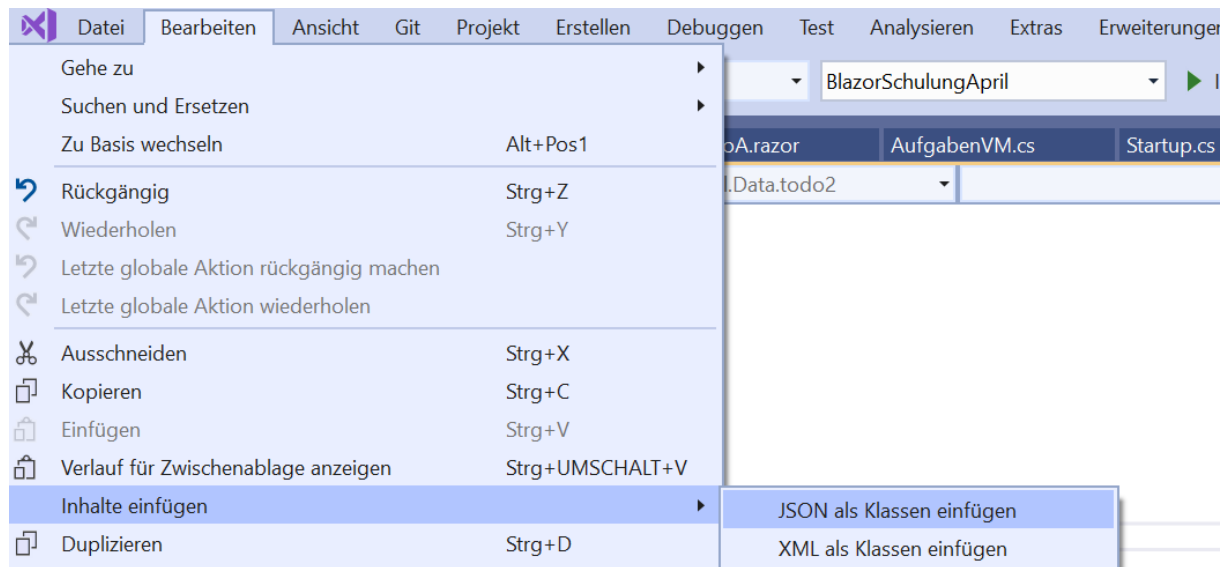


Fake Rest

<https://jsonplaceholder.typicode.com/todos>

Zwischenablage kopieren

Klasse Todo2 in models Verzeichnis – Visual Studio Paste Json as ...



Erzeugten Code ggf anpassen

```
public class Todo2
{
    public int userId { get; set; }
    public int id { get; set; }
    public string title { get; set; }
    public bool completed { get; set; }
}
```

In einer Server App ist der HttpClient nicht vorhanden und muss erst hinzugefügt werden in Startup.cs

```
services.AddHttpClient();
```

neue Razor Komponente todohttp

im Kopf (exemplarisch bitte anpassen)

```
@page "/todo2"
@inject IHttpClientFactory Http
@using System.Text.Json
@using BlazorSchulungApril.Data
```

@Code Block

Property todolist List von

```
protected override LEERTASTE (Vorschlagsliste)
```

```
@code {
```

```
protected override
```

```
}
```

- * OnAfterRender(bool firstRender)
- * OnAfterRenderAsync(bool firstRender)
- * OnInitialized()
- * OnInitializedAsync()
- * OnParametersSet()
- * OnParametersSetAsync()
- * ShouldRender()

```
var client = Http.CreateClient();

var response = await
client.GetAsync(@"https://jsonplaceholder.typicode.com/todos");
if (response.IsSuccessStatusCode)
{
    using var responseStream = await response.Content.ReadAsStreamAsync();
    todoliste = await
JsonSerializer.DeserializeAsync<List<Todo2>>(responseStream);

}
```

HTML UI

```
@if (todoliste == null)
{
    <div>lade daten</div>
}
else
{
    <table class="table table-striped">
        @foreach (var item in todoliste)
        {
            <tr>
                <td>@item.id</td>
                <td>@item.userId</td>
                <td>@item.title</td>
                <td>@item.completed</td>
            </tr>
        }
    </table>
}
```

