

DOWNLOADS

WebAssembly und Klassengenerator



Sie erhalten Tools für WebAssemblies, und Quicktype macht aus JSON wunderbare C#-Klassen.

Da war es wieder, das Problem: eine dicke, fette JSON-Datei von einem Kunden. Und nein, nicht eine JSON-Datei eines dicken, fetten Kunden. Daten dieser JSON-Struktur sollen in C#-Objekte deserialisiert werden, um die Daten dann weiterverarbeiten zu können. Nichts leichter als das.

Doch halt. Bevor Sie jetzt anfangen, Properties per Hand einzutippen, sollten Sie lieber auf eines der Tools setzen, die es im Internet und auf GitHub gibt. Quicktype [1] ist ein solches Tool. Diesem Programm übergeben Sie eine JSON-Datei, und es produziert daraus ein – hm, ja was eigentlich?

Die App produziert nicht eines, nicht zwei, sondern viele mögliche Ausgabeformate. Dazu gehören Ruby, JavaScript, Flow, Rust, Kotlin, Dart, Python, C#, Go, C++, Java, TypeScript, Swift, Objective-C, Elm und JSON Schema.

Sie landen also mit Quicktype nicht in einer Sackgasse, sondern sind flexibel. Da der Quellcode zudem auf GitHub zur Verfügung steht, können Sie ihn bei Bedarf auch an weitere Sprachen anpassen.

Wenn Sie das Tool aber nicht häufiger einsetzen, sondern nur gelegentlich eine Klassenstruktur für eine JSON-Datei benötigen, können Sie die Umwandlung auch online auf einer Website vollziehen.

Unter [2] geben Sie dazu in der linken Spalte den Quelltext des JSON ein oder kopieren ihn per Copy-and-paste hinein. Innerhalb von Sekunden erscheint dann in der rechten Spalte der Quellcode für die Klassen in der ausgewählten Sprache. Gesetzt den Fall, Ihr JSON lautet folgendermaßen:

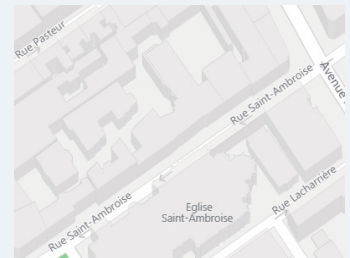
Schwerpunkt WebAssembly

Als wir den Schwerpunkt festgelegt haben, war uns nicht bewusst, wie viele Tools und Bibliotheken es schon für diese neue Webtechnologie gibt. Die Attraktivität ist klar: Statt ein Frontend in JavaScript schreiben zu müssen, kann man nun die Sprache der Wahl verwenden und sie in WebAssembly übersetzen lassen.



GMap.NET – Maps für Windows

Mit GMap.NET können Sie mit wenigen Zeilen Code Landkarten in Ihre Windows-Forms-Anwendungen einbauen. Die Karten kommen dabei von Bing, Google, OpenStreetMap und anderen.



```
{
  "rohkost": {
    "gemüse" : [
      "Sellerie"
    ],
    "obst" : [
      "Apfel"
    ]
  }
}
```

Dann wird mit Quicktype daraus beispielsweise:

```
public partial class VegetarischesEssen
{
  [JsonProperty("rohkost")]
  public Rohkost Rohkost { get; set; }
}
public partial class Rohkost
{
  [JsonProperty("gemüse")]
  public string[] Gemüse { get; set; }
  [JsonProperty("obst")]
  public string[] Obst { get; set; }
}
```

[1] <https://github.com/quicktype/quicktype>

[2] <https://app.quicktype.io>

● **Schwerpunkt**

WasmWinforms

Mit WasmWinforms können Sie Ihre WinForms-Formulare im Browser ausführen.

WebAssembly für .NET

Diese Library kann WebAssembly-Dateien aus .NET-basierten Applikationen erstellen, lesen, modifizieren, schreiben und ausführen. Bei der Ausführung wird kein Interpreter verwendet.

Wasm2CIL

Wasm2CIL ist ein .NET-Compiler für WebAssembly-Module, einschließlich Support für WASI. Es kann ein WebAssembly-Modul in eine .NET-Assembly auf Disk übersetzen. Die resultierende DLL ist nicht interpretiert.

vscode-wasm

WebAssembly-Toolkit für VS Code.

Uno.Wasm.Bootstrap

Uno.Wasm.Bootstrap bietet eine einfache Möglichkeit, eine C# .NET Standard 2.0 Library zu verpacken und sie in einer kompatiblen Browserumgebung auszuführen. Es handelt sich um einen eigenständigen Mono-Web-Assembly-SDK-Bootstrapper in Form eines NuGet-Packages.

Uno Quick Start

Dieses Repository ist ein Basismuster für eine Uno-Applikation für UWP, iOS, Android sowie WebAssembly. Verwenden Sie Visual Studios UWP-Tools auf Windows, wie zum Beispiel Xaml Edit und Continue, C# Edit und Continue, dann führen Sie Ihre App auf iOS, Android und Windows aus.

WasmLib

C#-Bibliothek zum Lesen, Ändern und Schreiben von Web-Assembly-Binärdateien (Wasm).

dna-redux

Ausführen von C#/F#-Code in Node.js oder im Browser mit WebAssembly.

WebAssembly- (Wasm) und Webpack-Starter-Projekt

Das Erstellen der WebAssembly-Binary (Wasm) basiert auf diesem Tutorial. Aber es werden Docker-Container verwendet, um das Teilen der Compiler-Toolkette effizienter zu machen.

Nethereum-Blazor-Wasm

Dieses Repo bietet einen Ansatzpunkt zum Testen der experimentellen .NET-Wasm/Blazor-Kompatibilität mit Nethereum.

dotWasmBenchmark

Kleiner Benchmark zum Vergleichen von .NET Core, Mono, mono-wasm und JavaScript.

● **Aktuelle Downloads**

Azure SignalR Service SDK für .NET

Azure SignalR Service SDK für .NET hilft beim Erstellen von Azure-Anwendungen mit Echtzeit-Nachrichtenfunktion, mit dem Vorteil skalierbarer Cloud-Computing-Ressourcen.

DotNetAutoTestExtras

DotNet-Automatisierungstest für API, Web, Mobile und Performance.

AspnetRun

Diese ASP.NET-Core-Referenzanwendung mit Entity Framework Core zeigt eine Anwendungsarchitektur mitDDD-Best-Practices. Sie implementiert NLayer-Hexagonal-Architektur (Core, Application, Infrastructure and Presentation Layers) und Domain Driven Design (Entities, Repositories, Domain/Application Services, DTOs et cetera).

Harmony Core

Das Framework Harmony Core besteht aus Libraries, CodeGen-Templates und Konventionen, mit deren Hilfe Sie Synergy-Logik und -Daten unter Verwendung von OData und ASP.NET Core als RESTful-Webdienst verfügbar machen können.

Avalonia

Avalonia ist ein von WPF inspiriertes, plattformübergreifendes und XAML-basiertes UI-Framework, das ein flexibles Styling-System bietet und eine Vielzahl von Betriebssystemen unterstützt: Windows (.NET Framework, .NET Core), Linux (GTK), macOS, Android und iOS.

ADN.Extensions

ADN.Extensions ist eine plattformübergreifende Open-Source-Bibliothek, die .NET-Entwicklern Erweiterungsmethoden zum Beispiel für Arrays oder Strings bietet.

DocFX

Mit DocFX können Sie ganz einfach Ihren Entwickler-Hub mit API-Referenz, Zielseite und Anleitungen erstellen.

Fast.Data.Core

.NET Core ORM (DB First, Code First) für SQL Server, MySQL et cetera.

ZKWeb

Ein flexibles Web-Framework für .NET Framework und .NET Core, das unter anderem ein Plug-in-System, ein MVC Framework und ein Template-System bietet.

core-setup

Dieses Repository enthält den Code zum Erstellen der .NET Core Runtime, Libraries und Shared-Host-Installationsprogramme für alle unterstützten Plattformen.