<Category>?> GetCategoriesAsync()

{

return await \_client.GetFromJsonAsync<IEnumerable<Category>>("categories");

}

public async Task<Category?> GetCategoryAsync(int id)

{

return await \_client.GetFromJsonAsync<Category>($"categories/{id}");

}

public async Task AddCategoryAsync(Category category)

{

await \_client.PostAsJsonAsync($"categories", category);

}

public async Task UpdateCategoryAsync(Category category)

{

await \_client.PostAsJsonAsync($"categories/{category.CategoryId}", category);

}

public async Task DeleteCategoryAsync(int id)

{

await \_client.DeleteAsync($"categories/{id}");

}

Генерация исключения для Get:

var response = await \_client.GetAsync($"categories/{id}");

response.EnsureSuccessStatusCode();

return await response.Content.ReadFromJsonAsync<Category>();

Получение заголовка(адреса) созданного объекта:

var location = response.Headers.Location;

return location.AbsoluteUri;

При обработке HttpRequestException можно проверять статус код ошибки:

if (ex.StatusCode == System.Net.HttpStatusCode.NotFound)

Console.WriteLine("bup");